Imperial College of Science, Technology and Medicine

Royal School of Mines

Department of Bioengineering

Ultrasound Imaging Augmented 3D Flow Reconstruction and Computational Fluid Dynamics Simulation

by

Xinhuan Zhou

A thesis submitted for the degree of
Doctor of Philosophy of Imperial College London
October 2019

Abstract

Cardiovascular Diseases (CVD), including stroke, coronary/peripheral artery diseases, are currently the leading cause of mortality globally. CVD are usually correlated to abnormal blood flow and vessel wall shear stress, and fast/accurate patient-specific 3D blood flow quantification can provide clinicians/researchers the insights/knowledge for better understanding, prediction, detection and treatment of CVD. Experimental methods including mainly ultrasound (US) Vector Flow Imaging (VFI), and Computational Fluid Dynamics (CFD) are usually employed for blood flow quantification. However current US methods are mainly 1D or 2D, noisy, can only obtain velocities at sparse positions in 3D, and thus have difficulties providing information required for research and clinical diagnosis. On the other hand while CFD is the current standard for 3D blood flow quantification it is usually computationally expensive and suffers from compromised accuracy due to uncertainties in the CFD input, e.g., blood flow boundary/initial condition and vessel geometry.

To bridge the current gap between the clinical needs of 3D blood flow quantification and measurement technologies, this thesis aims at: 1) developing a fast and accurate 3D full field flow reconstruction algorithm, using existing sparse 1D/2D US velocity measurements; 2) developing GPU based and US augmented Lattice Boltzmann Method (LBM) solver for faster and more accurate blood CFD computation. The thesis has made the following contributions:

(1) Chapter 2 proposed a new algorithm for 3D flow reconstruction using Divergence Free Interpolation (DFI) of sparse 2D in-plane velocity vectors obtained from US particle imaging velocimetry (UIV), with interpolation spatial basis functions which satisfy mass conservation. It is validated via two numerical reconstruction cases, and the reconstruction has similar accuracy with CFD and is ~4 times faster than CFD

simulation for a range of problems. The proposed algorithm is also demonstrated and evaluated using experimental 2D UIV measurements.

(2) Chapter 3 has conducted optimizations to improve the DFI based 3D flow reconstruction approach in terms of accuracy and the time-to-solution, through optimizing the interpolation spatial basis function parameters and using a highly optimized iterative solver Generalized Minimal Residual method (GMRES) which is faster than the SVD solver used in the Chapter 2. The optimized algorithm is validated by successfully reconstructing 3D flows of *in silico* and *in vitro* UIV measurement cases, and the results show improved accuracy of ~3% and up to 728-fold speedup compared with previous method in Chapter 2. In Appendix A1 3D flow reconstruction using 1D vector US Doppler imaging has also been studied, and the feasibility of the method is demonstrated numerically by reconstructing 3D flow of a widely used benchmark (i.e., lid driven cavity flow). The method has a typical reconstruction time of <1s and error of ~5%, and can be ~800 times faster than CFD with slightly reduced but still acceptable accuracy.

(3) Patient-specific CFD suffers from high computational overheads and compromised accuracy due to input uncertainties. A measurement augmented GPU based Lattice Boltzmann Method (LBM) CFD solver is developed in order to speed up LBM simulation and reduce LBM error caused by such uncertainties, as shown in Chapter 4. We assume CFD inlet/outlet velocities and internal flow measurements are from the same US measurements and thus have similar level of fidelity. US VFI internal measurements are integrated into LBM CFD solver to constrain the simulation and augment it in terms of convergence speed and accuracy. The algorithm is validated with both *in silico* and *in vitro* cases, and the results show significant improvements of convergence speed (>30%) and accuracy compared with similar CFD without measurement augmentation. Comparing to the divergence free interpolation method, such estimation combines the advantages of the full physics in CFD and the sparse US measurements. In Appendix A2 the solver is then extended to consider the non-Newtonian rheological effects and validated *in silico* with a low error of ~1% using well-known commercial CFD software as a reference.

The proposed algorithms can be potentially employed to clinical applications/research, for faster and more accurate 3D blood flow quantifications.

Affirmation

The work submitted in this thesis is my own, and has not been submitted previously for any other degree. Results obtained in the course of this research have been presented in the following journal publications.

PEER-REVIEWED PAPERS

- **Zhou, X.**, Papadopoulou, V., Leow, C. H., Vincent, P., Tang, M. X* (2019). 3-D Flow Reconstruction Using Divergence-Free Interpolation of Multiple 2-D Contrast-Enhanced Ultrasound Particle Imaging Velocimetry Measurements. Ultrasound in medicine & biology, 45(3), 795-810.

- **Zhou, X.**, Vincent, P., Zhou, Xiaowei, Leow, C. H., Tang, M. X*. Optimization of 3D Divergence Free Flow Field Reconstruction Using 2D Ultrasound Vector Flow Imaging. Ultrasound in medicine & biology (2019)

- Zhou, Xiaowei, **Zhou, Xinhuan**, Leow, C. H., Tang, M. X*. Measurement of Flow Volume in the Presence of Reverse Flow with Ultrasound Speckle Decorrelation. Ultrasound in Medicine & Biology (2019)

SUBMITTED PAPERS

- **Zhou, X**., Vincent, P., Riemer, K., Zhou, Xiaowei, Leow, C. H., Tang, M. X*. Measurement Augmented 3D Lattice Boltzmann Flow Simulation for Convergence Acceleration and Uncertainty Suppression. (Submitted to the Journal of Biomechanics).

CONFERENCE PROCEEDINGS

- Papadopoulou, V., Corbett, R., **Zhou, X**, Toulemonde, M., Leow, C. H., Cosgrove, D., Duncan, N., and Tang, M.X. *. "3D flow velocity reconstruction in a human radial artery from measured 2D high-frame-rate plane wave contrast enhanced ultrasound in two scanning directions—A feasibility study." In 2017 IEEE International Ultrasonics Symposium (IUS), pp. 1-1. IEEE, 2017.

- **Zhou, X**, Zhou, Xiaowei, Leow, C. H., Vincent, P., and Tang, M.X. *. "3D Flow Reconstruction and Wall Shear Stress Evaluation with 2D Ultrafast Ultrasound

Particle Imaging Velocimetry." In 2018 IEEE International Ultrasonics Symposium (IUS), pp. 1-4. IEEE, 2018.
(Note: corresponding authors are marked with *)

Copyright Declaration

Hereby I declare that I am the author of this thesis, and that it is the product of my own work. This dissertation has not been submitted for consideration for any other degree or awards. I have fully acknowledged the ideas and results from the work of other people where appropriate. The copyright of this thesis rests with the author and is made available under a Creative Commons Attribution Non-Commercial No Derivatives licence. Researchers are free to copy, distribute or transmit the thesis on the condition that they attribute it, that they do not use it for commercial purposes and that they do not alter, transform or build upon it. For any reuse or redistribution, researchers must make clear to others the licence terms of this work.

Acknowledgements

I am also grateful for the help and support from many of my Chinese friends in Imperial College London including: Dr. Wenzhuo Cao, Mr. Jinrui Guo, Mr. Weifeng Wan, Mr. Han Yuan, Dr. Xiaofei Xiao. I cherish their priceless friendship, company and knowledge with the years very much.

Finally I would like express my special thanks to my family for their continuous support, encouragement, and love through the years. I could not have done it without you.

Table of Contents

List of Figures

List of Tables

Nomenclature

Abbreviation	Description
AOS	Array of Structure
BC	Boundary Condition
IC	Initial Condition
CFD	Computational Fluid Dynamics
CVD	Cardiovascular Diseases
DFI	Divergence Free Interpolation
ECG	electrocardiogram
FSI	Fluid Structure Interaction
GCV	Generalized Cross Validation
GMRES	Generalized Minimum Residual
GPU	Graphic Processing Unit
IM	inverse multiquadric
LBM	Lattice Boltzmann Method
MLUPS	Million Lattice Updates per Second
MU	multiquadric
NCP	Normalized Cumulative Periodogram
PCMRI	Phase Contrast Magnetic Resonance Imaging
RBC	Red Blood Cells
RBF	Radial Basis Function
RMS	Root Mean Square
TPS	Thin Plate Spline
UIV	Ultrasound Imaging Velocimetry
VFI	Vector Flow Imaging
WSS	Wall Shear Stress

Symbol	Description
A	product of R and G, $A = R * G$
b	product of R and $\vec{v}$, $b = R * \vec{v}$
$\vec{c}_i$	velocity of LBM virtual particles in the ith direction
c_s	speed of sound
D	diagonal matrix
D_{II}	the second invariant of the strain rate tensor
$\vec{F}$	external force in the Boltzmann equation
f	particle probability density function
f^0	f at equilibrium
$f_i^{(1)}$	non-equilibrium part of f in the ith direction, i.e., $f_i^{(1)} = f - f^0$

G	divergence free interpolation matrix
K_k	the kth krylov subspace
P	permutation matrix
$P_{\vec{B}}$	projection matrix for vector $\vec{B}$
Q	orthogonal matrix
R	combined projection matrix
$S_{\alpha\beta}$	strain rate tensor
U, Σ, V	U and V are unitary matrices, Σ is a diagonal matrix of singular values of A
$\vec{u}(\vec{x}, t)$	fluid velocity at location $\vec{x}$ and time t
$\vec{v}$	combined velocity vector of the flow fields
$\vec{v}_{\vec{B}}$	velocity component along the direction of vector $\vec{B}$ or onto the plane whose normal is $\vec{B}$
w_i	weights of LBM virtual particles in the ith direction
$\varphi(\vec{r})$	the scalar RBF
$\emptyset(\vec{x}_j, \vec{c}_i)$	matrix form divergence free RBF
λ	the weighting coefficient of RBFs
λ_{reg}	regularized approximation to λ
τ	the regularization parameter
σ_i	the ith singular value of A
$\rho(\vec{x}, t)$	fluid density at location $\vec{x}$ and time t
τ	LBM relaxation time
α	the LBM regularization/relaxation parameter
μ	flow dynamic viscosity
$\dot{r}$	flow shear rate
μ_0, μ_∞	flow dynamic viscosity at zero and infinite shear rate
τ_∞, τ_0	shear dependent relaxation time at zero and infinite shear rate

Chapter 1 Introduction and objectives

1.1   Research background

Cardiovascular Diseases (CVD), defined as disorders of the heart and blood vessels, are the leading cause of mortality in the industrialized world and lead to billions of dollars in annual health care expenditures. 31% of the total deaths (~17.9 million) across the world is estimated to be caused by CVD each year, among which 85% are due to strokes and heart attacks. In England, CVD account for 34% of the deaths while for EU for 40% (1). The global rate of CVD is still increasing, mainly due to the increasing level of metabolic diseases and age (2).

CVD include a series of diseases, such as coronary heart disease, rheumatic heart disease, angina, heart failure/attack and aortic diseases (3). It is of great clinical interest to develop fast and accurate CVD diagnosis tools. An example is blood clot that forms in vessels may detach and travel to brain causing stroke and sudden death (4). CVD are usually associated with artery damages and atheroma, and have an important effect on the heart, kidney, brain or blood vessels functionalities. The exact cause of CVD is still unclear but there are many risk factors, such as obesity and diabetes (5), high blood pressure (6), infectious diseases (7), genetic predisposition (8), smoking and inactivity (9), for example heart failure occurs due to large myocardial infarctions and its incidence increases with age.

CVD are extensively correlated with abnormal blood flow patterns and Wall Shear Stress (WSS)(10). An example is coronary heart disease, which occurs when plaque (a combination of fat, cholesterol, calcium and other blood substances) builds up in arteries and leads to increased risk of atherosclerosis and clogged arteries (termed thrombosis) (11). As a result of reduced blood supply, the plaque reduces oxygen and nutrients supply to organs including the brain, and may cause chest pain, heart attack, brain damage and other symptoms. The frequent occurrence of atherosclerosis in the regions of bifurcation, curvature and branching of vessels (12), suggests that complex blood flow patterns and vessel geometry may lead to localized plaque formation (13). In addition *in vivo* and *in vitro* results show that highly oscillatory and low WSS appear to be more likely to be linked to the pathogenesis of atherosclerosis in a complex manner (14, 15). The elongation and orientation of cultured endothelial cells in response to shear stress show that WSS can change the endothelial cell layer morphology and orientations, i.e., endothelial cells exposed to increased WSS tend to

elongate and align in the flow direction, while those exposed to low/oscillatory WSS have no preferred alignment pattern (16). Okano et al. (17) show that intercellular permeability increases when arteries are exposed to low and oscillatory WSS, resulting in increased vulnerability to atherosclerosis. The study found that non-pulsatile high WSS leads to an increase of artery permeability, and due to that the release of factors from the endothelial cells inhibit coagulation, leukocytes migration and smooth muscle proliferation, as well as benefit endothelial cells survival (18). To the contrary, low and oscillatory WSS leads to the development of atherosclerosis (14, 19).

Previously the detection of atherosclerosis and risk of stroke relied on catheter intervention and imaging of the intimal wall thickening and plaque geometry (20, 21). These methods are either invasive or provide limited information for researchers and clinicians. As the primary goal of the cardiovascular system is to circulate adequate blood flow throughout the body system, patient-specific quantification of blood flow dynamics provides rich information and insights on the interactions between flowing blood on endothelial cells and the rupture of atherosclerotic plaques, and also the blood flow volume supply (22). Fast, non-invasive and accurate 3D to 4D patient-specific blood flow field visualization/quantification (including the WSS) is both informative and of clinical importance for clinicians and researchers, for better understanding, prediction, detection and treatment of CVD.

1.2 3D Blood flow quantification methods and challenges

Experimental flow imaging methods (e.g., MRI or US) and Computational Fluid Dynamics (CFD) are common methods for blood flow quantification, and are introduced below.

1.2.1 MRI and US VFI

3D or 4D Phase Contrast (PC) MRI can acquire the spatiotemporal characteristics of blood flow non-invasively, by imaging a 3D spatial volume in multi-slice mode or volumes over time. The blood flow velocity is determined by a phase change due to moving protons, acquired in two interleaved measurements (23). Then post-processing techniques, e.g., B-spline interpolation, are used to provide optimized quantification of blood flow and vessel wall parameters (24).The functional evaluation of the cardiovascular system using PCMRI, for example volumetric blood flow, has become part of the clinical routine, to aid in the diagnosis and therapeutic management of CVD. However PCMRI suffers from high cost, relatively low temporal and spatial resolution, especially for 3D MRI velocimetry which requires three-directional velocity encoding and full volume scan to cover the flow domain of

interest (24, 25). Furthermore, the spatial resolution of a typical clinical MRI scanner is at the level of 1 mm, which may not be sufficient for accurate WSS estimation (26).

Compared with PCMRI, US Vector Flow Imaging (VFI) provides various experimental methods for faster and more cost effective blood flow velocity quantification, including 1D to 4D Doppler imaging (27-29), Ultrasound Imaging Velocimetry (UIV, also called echo Particle Image Velocimetry or echo-PIV) (26, 30) and Transverse Oscillation(31). Due to its relative simplicity, low cost, higher frame rate, wide accessibility and ever increasing possibilities, the last few decades have seen an increase of US application ranging from *in vivo* cardiovascular flow studies to other fluid mechanics studies, e.g., in ductus venosus (32), ventricles (33) and cerebral blood flow of fetus (34).

However most current US VFI techniques are 1D or 2D, and 3D velocity components and the volumetric flow measurements usually require 3D US systems, which use either a matrix probe or linear probe with mechanical motion technique, e.g., translation or tilting. Compared to US systems using linear probe, 3D US using a matrix probe has the following disadvantages:

- higher cost due to larger number of transducers and cabling in a matrix probe;
- reduced frame rate and larger amount of data to be processed;
- sub-optimal image resolution;
- computationally intensive post-processing given the large amount of data, which can be more than an order of magnitude more than the 2D case. For example, post-processing of 3D cross-beam Doppler requires a least-square solution of a much larger linear system via a pseudo-inverse operation and is more computationally expensive and memory bound than 2D Doppler. Speckle tracking for 3D UIV requires cross-correlation in three directions respectively (35).

It is of great clinical and research interest to develop technologies for fast and accurate reconstruction of 3D full field flows using sparse velocity data generated by a cost-effective 1D or 2D US system. For this purpose three problems have to be solved: 1) the missing out-of-beam or out-of-plane velocity components have to be recovered, as 1D/2D US VFI is limited to acquire 1D along-beam or 2D in-plane velocity components; 2) the missing gaps in the volume with no velocity measurements have to be estimated; 3) US VFI measurements can be noisy and would benefit from being regularized by flow physics, e.g., blood flow must be divergence free.

1.2.2   Patient-specific CFD

On the other hand, 3D patient-specific CFD is the current standard for blood flow quantification. CFD involves solving the flow governing equations, i.e., the Navier Stokes (NS) equations and the continuity equation (36-38), or the Boltzmann equation if the Lattice Boltzmann Method (LBM) is used. Different discretisation algorithms, include Finite Difference Method (39), Finite Volume Method (40), Finite Element Method (41), Spectral Element Method (42), Boundary Element Method, Smooth Particle Hydrodynamics (SPH) (43) etc, and LBM (44) which solvers the Boltzmann equation, are used for blood flow simulation.

Patient specific blood flow CFD requires steps involving: 1) patient-specific vascular system imaging using MRI, CT or US, image segmentation and 3D lumen geometry reconstruction; 2) inlet/outlet boundary condition specification from MRI, US or cardiac catheterization; 3) geometric models discretization using automatic mesh generators; 4) solution of the NS equations iteratively until convergence is achieved; 5) visualization of the resulting physiologic information. If Fluid Structure Interaction (FSI) is considered, an additional solid solver is also required for coupling between the flow and solid simulation. CFD can be used to obtain detailed flow information including WSS and its spatial distribution, pressure drop, stagnation points and recirculation regions.

With the advent of more powerful image acquisition and processing techniques (e.g., deep learning for image segmentation (45), super-resolution (46) and increasing imaging resolution (47)), high-order CFD methods (48), data assimilation(49, 50) and the many-core computing platform (51, 52), the accuracy and speed of patient-specific cardiovascular CFD simulations has increased significantly. Examples include applications of GPU clusters in cardiac flow simulation where one cardiac cycle could be simulated in minutes (53-55), the Immersed Boundary Method (IBM) where Fluid Structure Interaction (FSI) is considered for higher simulation accuracy (56, 57), and minimizing difference between CFD and experimental measurements by data assimilation (58).

However accurate 3D blood flow estimation is still challenging as a result of its complexity originating from:

- pulsatility due to ventricle expansion/contraction. The ventricles serve as a pumping chamber with one-way intake and outflow valves, and the chamber's volume changes during pumping, leading to both pulsatile flow and moving vessel wall;

- complex geometry of vessel lumen due to the presence of bifurcation, curvature, Y-shaped and T-shaped junctions, aneurism, vessels with deployed stent and scaffolds;

- non-Newtonian blood rheology. Blood is a complex mixture of plasma (55%) and several kinds of corpuscles (45%) which include deformable cells, i.e., red blood cells (or erythrocytes), white blood cells (or leukocytes), platelets (or thrombocytes). The viscosity of blood is shear-dependent due to the existence of corpuscle particles, and is termed "non-Newtonian". The influence of non-Newtonian rheology becomes significant when the particles have comparable size with arteries' diameter, especially in capillary vessels. In some disease conditions blood exhibits remarkable non-Newtonian behaviours, which cause a higher Wall Shear Stress (WSS) than the Newtonian models (59, 60);

- complex fluid-structure interaction. The vessel wall is distensible and blood CFD simulation with rigid wall fails to predict essential flow characteristics;

- difficulty in boundary condition specification and vessel lumen segmentation. The vessel is optically opaque and experimental acquisition of patient specific vessel geometry, inlet/outlet velocities requires reliable experimental acquisition, and post-processing including segmentation, surface smoothing, speckle tracking etc;

- high computational cost. Blood flow CFD involves numerically solving a set of unsteady nonlinear partial differential equations, and may contain billions of degrees of freedom if micro-scale flow features are of interest;

- blood being slightly compressible due to the corpuscles which behave like gas bubbles and are somewhat compressible (61).

In general two problems exist as a result of these challenges: 1. Patient specific CFD is computational expensive and slow; 2. Large blood flow CFD error may exist due to uncertainties. Addressing these challenges would require the characterisation of vasculatures' mechanical properties (stiffness, viscoelasticity), CFD mesh refinement and data assimilation, accurate blood flow boundary condition specification, blood rheology and compressibility specification and vessel motion tracking. Various simplifications/assumptions have been made to reduce the uncertainties, including assumptions about the inlet/outlet velocity profile (e.g., Womersley, uniform or parabolic velocity profile), Newtonian rheology, blood being incompressible, static and smooth artery geometry. These assumptions/simplifications have their restrictions and usually cannot deliver satisfactory accuracy given the complexity of patient specific blood flow.

1.3   Research objectives

Due to the challenges in 3D blood flow quantification using US and CFD mentioned in Section 1.2, current 3D blood flow quantification is primarily confined to research purposes. To pave the way towards clinical translation, this thesis aims at developing the following algorithms to make 3D full flow quantification cheaper, faster, more computationally efficient and more accurate compared with the current techniques, thus to address the above-mentioned challenges:

- US augmented Divergence Free Interpolation (DFI). A fast 3D full flow quantification algorithm with 3D distribution of 1D or 2D sparse velocity component measurements input acquired by US VFI;

- A parallel GPU based measurement augmented Lattice Boltzmann Method (LBM) CFD solver. The solver takes advantage of the state-of-the-art many-core computing platform, i.e., a NVIDIA GPU, and integrates the measurements from US VFI into LBM as "internal" boundary condition, to suppress the blood CFD uncertainty and accelerate LBM convergence.

1.4   Previous research

1.4.1.   Previous 3D US blood VFI and reconstruction studies

Previously 3D or 4D US imaging system either uses a linear probe, with mechanical scanning system to obtain a series of 2D US images which are then temporally and spatially registered (62-64), or uses a matrix probe to scan a flow volume directly.

Linear probe with different mechanical scanning systems, including linear scanning, tilt scanning and rotational scanning used to scan a whole 3D volume are introduced in (65). (66) shows 3D flow quantification for liver and portal venous system, by linearly scanning and tilt scanning a whole 3D volume using 2D Doppler. (67) shows carotid artery flow scanning using a motorized linear translation system. (68) shows the results of scanning the 3D whole volume of steady flow with Reynolds numbers between 1300 and 1600 by colour Doppler. For such 3D systems, temporal registration, e.g., using ECG, is necessary. These 3D scanning systems with linear probe require spatially dense acquisition of three velocity components separately, and then conduct interpolation for each velocity component. As a result they suffer from high experimental cost and suboptimal accuracy due to lack of flow physics.

Later 2D matrix probe was developed to measure whole flow volumes of interest, with the advantage of having control over the sound beam in two perpendicular angles. 3D and 4D US imaging using matrix probe was first proposed for fetal heart evaluation (69-72). Then based on the matrix probe technique, different 3D US VFI methods were proposed recently. Provost et al (73, 74) conducted 3D ultrafast Doppler with a 32-by-32 matrix probe, at a frame rate up to thousands of volumes per second. Multi-beam Doppler using a matrix probe used for 4D vascular flow quantification was conducted by Yiu (75). Falahatpisheh et al (76) showed the possibility of 3D *in vivo* cardiac vector flow at a frame rate of 30 imaging volumes per second, using volumetric echocardiographic Particle Imaging Velocimetry (PIV) after the injection of microbubbles. Voorneveld (77) conducted 3D/4D echo-PIV for unsteady flow in ventricle phantoms using a matrix probe. In parallel the transverse oscillation approach developed by Holbek et al (78) was used to estimate the 3D velocity components using an US matrix probe. Such 3D/4D systems, although do not require mechanical probe motion such as tiling and rotating, have the disadvantages of 3D US systems listed in Section 1.2.1, i.e., the compromise of image quality, high computational cost, and larger number of transducers and cabling.

In addition US measurements are noisy and usually have to be smoothed and interpolated/reconstructed to the full flow domain. Previous 3D VFI flow reconstruction relies on densely scanning three velocity components in a volume of interest and then conduct interpolation for each direction, e.g., using cubic spline interpolation (79). Later on several interpolation methods (80, 81) were proposed, which smooth/reconstruct the 3D flow in accordance to physical laws but are only eligible for 3D velocity components input and are computational intensive. In addition the reconstruction accuracy is highly dependent on the selection of hyperparameters of the interpolation system. Another study (82) recovers 2D flow from 1D velocity components acquired by Doppler imaging, based on the assumption that blood flow is 2D planar and out-of-plane flow velocity is ignored. However this method has a relative large error level (<20%) and the reconstruction is slow when a large amount of US input data is used as input.

Divergence Free Interpolation (DFI) using Radial Basis Functions (RBF) is a mesh free flow reconstruction method, which has a low computational cost and high accuracy as it reconstructs the flow field from real measurements. It assumes blood flow is incompressible and complies the divergence free condition, i.e., $\nabla \cdot \vec{v} = 0$. As an alternative flow quantification algorithm to CFD, it takes sparse 3D velocity data in a 3D volume as input,

and output 3D velocity elsewhere in the flow domain. DFI does not require fluid properties such as viscosity, or even the full vessel geometry, to be known. It is therefore more reliable than CFD in cases when unphysical assumptions/simplifications are made or large uncertainties exist in CFD input. DFI was first proposed by Narcowich et al. (83), and was then proven accurate to obtain full field flow velocity by Lowitzsch (84, 85). Chan et al. recently proposed a DFI method from multiple directional PIV measurements in optical microscopy but did not consider regularization (86). All previous DFI studies cannot recover the lost VFI velocity component, i.e., 3D velocities input is necessary for 3D reconstruction, while most current US VFI techniques are 1D or 2D. This gap between 3D DFI and current US measurement technologies has to be filled.

1.4.2. Previous 2D US UIV study

This section gives a brief review on US UIV, which is the default US VFI technique in this thesis. Compared with pulsed Doppler techniques which have angle dependent accuracy and depth dependent maximum velocity, 2D UIV tracks the motion of US speckle signals scattered from within blood between consecutive B-mode images, based on the hypothesis that local echogenicity is sufficiently preserved from one frame to the next. The speckle tracking technique of UIV using cross-correlation was reported by Bohs et al. in (87, 88). Microbubble is usually used as contrast agents to increase backscattered ultrasound signal and thus accuracy. Being similar to optical PIV, UIV identify the direction and distance for maximum correlation of speckle features by means of optical flow (89, 90) or block-matching techniques (91, 92). Advanced PIV processing algorithms including an iterative scheme, window offsetting, normalized cross-correlation in the Fourier domain (89) and peak fitting with sub-pixel accuracy using parabolic or Gaussian curve (93, 94) are usually employed to increase the spatial resolution of velocity measurements (95). Given the potential errors and noise in velocity estimation, robust regularization is essential to obtain reasonable and smooth velocity fields against large noise (96-98). 3D UIV based on 3D cross-correlation, using a matrix probe, is also studied in (99, 100) but suffers from sub-optimal spatial resolution and large amount of data to be processed. Cross-correlation is computationally intensive due to the large number of frames, GPU processing is a popular and cheap approach adopted for faster post-processing (26, 30).

UIV has been applied for feasibility study in non-transparent curved tube (79), and in a number of human vascular flow studies, for example, blood flow quantification in superficial vein (101), in small animals and phantoms (102), in abdominal aorta (103), in carotid (104,

105). UIV with contrast agents is also utilized to quantify complex flows, e.g., vortices and flow with large gradients, especially in the left ventricle (106-111) during diastole.

1.4.3.   Previous physiological CFD study

CFD has been applied intensively *in vivo* for hemodynamics study (112, 113) where MRI (36), US (114) or interventional catheter (115) are usually used to acquire temporally changing boundary conditions and the geometry of the vessel. Different CFD algorithms are applied for blood flow simulation, including the currently most popular Finite Volume Method (36, 116), the Finite Element Method (117, 118), the Lattice Boltzmann Method (106)(119) and the Smooth Particle Hydrodynamics (120, 121) etc. Fast CFD simulations require high-speed hardware with large memory, and clusters of CPUs (122-124) and NVIDIA GPUs (125-128) are usually used for parallel computation to accelerate CFD.

CFD has been used to establish the link between haemodynamic disturbance and atherogenesis (129), study coronary artery diseases (130-132), aortic/aortic/cerebral aneurysm, pulmonary hypertension, heart failure(133, 134), and provides information for WSS mapping, valve prosthetics and stent design. (135, 136) used CFD for aorta blood flow simulation, and (137, 138) predicted the atheroma location in regions where WSS is low using CFD. (138) and (139) simulated blood flows in stenosed carotid bifurcation models to study the effects of geometrical parameters. (140) simulated a vessel with eccentric stenosis, where boundary conditions are acquired by Doppler imaging. Their results show that the increase in the severity of stenosis at throat region has a dramatic influence on local flow velocity and WSS. (141) investigated the WSS, velocity and pressure distribution near carotid artery plaques, with different degrees of stenosis. (142, 143) simulated blood flow considering non-Newtonian rheological effects. Others (144, 145) conducted patient-specific CFD in diseased arteries, and the results show unique flow characteristics in each simulation.

LBM has been developed in the last several decades as an alternative CFD algorithm. Instead of solving the continuity and the Navier Stokes equations, LBM solves the discretized Boltzmann equation which describes the motion and interaction of virtual particles. LBM has attracted wide interest in blood flow simulation (146-151), with elastic and moving boundaries (152-155), complex boundaries (156), pulsatility (147, 157-160) and non-Newtonian rheology (161, 162), and also blood flow through a mechanical heart valve (146, 147), in the stented aneurysm (148, 151), in microvascular bifurcations (160, 163), FSI with moving boundaries (164), prediction of the thrombus formation. As a mesoscale description

of virtual particle probabilities quantified by the Boltzmann equation, LBM recovers the Navier-Stokes and the continuity equations which describes flow motion in the macroscopic scale (165). LBM is ideally suited for parallel computation because: 1) the update of flow information for each virtual particle only requires information from adjacent grid points, and virtual particles move into adjacent nodes during each time-step, resulting in low memory latency; 2) LBM has a simple kernel function.

1.5   Contributions

The thesis advances the state-of-the-art in 3D physiological flow simulation and ultrasound flow reconstruction, in the following aspects:

1. This thesis proposes a novel Ultrasound Augmented DFI (UADFI) algorithm to reconstruct 3D full flow from 3D distribution of 1D or 2D ultrasound velocity measurements, using an improved DFI. The resulting linear system is ill-posed and to obtain a stable solution, truncated Singular Value Decomposition (TSVD) is used to find a low rank approximate solution which is robust to input error (including measurement/registration error etc). Three different methods are used to parameterize the regularization parameter of the low rank approximate solution, and the results show the L-curve method has highest accuracy.

2. Further optimizations of UADFI are conducted in terms of accuracy and computational efficiency. First the thesis studied the shape parameter of the RBF to optimize the linear basis space of UADFI, to minimize the influence of noise and numerical oscillation near the interpolation boundary (known as the Runge Phenomenon). Then rank analysis of the UADFI system's Krylov subspace is conducted, and based on the analysis we prove the Generalized Minimal Residual method (GMRES) is valid for the singular system with conditions about the maximum iteration steps. Later on we increase the sparsity of the linear system by a series of matrix transformations, and obtain a more computational efficient solution using GMRES.

2. To reduce the CFD error caused by input uncertainties and numerical models, and accelerate LBM convergence, we integrate ultrasound velocity measurements to calculate the equilibrium probability distribution of LBM virtual particles, so that measurements are used as "internal" boundary conditions. This study is based on the assumption that internal flow measurements and inlet/outlet are acquired by the same US measurement, and have similar fidelity level. The results show significant speed-up and error suppression effects of LBM by measurements as augmentation data. Then we extend the LBM solver to non-Newtonian flow

simulation and the results show high accuracy validated with commercial software. The two solvers are developed based on a NVIDIA GPU.

1.6   Thesis overview

Chapter 2 introduces two common US VFI experimental methods, i.e., Doppler imaging and US speckle tracking (also known as echo-PIV or UIV), and their corresponding projection matrices. Then the Radial Basis Function (RBF) and the fundamentals of 3D DFI using RBF are introduced. We then propose the UADFI algorithm and its solution obtained by Truncated Singular Value Decomposition (TSVD), where different regularization methods are compared to find the optimal regularization parameter. The algorithm is validated in both *in silico* and *in vitro,* using CFD simulation obtained with commercial package as reference The results show the L-curve method has highest accuracy.

Chapter 3 first studies the influence of RBF shape parameter on UADFI accuracy and robustness against input noise, and adopted the hold-out validation method to optimize the RBF shape parameter, for the optimization of solution space spanned by the divergence free basis. As the UADFI system is under-determined (singular) and GMRES may break down, Chapter 3 then gives the condition when GMRES is breakdown free and valid, and its regularization for the suppression of the input noise. Later the Krylov subspace of GMRES is constructed using the householder transformation due to its high computational robustness, before GMRES is applied to solve the UADFI system. The results show up to hundreds of times speedup and higher accuracy (the reference is CFD obtained by commercial CFD package) compared with Chapter 2.

Chapter 4 first presents the fundamentals of the Lattice Boltzmann method. Then a GPU version LBM code was developed and to augment LBM by integrating US VFI measurements into the solver as internal "Boundary Conditions". By assuming the internal measurements have the similar fidelity level with CFD inlet/outlet, the algorithm is validated by two *in silico* benchmark problems where the results show significant uncertainty suppression and convergence acceleration. Then the experimental feasibility is tested for steady carotid bifurcation phantom flow *in vitro*. To evaluate the effectiveness of the augmentation method, we conduct LBM simulation with or without augmentation using perturbed boundary conditions, and evaluate the accuracy using LBM simulations (without augmentation) using error free boundary conditions as reference. In Appendix A2 the GPU

version non-Newtonian LBM solver is built, followed by its *in silico* validation in steady straight vessel flow simulation.

Chapter 5 summaries the thesis, lists the main conclusions, limitations and potential future work, and states the possible clinical applications of the proposed algorithms.

Chapter 2 US VFI measurements and VFI Augmented DFI reconstruction

2.1. Introduction

Previous 3D DFI method reconstructs the full field 3D velocities using sparse 3D velocity measurements, however most current US systems are 2D or 1D. This chapter aims at extending DFI to be capable of full field 3D velocities reconstruction using 3D distribution of 1D or 2D sparse velocity measurements. This chapter starts with an introduction about two most common US VFI experimental methods, including the Doppler method and UIV, and their corresponding velocity projection matrices. Then a new algorithm of 3D DFI reconstruction using sparse 2D VFI velocity input is proposed. The resulting system is singular and to find the regularized reconstruction solution for error robustness, the Truncated Singular Value Decomposition (TSVD) is used for regularization, where Generalized Cross Validation (GCV), Normalized Cumulative Periodogram (NCP) and the L-curve method are compared to find the optimal regularization parameter in the end.

2.2. 2D and 1D US VFI methods

Doppler method and UIV are introduced in this section. Other VFI methods, e.g., the transverse oscillation (166-169), directional beamforming (170, 171), spectral based methods (172, 173) and colour Doppler based methods (174, 175) are not introduced.

2.2.1. Doppler Imaging

Conventional US Doppler system only measures the 1D along-beam velocity component, and its accuracy depends on the angle between flow and US beam (176, 177). In addition the maximum velocity that can be acquired by the pulsed Doppler is depth dependent. The US frequency change returned from a moving flow scatterer is (176):

$$\mathbf{f_D} = \frac{2|\vec{\mathbf{v}}|\cos\theta}{\mathbf{c}}\mathbf{f_0} = \frac{2\mathbf{v_z}}{\lambda_0} \tag{2.1}$$

where f_0 and f_D are the emitted and the Doppler frequencies, λ_0 is US wavelength, c and $|\vec{v}|$ are sound speed and blood velocity respectively. θ is the beam flow angle and should be smaller than 60° for accuracy. Fig. 2-1 illustrates the US beam and flowing scatterer for conventional 1D Doppler imaging.
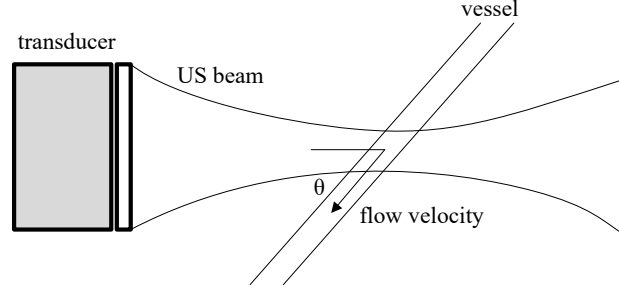
Figure 2-1 Doppler 1D velocity measurement

Later cross-beam and multi-beam Doppler methods have been proposed to acquire 2D and 3D velocity. Cross-beam Doppler (178-180) insonifies one scatterer twice with two different pairs of transmitting/receiving beams, and for each transmitting/receiving beam pair an equation containing the scatterer velocity magnitude $|\vec{v}|$ and angle θ can be obtained

$$\frac{f_D}{f_0}c = |\vec{v}| \cos(\theta_T - \theta) + |\vec{v}|\cos(\theta_R - \theta) \qquad (2.2)$$

where θ_T and θ_R are the beam transmitting and receiving angle. 2D cross-beam Doppler imaging using a linear US probe is shown in Fig. 2-2. Substituting $v_x = |\vec{v}|\cos\theta$ and $v_y = |\vec{v}|\sin\theta$ into Eq. 2.2 we have:

$$\frac{f_D}{f_0}c = v_x(\cos\theta_T + \cos\theta_R) + v_y(\sin\theta_T + \sin\theta_R) \qquad (2.3)$$

Eq. 2.3 contains two unknowns and can be closed with two transmitting/receiving beam pairs insonifying the same scatterer. Similarly three velocity components can be obtained using a matrix probe transmitting/receiving three beam pairs insonifying the same scatterer (181-183).
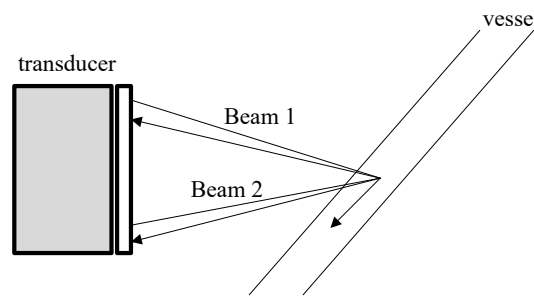
Figure 2-2 Doppler 2D velocity measurement using cross beam method

Eq. 2.3 is over-determined if multi-beam method is used and the number of equations is larger than unknowns, e.g., more than two transmitting/receiving beam pairs are used to acquire the 2D velocity of a scatterer, or more than three beam pairs are used to insonify the 3D velocity of a scatterer. The over-determined system can be solved via a least-square

estimation with proper regularization to suppress the influence of noise, but this is not introduced here.

2.2.2. UIV

US speckle tracking, also termed Ultrasound Imaging Velocimetry (UIV), echo-PIV, is a method originating from the laser speckle velocimetry and PIV. UIV tracks speckle motion between consecutive B-mode frames or directly from US Radio-frequency (RF) signal, usually with microbubbles for contrast enhancement or by tracking blood speckles directly. Velocities are estimated by tracking distance of tracers travelled over time. Speckle tracking is usually implemented by a block-matching technique in different interrogation scales for refinement (184, 185) to track different scales of flow features, and can be implemented in Fourier domain (186) or in parallel with GPU (30) for computational efficiency. Speckle tracking usually comprises four steps: 1) subdivision of two successive B-mode images into small interrogation windows; 2) conduct normalized cross correlation between window pairs, usually in the Fourier domain; 3) peak fitting and displacement estimation; 4) refine the interrogation window size to increase spatial resolution. The displacement accuracy can be sub-pixel size and the resulting velocities are usually fitted for smoothing. In this thesis, UIV is the default *in vitro* US VFI method.

2.2.3. UIV experiment for DFI

In this section, we demonstrate our algorithm for 3D full field flow reconstruction with 3D distribution of 2D velocities acquired by UIV, using a linear US probe moved by a step controller.

A High Frame Rate (HFR) plane wave imaging system was developed for 2D ultrasound speckle tracking, using decafluorobutane microbubbles as contrast agents. The decafluorobutane microbubbles were prepared as described in (187), and were diluted in water to a concentration of 2×10^5 microbubbles/mL with an average size of ~1μm. Water containing microbubbles was driven by gravity from an upstream tank through an entrance tube to the phantom at a constant flow rate. A L12-3v linear array probe connected to a Vantage 128 platform (Verasonics, Redmond, WA, USA) was used for UIV acquisition. Pulse inverse scheme was used to distinguish microbubbles from the background tissues, with 5 transmitted compounding imaging angles between -9° and 9°, at a frame rate of 1000 fps.

In this study the ultrafast UIV system was used to track steady flow in a carotid bifurcation tissue-mimicking phantom or a straight vessel flow phantom. The entrance of the tube could be controlled by a restrictor thereby permitting adjustment of flow rate, and the location of the restrictor is ~0.5 m away from the inlet of the scanned flow phantom, to ensure that the inlet velocity profile of the measured flow field is fully developed. The phantom is placed in a tank filled with water, and the US transducer is mounted over the phantom. An acoustic absorber is placed at the bottom of the tank to prevent acoustic reflection of sound from the walls of the tank. The received RF data from the Verasonics system is then transferred to a local workstation, and then beamformed by a DAS (delay and sum) method to produce B mode images. A highly accurate (error<10% for *in vivo* flow quantification) autocorrelation technique running in GPU developed by Leow et al (184) was used for 2D in-plane flow velocity quantification, from the consecutive B-mode images. The 2D experimental setup and acquisition procedures are described in (184). It takes a matter of microseconds to acquire one image and several seconds for UIV post processing using GPU, i.e., in-plane velocity calculation. The workflow of the UIV acquisition and postprocessing is shown in Fig. 2-3 (b).

(a)
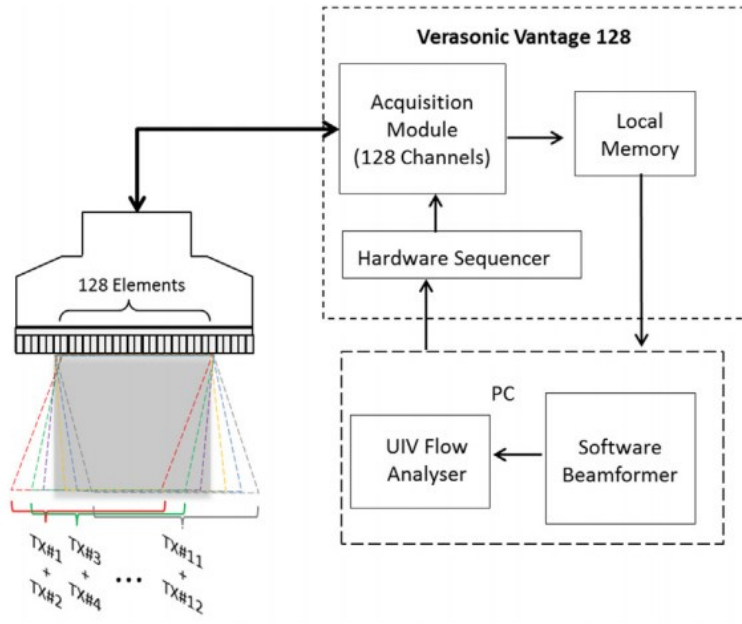
Figure 2-3 (a) Verasonics US system; (b)UIV hardware and data processing workflow (30)

For full flow interpolation in 3D space, 3D distributions of velocities are necessary. Thus in UADFI, multi-angle 2D velocimetry measurements are acquired (or simulated). In this study, two distinct acquisition angles are required since these provide the projections of the real 3D flow field onto two plane directions which contain independent flow information to allow for full field reconstruction. The angles between the imaging plane (probe plane) and probe motion direction are denoted as θ_1 and θ_2 for two imaging angles respectively. In practice, ultrasound scanning is achieved in the following manner: after scanning at discrete planes separated by spacing along Z axis (defined along the direction of the probe motion, see Fig. 2-4a for the first angle (θ_1) from Z_1 to Z_2, the imaging probe was rotated to θ_2 and scan the same flow domain along –Z direction until the probe reaches Z_1. Acquired B-mode image is shown in Fig. 2-4b. The motion of the probe is controlled by a programmed stage controller and its step size is 5 mm (spacing=5 mm). Acquisition angles and spacing are illustrated in Fig. 2-4a. 2D velocity vectors are then registered according to acquisition parameters and segmented by a level set method (188), as shown in Fig. 2-4c. θ_1 and θ_2 can be any value, but to increase the independence of acquired data the two angles are chosen to be perpendicular. In this study we choose $\theta_1=45°$ and $\theta_2=135°$ for simplicity.
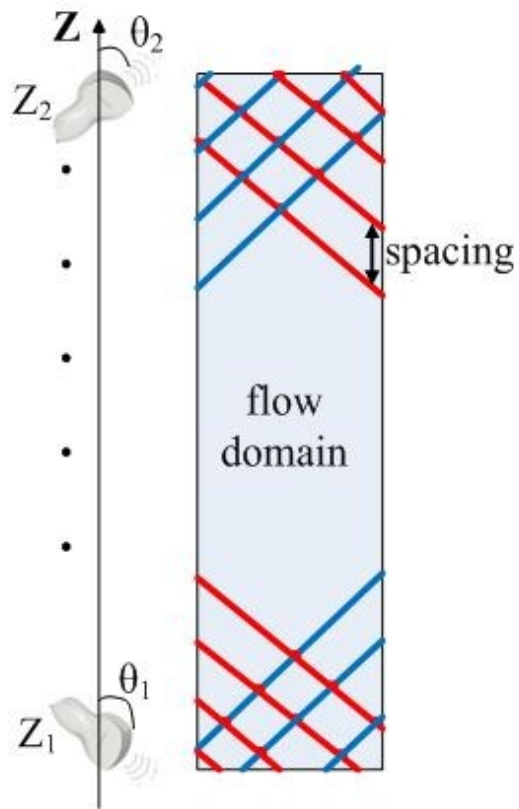
(a)
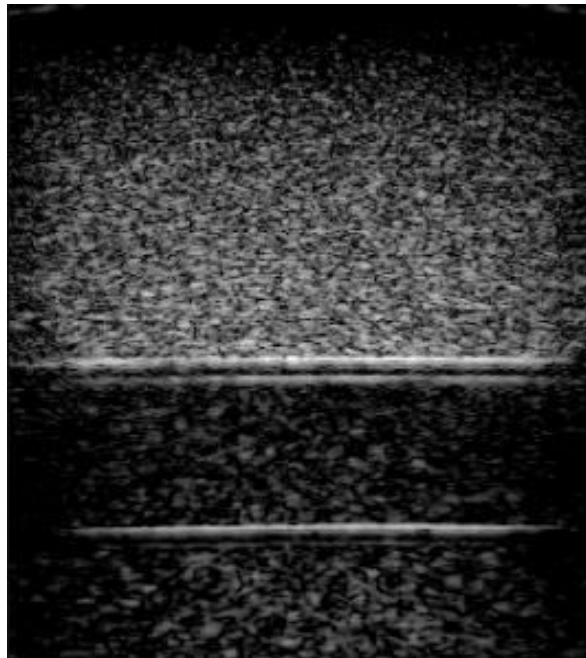
(b)
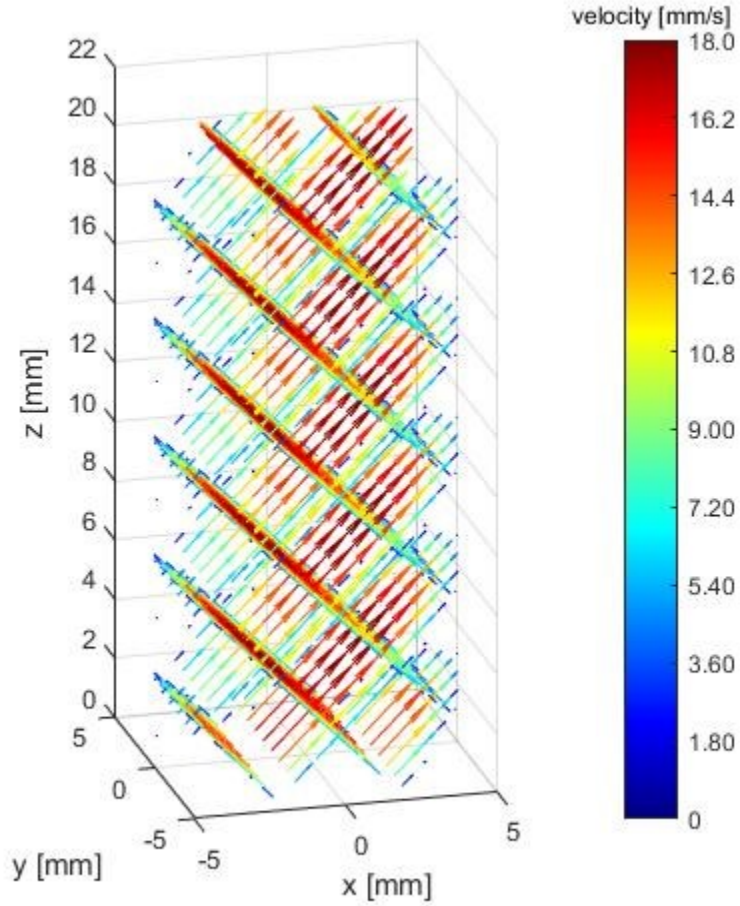
(c)

Figure 2-4 (a) Schematic of straight cylindrical vessel (top view) along the probe motion direction, with blue and red imaging planes demarcating the two independent scanning directions; (b) one B-mode image acquired in a straight vessel; (c) illustration of 2D velocity acquired in a straight vessel by UIV

2.3. Divergence Free Interpolation Using the Radial Basis Function

Radial Basis Functions (RBFs, also termed RBF kernels) include a family of real-valued axisymmetric functions, and their value depends only on the radial distance (usually the Euclidean distance) from the RBF centre (also termed RBF origin). RBF can be denoted by $\varphi(\vec{x}) = \varphi(\|\vec{x} - \vec{c}\|)$ where $\vec{c}$ is the interpolation centre, $\vec{x}, \vec{c} \in R^3$ for 3D interpolation. Common RBFs include:

$$\text{Gaussian: } \boldsymbol{\varphi}(\vec{\mathbf{r}}) = \exp(-\boldsymbol{\varepsilon}\vec{\mathbf{r}}^2) \tag{2.4}$$

$$\text{Multiquadric: } \boldsymbol{\varphi}(\vec{\mathbf{r}}) = \sqrt{1 + (\boldsymbol{\varepsilon}\vec{\mathbf{r}}^2)} \tag{2.5}$$

$$\text{Inverse Multiquadric: } \boldsymbol{\varphi}(\vec{\mathbf{r}}) = \frac{1}{\sqrt{1 + (\boldsymbol{\varepsilon}\vec{\mathbf{r}}^2)}} \tag{2.6}$$

$$\text{Thin Plate Spline: } \boldsymbol{\varphi}(\vec{\mathbf{r}}) = \vec{\mathbf{r}}^2 \ln(|\vec{\mathbf{r}}|) \tag{2.7}$$

where $\vec{r} = \vec{x} - \vec{c}$, ε in Eqs. 2.4-2.6 is referred as the shape parameter, and is related with spatial width of the kernel function. Shape parameter has a significant influence on RBF accuracy and numerical stability, and will be investigated in Chapter 3. For simplicity, a choice is to set ε close to the inverse of the distance between two adjacent centre points (the in-plane US resolution), if a nearly uniform input scheme is used. For approximation/interpolation a function can be written as a weighted summation of RBFs centred at different spatial locations:

$$\mathbf{y}(\vec{\mathbf{x}}) = \sum_{i=1}^{N} \mathbf{w_i}\boldsymbol{\varphi}(\|\vec{\mathbf{x}} - \vec{\mathbf{c}_i}\|) \tag{2.8}$$

The above equation is linear in the unknown weighting coefficient w_i, and with N equations the system is well-determined and w_i can be solved by matrix inversion, otherwise a least square solution of w_i can be found. Previous 3D DFI techniques require 3D sparse velocity input for flow reconstruction, in which the matrix-valued divergence free RBF, $\emptyset(\vec{r})$, is defined by:

$$\boldsymbol{\emptyset}(\vec{\mathbf{r}}) = (-\boldsymbol{\Delta}\mathbf{I} + \boldsymbol{\nabla}\boldsymbol{\nabla}^{\mathbf{T}})\boldsymbol{\varphi}(\vec{\mathbf{r}}) \tag{2.9}$$

where $\varphi(\vec{r})$ is the scalar valued RBF (see examples of such RBF in Eqs. 2.4~2.7) and I is the identity matrix. Using Gaussian as the RBF kernel Eq. 2.9 can be expanded as:

$$\boldsymbol{\emptyset}(\vec{\mathbf{x}}_\mathbf{j}, \vec{\mathbf{c}}_\mathbf{i}) = \{(2(\mathbf{d} - 1)\boldsymbol{\varepsilon} - 4\boldsymbol{\varepsilon}^2\|\vec{\mathbf{r}}\|^2)\mathbf{I} + 4\boldsymbol{\varepsilon}^2\vec{\mathbf{r}}\vec{\mathbf{r}}^{\mathbf{T}}\}\exp(-\boldsymbol{\varepsilon}\vec{\mathbf{r}}^2) \tag{2.10}$$

where ε is the shape parameter of Gaussian and equal to the reciprocal of the variance of the Gaussian function, $\mathbf{d}$ is the spatial dimension. $\vec{x}, \vec{c} \in R^3$ and $\emptyset(\vec{x}_j, \vec{c}_i)$ is a 3-by-3 symmetric matrix. $\vec{x}_j$ is the coordinate of the jth velocity vector and $\vec{c}_i$ is the coordinate of the ith interpolation center point. For 3D reconstruction Eq. 2.10 becomes:

$$\boldsymbol{\emptyset}(\vec{\mathbf{x}}_\mathbf{j}, \vec{\mathbf{c}}_\mathbf{i}) = \{(4\boldsymbol{\varepsilon} - 4\boldsymbol{\varepsilon}^2\|\vec{\mathbf{r}}\|^2)\mathbf{I} + 4\boldsymbol{\varepsilon}^2\vec{\mathbf{r}}\vec{\mathbf{r}}^{\mathbf{T}}\}\exp(-\boldsymbol{\varepsilon}\vec{\mathbf{r}}^2) \tag{2.11}$$

For simplicity the spatial coordinates of the point cloud of centre points coincide with those of measurement data points, and the velocity interpolation scheme has the following form:

$$\vec{v}(\vec{x}_j) = \sum_{i=1}^{m} \emptyset(\vec{x}_j, \vec{c}_i)\lambda_i \qquad (2.12)$$

where $\vec{v}(\vec{x}_j) = [\vec{v}_{j1}, \cdots, \vec{v}_{jd}]^T$, m is the number of interpolation centres, $\lambda_i = [\lambda_{i1}, \cdots, \lambda_{id}]^T$ is a vector of weight coefficients for the RBFs centred at $\vec{c}_i$. Each column of $\emptyset(\vec{x}_j, \vec{c}_i)$ is divergence free via Eq.2.11 and the interpolated velocity $\vec{v}(\vec{x}_j)$ is a linear combination of divergence free column vectors (termed linear basis) and is thus divergence free (83). We rewrite Eq.2.12 in a compact matrix form:

$$\underbrace{\begin{bmatrix} \vec{v}(\vec{x}_1) \\ \vdots \\ \vec{v}(\vec{x}_m) \end{bmatrix}}_{\vec{v}} = \underbrace{\begin{bmatrix} \emptyset(\vec{x}_1, \vec{c}_1) & \cdots & \emptyset(\vec{x}_1, \vec{c}_m) \\ \vdots & \ddots & \vdots \\ \emptyset(\vec{x}_m, \vec{c}_1) & \cdots & \emptyset(\vec{x}_m, \vec{c}_m) \end{bmatrix}}_{G} * \underbrace{\begin{bmatrix} \lambda_1 \\ \vdots \\ \lambda_m \end{bmatrix}}_{\lambda} \qquad (2.13)$$

λ and $\vec{v}$ are sequentially stacked vectors of weighting coefficients for all centres and velocities for all input points respectively. The interpolation matrix G is real, square, positive definite, and symmetric if two conditions are satisfied: 1) the data points, denoted by $\{\vec{x}_i\}$, coincide with centre points $\{\vec{c}_i\}$; 2) the shape parameter ε is uniform for all centres. If globally supported RBFs, e.g., Gaussian, Multiquadric or Inverse Multiquadric, are chosen as the kernel, G is dense. On the contrary, if compactly supported RBFs, e.g., compact Thin Plate Spline or Wendland's functions, are chosen as the kernel G is sparse.

The range space (also referred as the column space) of G, denoted by R(G), is a weighted linear combination of the divergence free basis. λ is obtained by solving the linear system Eq. 2.13, and then velocity at any spatial location in the flow domain can be interpolated by Eq. 2.13.

As previous 3D DFI studies take sparse 3D velocity measurements as input, below we derive the new algorithm of Ultrasound Augmented DFI based on 1D or 2D US VFI velocity input.

2.4. Projection matrix

This section introduces three projection matrices for US imaging. Although this thesis focuses on 3D reconstruction, we first start with a 2D projection matrix which is similar to 3D projection matrix but simpler to start. The 2D projection matrix can be used for 2D flow reconstruction from 1D measurements.

A) projection of a 2D velocity along a 2D vector

Assuming for single beam Doppler using a 1D probe (Fig. 2-1), we acquire the along-beam velocity component, i.e., the velocity component of a 2D velocity $\vec{v}$ along the US beam direction denoted by $\vec{B} = (B_x, B_y)$. The projection can be written in a matrix form as:

$$\vec{v}_{\vec{B}} = P_{\vec{B}} * \vec{v} \qquad (2.14)$$

where $P_{\vec{B}}$ is the projection matrix defined as

$$P_{\vec{B}} = \frac{1}{\|\vec{B}\|_2} * \begin{bmatrix} B_x B_x & B_x B_y \\ B_x B_y & B_y B_y \end{bmatrix} \qquad (2.15)$$

$\vec{v}$ and $\vec{B}$ are in the same imaging plane. Left multiplying $P_{\vec{B}}$ to $\vec{v}$ has the effects of setting the velocity component which is normal to $\vec{B}$ equal to 0, while keeping the velocity component along $\vec{B}$ unchanged.

B) projection of 3D velocity onto a plane

We assume for UIV using a linear probe, and cross-beam or multi-beam Doppler using a linear probe. We acquire the in-plane velocity component, i.e., the velocity component of a 3D velocity vector $\vec{v}$ onto the imaging plane whose normal is denoted by a vector $\vec{B} = (B_x, B_y, B_z)$. The projection can be written as:

$$P_{\vec{B}} = \frac{1}{\|\vec{B}\|_2} * \begin{bmatrix} B_y{}^2 + B_z{}^2 & -B_x * B_y & -B_x * B_z \\ -B_x * B_y & B_x{}^2 + B_z{}^2 & -B_y * B_z \\ -B_x * B_z & -B_y * B_z & B_x{}^2 + B_y{}^2 \end{bmatrix} \qquad (2.16)$$

C) projection of 3D velocity along a 3D vector

Assuming for single beam Doppler using a matrix probe, we acquire the 3D distribution of along-beam velocity component, i.e., velocity component of a 3D velocity $\vec{v}$ along the US beam direction denoted by a vector $\vec{B} = (B_x, B_y, B_z)$. $\vec{B}$ points from the virtual Doppler focusing point to the flowing scatterer. The projection matrix $P_{\vec{B}}$ can be written in Eq. 2.17:

$$P_{\vec{B}} = \frac{1}{\|\vec{B}\|_2} * \begin{bmatrix} B_x B_x & B_x B_y & B_x B_z \\ B_x B_y & B_y B_y & B_y B_z \\ B_x B_z & B_y B_z & B_z B_z \end{bmatrix} \qquad (2.17)$$

It has to be noted that the projection is denoted by $P_{\vec{B}}$, which is a linear transformation matrix that transforms a vector space to itself, and can be written as $P_{\vec{B}}{}^2 = P_{\vec{B}}$. That is, whenever $P_{\vec{B}}$ is applied twice to any value it gives the same results as if it is applied once (termed idempotent). We conduct eigenvalue decomposition to the projection matrix $P_{\vec{B}}$, and find that

$P_{\vec{B}}$ is singular with only 0 or 1 as its eigenvalues: A) for projection matrix of 2D velocity along a 2D vector, $P_{\vec{B}}$ has one eigenvalue 1 and one eigenvalue 0, i.e., $\text{rank}(P_{\vec{B}}) = 1$; B) for projection of 3D velocity onto a plane, $P_{\vec{B}}$ has two equal eigenvalues 1 and one eigenvalue 0, i.e., $\text{rank}(P_{\vec{B}}) = 2$; C) for 3D velocity along a 3D vector, $P_{\vec{B}}$ has one eigenvalue 1 and two equal eigenvalues 0, i.e., $\text{rank}(P_{\vec{B}}) = 1$. The rank analysis of $P_{\vec{B}}$ is of importance for the Krylov subspace analysis in Chapter 3.

2.5.   Ultrasound Augmented 3D DFI

2.5.1   Combined projection matrix

In order to force compliance of mass conservation it is essential to consider the non-slip boundary condition at the vessel wall. We track the vessel motion from successive B-mode images, and obtain the full 3D velocity of such wall velocities. Assume a total number of m ($m = m_1 + m_2$) velocities are used as input, from which m_1 are from US VFI and m_2 are from US wall motion tracking. As introduced in Section 2.3, 3D distribution of 2D in-plane velocity component or 1D along-beam velocity components can be acquired by stepping or rotating a linear US probe or directly by a matrix probe. We construct the combined projection matrix R:

$$R = \begin{bmatrix} I & & & & & \\ & \ddots & & & & \\ & & I & & & \\ & & & P_1 & & \\ & & & & \ddots & \\ & & & & & P_{m_1} \end{bmatrix} \qquad (2.18)$$

where $P_1 \sim P_{m_1}$ are the corresponding projection matrix (see Section 2.2.3) for each projected velocity input from US VFI. Wall velocities do not involve projection and correspond to the identity matrix I (here I is a 3-by-3 identity matrix). In this section the VFI method is UIV, and left multiplication the combined projection matrix with $\vec{v}$ in Eq. 2.13 is equivalent to setting the corresponding out-of-plane velocity components equal to 0, while does not change the vessel wall velocities which correspond to I.

2.5.2   UADFI

We left multiply the combined projection matrix R to both sides of Eq. 2.13 and the resulting system is:

$$\mathbf{A\lambda = b} \qquad\qquad (2.19)$$

where $A = R * G$, and $b = R * \vec{v}$. b is the corresponding projection of full 3D velocities on US imaging beams or imaging planes, i.e., VFI measurements, and wall velocity remains unchanged. It has to be noted that those VFI measurements have to be registered to the flow domain, with a rigid body transformation if the US probe translates or rotates. Moreover the velocity vectors have to be rotated according to the angle of the probe, and each velocity measurement has three components in the Cartesian coordinate although it is acquired by 2D or 1D US, as the imaging plane/beam has a rotation/tilting angle.

2.5.3 Rank analysis of UADFI

Below we conduct rank analysis of UADFI, and take 3D reconstruction using 2D UIV input as an example. As shown in Section 2.5.1 R is a block diagonal matrix and its diagonal blocks $P_{\vec{B}}$ are singular. R is thus singular and left multiplication with the projection matrix $P_{\vec{B}}$ results in a planar projection of a 3D velocity. $P_{\vec{B}}$ is a symmetric 3-by-3 matrix whose rank is 2. We conduct eigenvalue decomposition for $P_{\vec{B}}$:

$$\mathbf{P_{\vec{B}} = Q_P * D_P * Q_P} \qquad\qquad (2.20)$$

where Q_P is the square 3-by-3 matrix whose ith column is the ith eigenvector of $\mathbf{P_{\vec{B}}}$, and

$$\mathbf{D_P} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \qquad\qquad (2.21)$$

The eigenvalue decomposition for R is:

$$\mathbf{R = Q * D * Q} \qquad\qquad (2.22)$$

where

$$\mathbf{Q} = \begin{bmatrix} \mathbf{I_{3m_2}} & & & \\ & \mathbf{Q_P} & & \\ & & \ddots & \\ & & & \mathbf{Q_P} \end{bmatrix}, \mathbf{D} = \begin{bmatrix} \mathbf{I_{3m_2}} & & & \\ & \mathbf{D_P} & & \\ & & \ddots & \\ & & & \mathbf{D_P} \end{bmatrix} \qquad (2.23)$$

I_{3m_2} is a $3m_2$-by-$3m_2$ identity matrix. G is full rank and $\text{rank}(G) = 3m_1 + 3m_2$. It can be proven $\text{rank}(R) = 2m_1 + 3m_2$ by analyzing its eigenvalues. Below we prove $\text{rank}(A) = 2m_1 + 3m_2$:

1). $\text{rank}(A) = \text{rank}(R * G) \leq \min\big(\text{rank}(R), \text{rank}(G)\big) = 2m_1 + 3m_2$;

2). according to Sylvester's rank inequality $\text{rank}(R * G) \geq \text{rank}(R) + \text{rank}(G) - 3m = 2m_1 + 3m_2$.

It shows that due to multiplication with R, the number of independent equations decreases from 3m to $2m_1 + 3m_2$ and matrix A in Eq. 2.19 is under-determined (singular system). The condition number of singular matrix is infinite. This means, even a very small change in b will cause an infinite change in λ when inversion of Eq. 2.19 is conducted. b is from US VFI measurements and it deviates from the ideal divergence free physics due to three reasons: 1) in real medical imaging applications US VFI error contains measurement noise and post-processing noise/error; b) blood flow is slightly compressible; c) floating point rounding-off error.

Given such deviations and the ill-posedness of the problem, a mathematically accurate solution of λ may be physically pointless due to overfitting noise. For example, assuming a relative deviation of measurements from the ideal divergence free physics is 15%, a solution to fit the model with the data with a residual smaller than 15% is not trustworthy physically. To solve the ill-posed system of Eq. 2.19 and obtain a stable and physically meaningful solution of λ, regularization should be used and below we introduce a Truncated Singular Value Decomposition (TSVD) solution to Eq. 2.19, and show its robustness by adding artificial noise to the input, i.e., vector b.

2.6. Regularization method

Eq. 2.19 is an under-determined system, with velocity vector b contaminated by noise. The solution of Eq. 2.19 becomes an error minimization least square problem with regularization. Different regularization methods can be used and TSVD is used in this chapter. We also give an introduction of the Tikhonov regularization below, although it is not used in this thesis.

A) Tikhonov regularization

Noise/error causes the absolute value of the elements of the solution, i.e., λ, largely oscillatory, and to suppress this effect we add the regularization term $\tau\lambda^2$ to suppress the L-2 norm of the solution λ:

$$\underset{\lambda}{\textbf{argmin}}\{\|\mathbf{b} - \mathbf{A}\lambda\|^2 + \tau\lambda^2\} \tag{2.24}$$

where τ is a positive number and termed the regularization parameter. The selection of τ is a trade-off between the perturbation error $\|b - A\lambda\|^2$ and the regularization error $\tau\lambda^2$, i.e., the

relative confidence in US VFI measurements and error robustness. With smaller τ the model will overfit noise and physically pointless, while for large τ the model does not comply with US measurements and lacks essential accuracy.

B) TSVD

Another regularization method is TSVD. The SVD for matrix A is:

$$\mathbf{A} = \mathbf{U\Sigma V^T} = \sum_{i=1}^{N} \mathbf{u_i}\sigma_i\mathbf{v_i^T} \qquad (2.25)$$

where U and V are unitary matrices, N is the number of singular values and $\mathbf{\Sigma}$ is a diagonal matrix composed of singular values in descending order. σ_i is the ith singular value of A. The regularized solution of Eq.2.19 is:

$$\lambda_{reg} = \sum_{i=1}^{k} \frac{u_i^T b v_i}{\sigma_i} \qquad (2.26)$$

where k is the truncation length and used as the regularization parameter for TSVD. As proven in Chapter 2.4.3 using UIV as input $rank(A) = 2m_1 + 3m_2$ and the smallest m_1 singular values of A are 0. For the terms $\frac{u_i^T b v_i}{\sigma_i}$ where σ_i is small, a small error in b will be magnified so that to suppress the error of b we truncate only k terms in Eq. 2.26 where $k \leq 2m_1 + 3m_2$.

Similar to the Tikhonov regularization, k acts as a measure of to what extent the noise contaminated US VFI input should be trusted. Small k means the UIV input is less trusted and results in large regularization error, while large k causes the system sensitive to noise and suffers large perturbation error. Chapter 2 only uses the TSVD to solve Eq. 2.19, and Tikhonov regularization is not considered. The optimal selection of k should balance the regularization error and the perturbation error, and three different methods to optimize regularization parameter are compared in Section 2.9. After solving the regularized weighting coefficient λ_{reg}, 3D full field flow is reconstructed by Eq.2.13.

2.7. Three testing cases

UADFI is tested on three steady flow cases below. Input of Case 1 and 2 is obtained by first extracting 3D velocity vectors on two sets of imaging planes from an analytical solution or CFD, and then calculating their in-plane components. The Reynolds numbers for the three cases are smaller than 1000. Water is the working fluids for the three cases in this chapter. To

avoid expensive computation, the reconstructed flow domain (length and diameter) is small and different flow parameters are used, as introduced below.

Case 1(simulated case): fully developed Poiseuille flow in a straight vessel with a diameter of 10 mm and length 10 mm. The volumetric flow rate Q is 1 ml/s. In plane UIV resolution is 1 mm.

Case 2 (simulated case): helical flow where the helix radius is 5 mm and total length 35 mm. Inlet flow velocity is uniform at 10 mm/s (mean flow rate is 10 mm/s). In plane UIV resolution is 1 mm.

Case 3 (experimental case): steady flow in a tissue-mimicking carotid bifurcation phantom fabricated with polyvinyl alcohol cryogel (PVA-C) with experimental 2D UIV measurements. The inlet flow rate is 0.4 ml/s and in plane UIV resolution is 1 mm.

Geometries of Cases 2&3 are shown in Fig. 2-5 and projected 2D velocity after image registration is shown in Fig. 2-6 (plane spacing=5 mm). The helical flow CFD simulation is conducted in STAR-CCM+ (v11.06, Siemens, Berlin, Germany) and a steady, incompressible laminar flow model is used to simulate the flow, after grid independence is achieved. Errors of case 1 and case 2 are analysed by comparing reconstructed flow with Poiseuille equation and CFD respectively. 5% random artificial Gaussian error is then added to the sampled 2D velocity of the first two cases to test the stability of the algorithm. In Cases 1 and 2, different RBF kernels are also compared and the influence of spacing on accuracy is investigated.

(a)   (b)
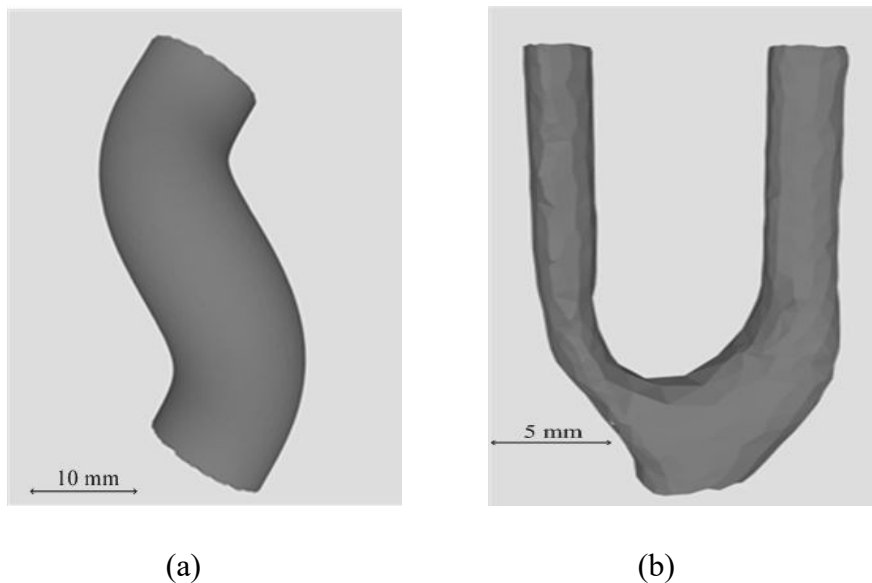
Figure 2-5 Geometries of Case 2&3: (a) helix; (b) carotid bifurcation (surface geometry from US scan)

(a)

(b)

(c)

45
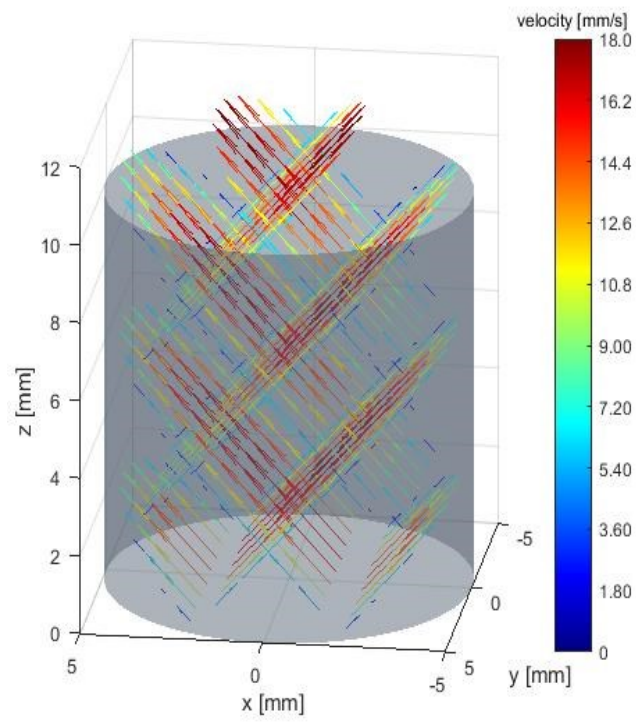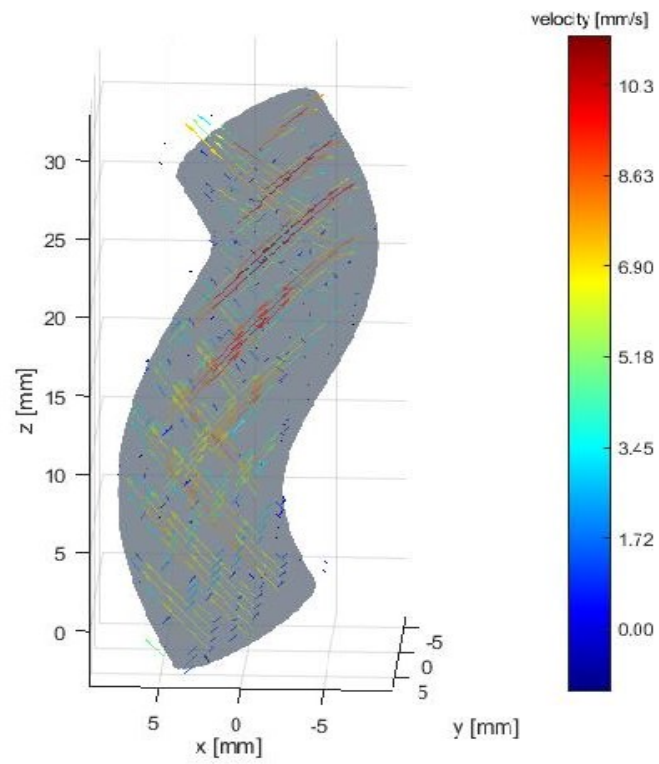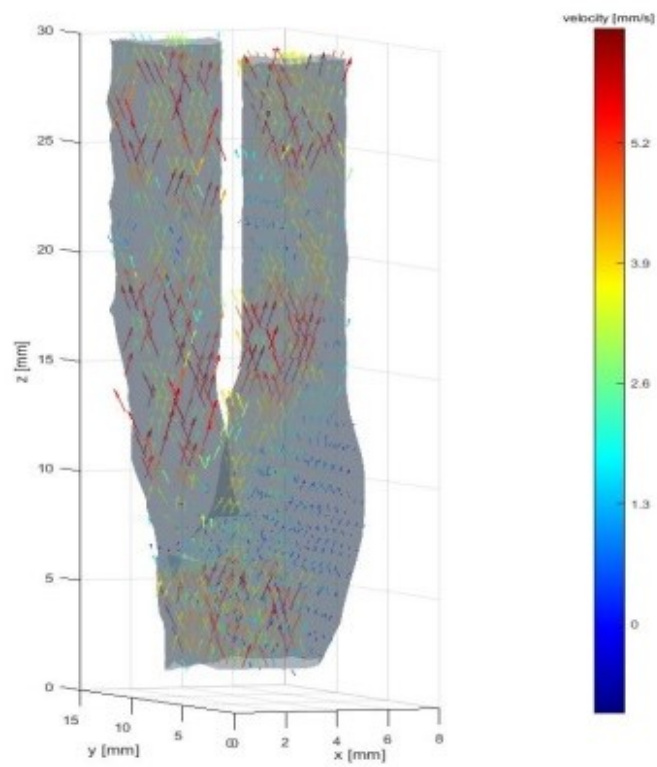
Figure 2-6 Input 2D velocities in 3D space for reconstruction of the three cases: (a) straight tube; (b) helix; (c) carotid bifurcation

2.8. Error metrics

The relative Root Mean Square (RMS) error of flow reconstruction and simulation in this thesis is defined by:

$$\mathbf{Err} = \sqrt{\frac{\sum(u_i - ut_i)^2 + (v_i - vt_i)^2 + (w_i - wt_i)^2}{\sum ut_i^2 + vt_i^2 + wt_i^2}} \qquad (2.27)$$

where u_i, v_i and w_i are the three velocity components along x, y and z axis, and ut_i, vt_i and wt_i are corresponding velocity components from the reference.

2.9. Methods to Choose the optimal Regularization Parameter

Three methods are usually used to find the optimal regularization parameter: Generalizaed Cross Validation (GCV) (189), Normalized Cumulative Periodogram (NCP) (190) and the L-curve (191). The theories of the three methods are not introduced, and only the results using TSVD with Case 1 are compared in Fig.2-7. We use the corresponding functions provided by the matlab toolbox regtools for the three regularization methods. To test the accuracy and robustness of the three methods, we add random Gaussian noise to vector b and locate the optimal regularization parameter by the three methods, and compare the reconstructed flow field by the three methods with the ground truth.
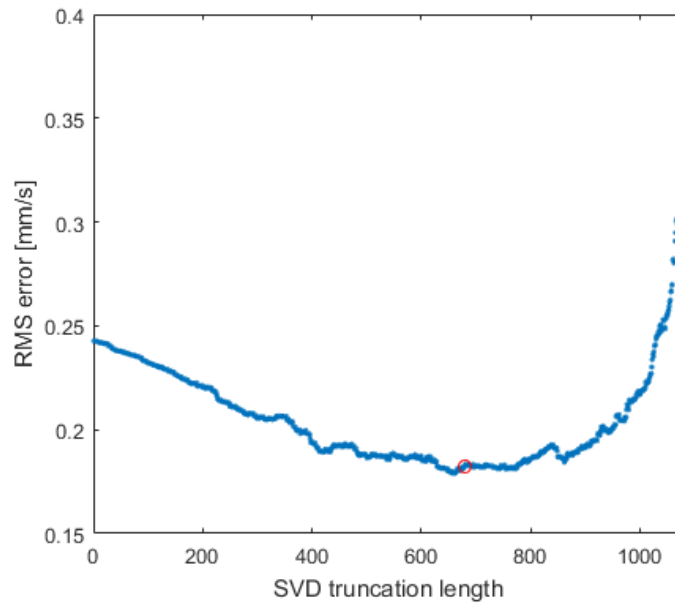
Figure 2-7 The relative error using different truncation length for Poiseuille flow case (m= 540), with 5% artificial random Gaussian noise

46

It can be derived from Eq.2.26 that when k > rank(A) (1080 for Fig. 2-7) the error is wildly oscillating because the denominator is close to the machine precision (epsilon using finite digit algorithm, ~2.2204e-16 using double precision floating point arithmetic in matlab), so that we only show k ≤ rank(A) in Fig. 2-7. The optimal regularization parameters k selected by NCG and GCV are larger than rank(A) and thus are wrong and not shown. Fig. 2.7 shows that with random Gaussian artificial noise the L curve method finds the optimal regularization parameter for UADFI, and the L-curve method is used as the default method for selection of the truncation length for TSVD in this thesis.

All three cases in this chapter are conducted at double precision in Matlab R2016b (The MathWorks, Natick, MA, USA). All simulations were performed with an Intel Core i7 3.4 GHz HP EliteDesk 800 G1 Tower workstation with 32 GB RAM-memory.

2.10. Results

2.10.1 Poiseuille flow case

As in practical applications, in plane resolution is kept constant and the influences of different RBF kernels and spacing between ultrasound acquisition planes on reconstruction accuracy are studied below.

A) reconstruction accuracy using different kernels

Table 2-1 compares the four different kernels (spacing =2 mm, ε=1). The results show that Gaussian, IM and MU have higher reconstruction accuracy than the TPS kernel (IM- inverse multiquadric, MU –multiquadric, TPS- Thin Plate Spline), and IM/multiquadric kernels have highest accuracy. This may be because TPS does not contain ε and has a fixed spatial support, while the other three has a larger spatial support. Gaussian is a widely used kernel function for interpolation and is chosen as the default kernel in the rest part of the thesis.

Table 2-1 Absolute error of four kernels for Poiseuille flow reconstruction using noise free input

Kernels	Error (mm/s)
Gaussian	3.99e-04
IM (inverse multiquadric)	3.0652e-04
Multiquadric	5.8330e-04
TPS (Thin Plate Spline)	0.6358

Fig. 2-8 shows the reconstructed 3D flow using Gaussian kernel and the ground truth from analytical solution, and Fig. 2-9 shows the velocity distribution at the cross and sagittal plane for sanity check. The reconstruction accuracy listed in Table 2-1 shows that high accuracy for Poiseulle flow can be achieved by the proposed method.
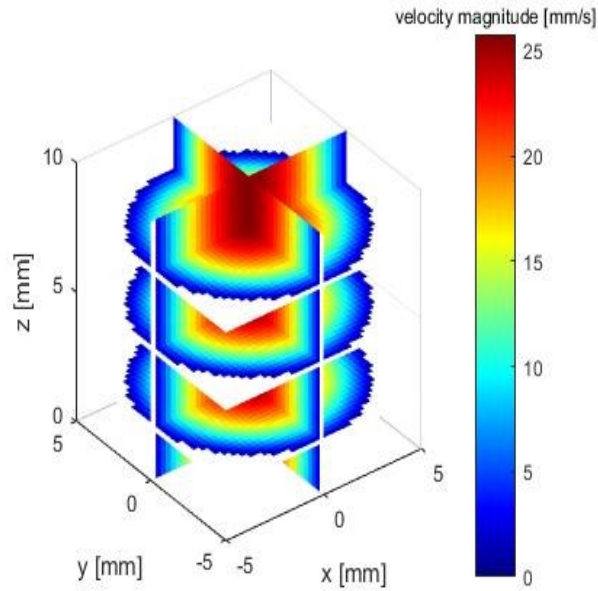
(a)
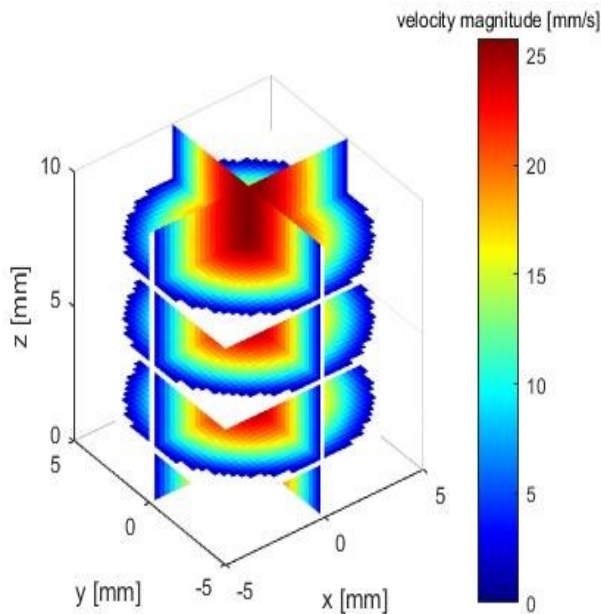
(b)

Figure 2-8 Poiseuille flow: (a) ground truth; (b) reconstructed flow field by Gaussian RBF

(a)   (b)
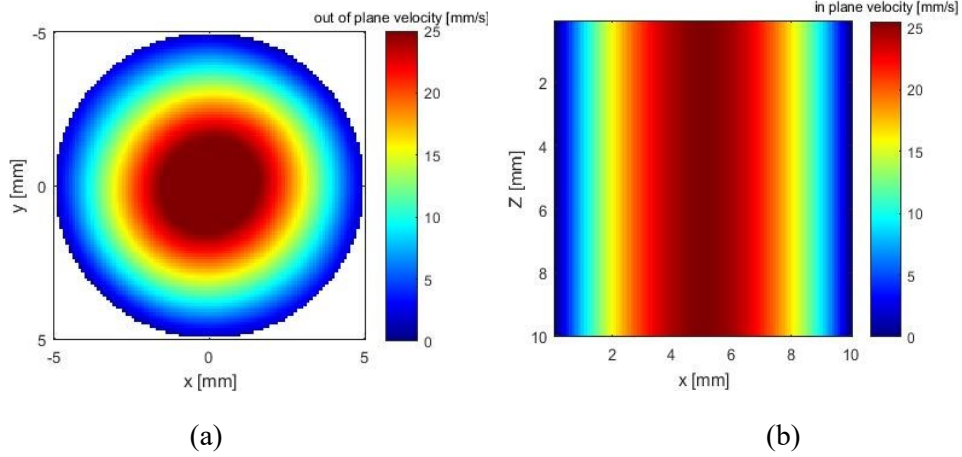
Figure 2-9 Poiseuille flow: (a) out of plane velocity at cross plane; (b) in plane velocity at sagittal plane

B) reconstruction accuracy using different spacing

To reduce the time/cost of ultrasound acquisition and UIV post-processing, it is of great interest to increase the spacing between acquisition planes and evaluate the accuracy when spacing is large. The acquisition time constraint is especially important in the case of CEUS (Contrast Enhanced Ultrasound) imaging so that acquisitions in all necessary planes for reconstruction have to be within the current FDA limits of contrast agent administrations in humans. Spacing is varied from 1 mm to 9 mm. It is shown in Fig. 2-10 that increasing spacing reduces the reconstruction accuracy, and the reason may be that low sampling density fails to fully capture the detailed flow features especially when the flow is complex. Fig. 2-10 shows that using a fixed shape parameter ε, for noise free input with a fixed value of in-plane resolution (1 mm), the accuracy is highest when the centres are nearly spatially uniform (spacing is close to in-plane resolution which is ~1mm in Chapter 2). This show that the selection of ε may have an important influence of the spatial width of the RBF function and affect the accuracy, and this will be further studied in Chapter 3. Fig. 2-11 shows the reconstructed flow using Gaussian and TPS kernels with different spacings. The in-plane resolution is fixed and so is ε (chosen as the reciprocal of the in-plane resolution in this chapter). Similarly the large jump at 6 mm may be due to lost spatial support of Gaussian RBF. Gaussian function loses spatial support (its value drops to ~1% of its peak value) when the distance to its centre is larger than 3* ε, and thus the accuracy may decrease dramatically when spacing is larger than 6* ε.
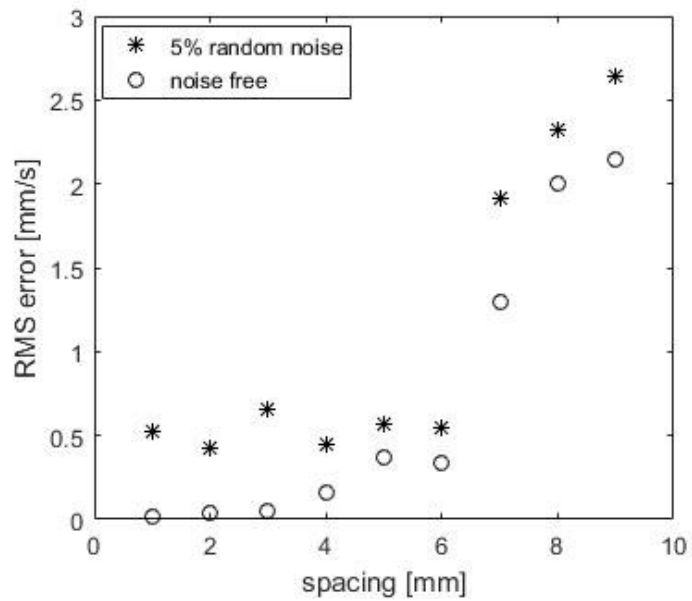
Figure 2-10 RMS error of Poiseuille flow at different spacing by Gaussian kernel, with or without 5% random noise (in-plane resolution is 1 mm)
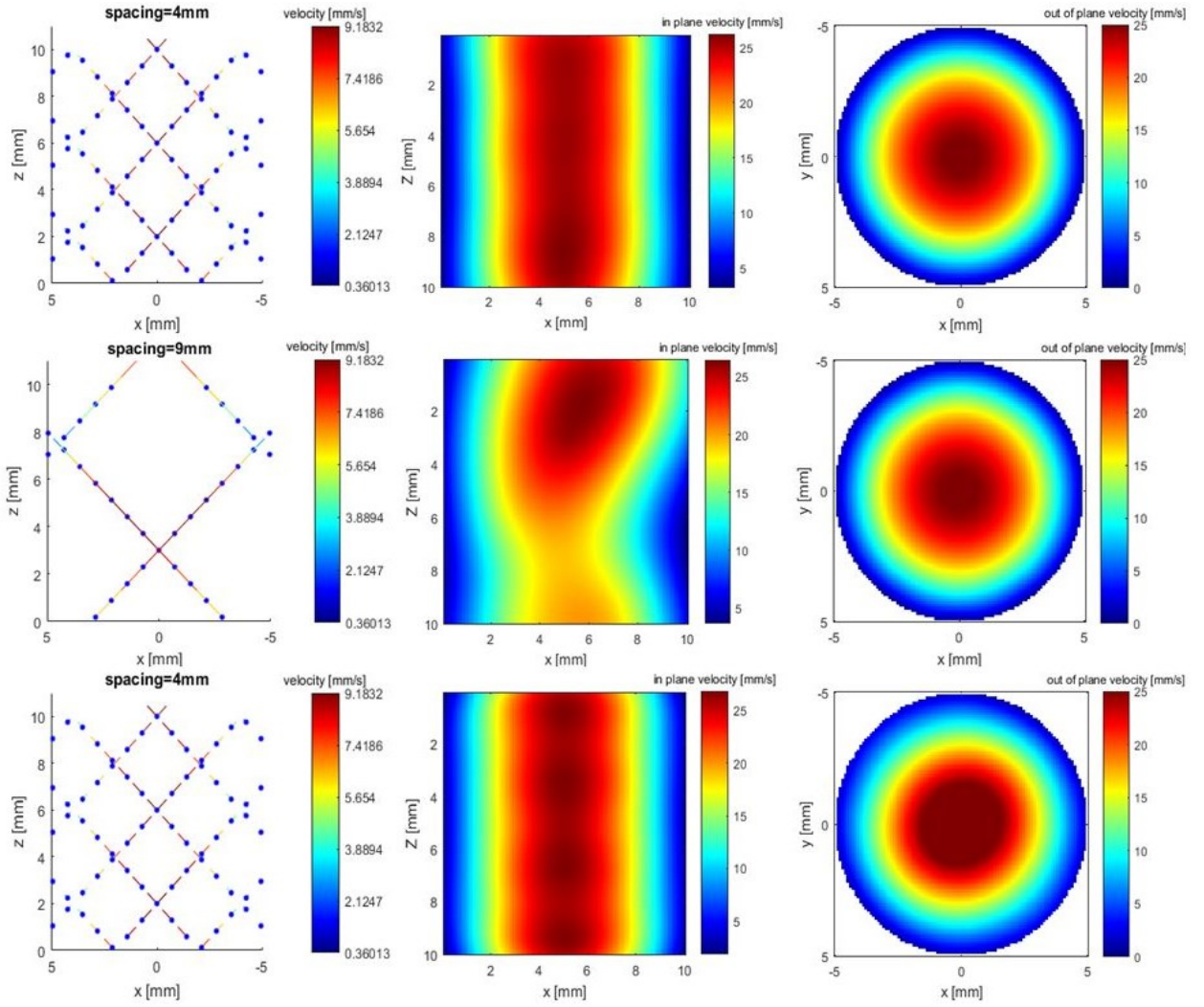
Figure 2-11 Poiseuille flow. The first two rows are interpolated flow using Gaussian kernel at spacing=4 mm and 9 mm respectively. The last row is using TPS kernel at spacing=4 mm. The three columns are sampled velocity, in plane velocity at sagittal plane and out of plane velocity at cross plane. The last column is the velocity magnitude at the middle plane (z=5 mm)

UADFI is ill-conditioned and to evaluate its numerical stability, 5% random Gaussian noise is added to the 2D velocity input, i.e., $\frac{\|b-\hat{b}\|_2}{\|\hat{b}\|_2} = 5\%$. The error of reconstructed flow, shown in Fig. 2-10 indicates that the algorithm is robust with measurement noise, and the accuracy generally decreases when spacing is larger. The relative error with 5% noise input increases from 4.4% to 20% (mean flow rate is 12.73 mm/s) when spacing increases from 1 mm to 9 mm.

2.10.2   Helical flow case

A) Reconstruction accuracy using different kernels

CFD result using STAR-CCM+ and the reconstruction result for helical flow are illustrated in Fig. 2-12. The CFD result is obtained on a non-uniform grid generated automatic by STAR-CCM+ and then interpolated to a regular grid, for comparison with the reconstructed flow evaluated on a regular grid. The CFD result is slightly blurred due to this interpolation and this may cause slight error.
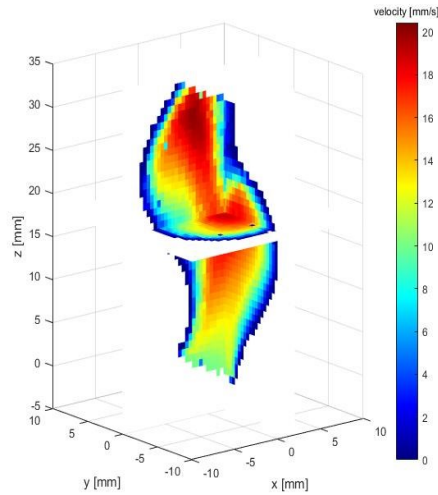
(a)
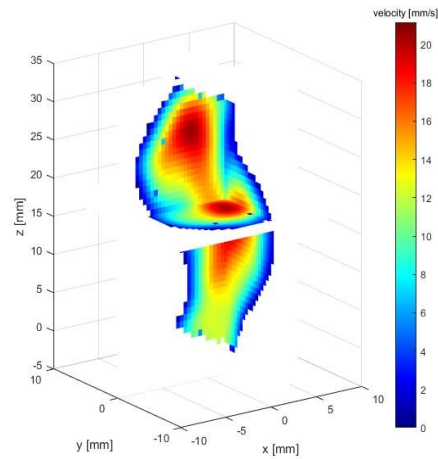
(b)

Figure 2-12 (a) 3D velocity of helical flow by CFD; (b) reconstructed 3D flow by Gaussian RBF

Table 2-2 shows the error using four kernels and the other three RBFs have higher accuracy than TPS whose mean relative error is 14% (spacing=2 mm, ε=1).

Table 2-2 Absolute error of four kernels for helical flow reconstruction using noise free input

Kernels	Error (mm/s)
Gaussian	0.9169
IM (inverse multiquadric)	0.9791
Multiquadric	1.0365
TPS (Thin Plate Spline)	1.4036

Table 2-2 shows that Gaussian, IM and multiquadric kernels have similar level of accuracy, and are more accurate than TPS. This is likely to be caused by the fixed spatial support of TPS. The accuracy of helical flow reconstruction is lower than Poiseulle flow, and this is due to that the input of helical flow is from CFD and contains larger error than the Poiseulle flow whose input is from analytical solution. The first column in the fourth row of Fig. 2-13 (TPS) has large error near the wall, and it may be caused by: 1. TPS does not have the shape parameter and has fixed spatial support; 2. the sample velocities as well as centres are located in two perpendicular planes that are both parallel to x-axis, and the error is higher in regions that are far from RBF centres.
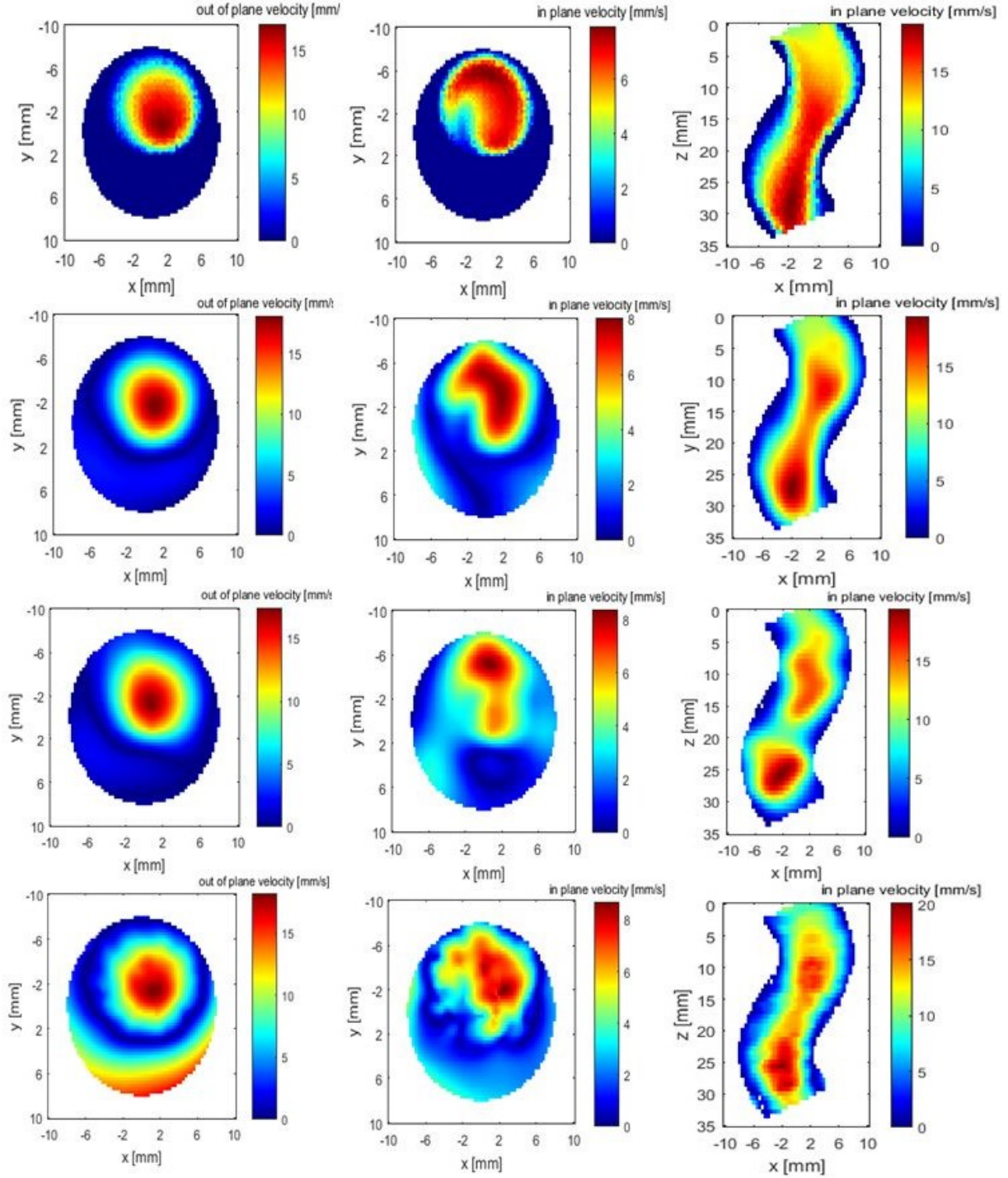
Figure 2-13 Helical flow. The four rows are: ground truth by CFD, reconstructed flow field using Gaussian kernel at spacing=2 mm and 8 mm, TPS at spacing=2 mm. The three columns are: out of plane velocity at cross plane, in plane velocity at cross plane, out of plane velocity at sagittal plane

B) Reconstruction accuracy for different spacing

Similarly to Case 1, the error increases with increasing spacing. Fig. 2-14 shows the error increases when spacing increases, and relative error increases to 14.4% for noise free input when spacing is 8 mm.
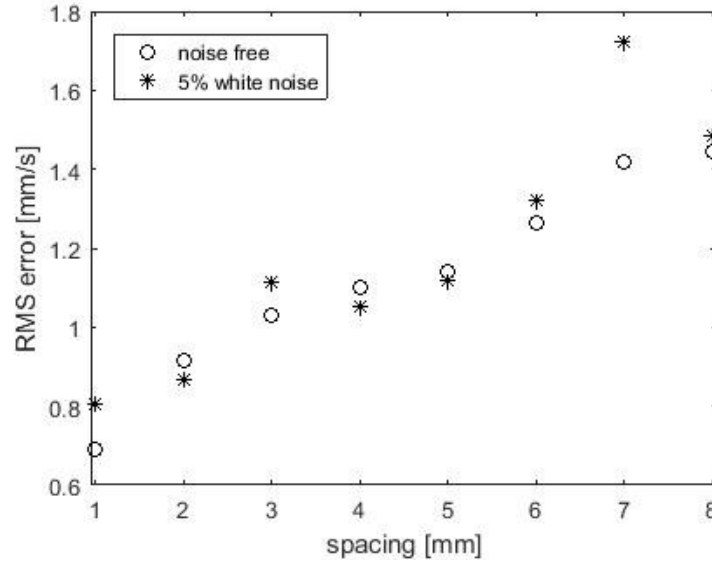
Figure 2-14 RMS error of helical flow with different spacings using the Gaussian kernel, with no input noise or with 5% white noise

It is interesting to note reconstruction error with noise-free input is not always lower than those with noise, which indicates that a sub-optimal regularization parameter is achieved by the L-curve method. The L-curve method is based on the assumption that the input noise is white but sometimes this assumption is not valid because numerical error and measurement errors are not spatially uniform. This motivates exploration of other regularization methods in Chapter 3.

2.10.3 *In vitro* experiment case

Fig.2-15 shows the reconstructed flow of Case 3 and we did not conduct CFD because the difficulty in measuring accurate boundary condition (especially flow inlet/outlet conditions) and CFD geometry. As a result CFD has compromised accuracy due to large input uncertainty. Due to lack of ideal standard for error evaluation, we did not compare influence of spacing/kernel on accuracy for the *in vitro* case, and Gaussian is used for reconstruction. The plane spacing of input is 5 mm. The reconstruction accuracy for *in vitro* flow is largely affected by the registration of the mechanical stepping system that is used to move the US probe. The mechanical stepping system built in the lab has a relatively large registration error, and this may affect the reconstruction accuracy so that we only show the experimental feasibility in Fig. 2-15. Fig. 2-15 shows reasonable velocity magnitude as well as successful recovery of 3D velocity from 2D measurements, although the accuracy is not evaluated.
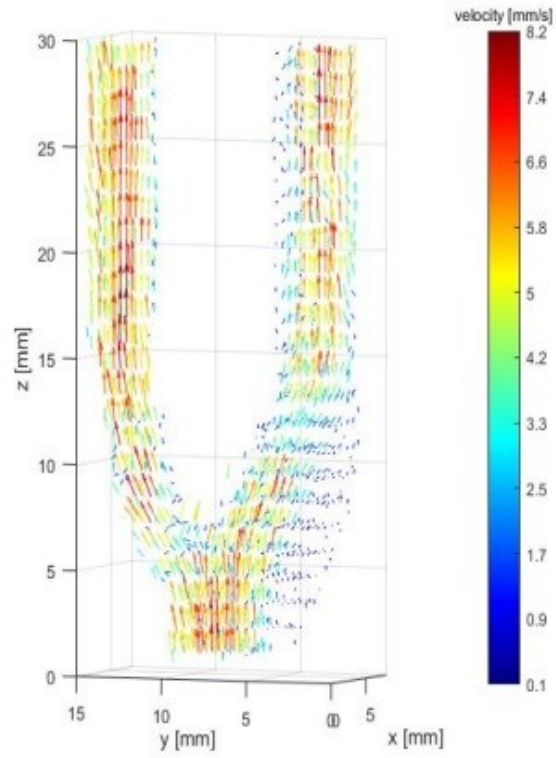
(a)
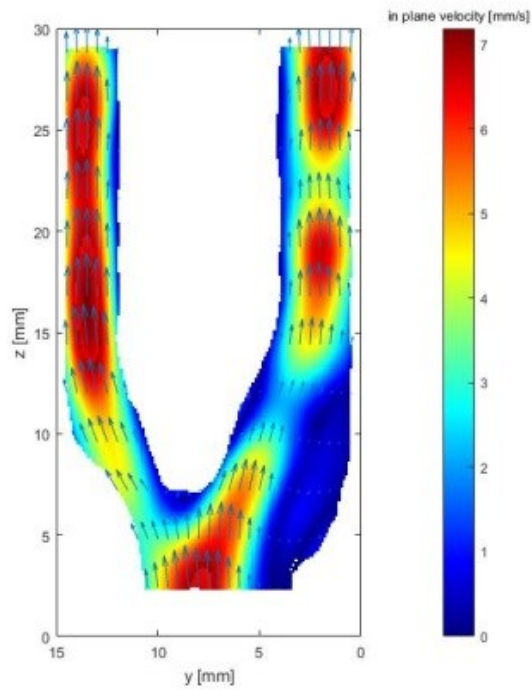
(b)

Figure 2-15 *In vitro* flow experiment: (a) reconstructed steady 3D flow; (b) velocity magnitude contour at the cutting plane of x=3 mm

2.11. Summary

This chapter demonstrates the generation of 3D flow reconstruction through integrating 2D ultrafast plane wave imaging, ultrasound imaging velocimetry, microbubble contrast agents and DFI. Previous 3D DFI and flow reconstruction methods require 3D velocity input and to the best of our knowledge this is the first application of 3D flow reconstruction using 2D velocity input. It fills a current gap between 3D DFI and 2D experimental ultrasound measurements. By reconstructing a full 3D flow velocity field with sparse samples of 2D measurements, it overcomes the problem of current 2D ultrasound measurements by providing full 3D velocity field. Unlike current CFD which usually only utilizes geometry and inlet/outlet condition from imaging, UADFI reconstructs the 3D full velocity field from experimental 2D in plane velocity input, can achieve reasonable accuracy and is robust to measurement noise. This method is simple, computationally efficient compared with CFD, mesh-free, independent of initial/boundary condition, and thus has potential for faster flow estimation in biomedical applications compared with CFD. The reconstruction accuracy and reconstruction speed are discussed below.

A). Influence of experimental/reconstruction parameters on accuracy

Reconstruction accuracy is related with UIV experimental parameters (including spacing, acquisition angles, US frequency, compounding angles), UIV post-processing parameters (size of the interrogation window), and UADFI parameters (including different RBF kernels, shape parameter which is not mentioned in this chapter and will be studied in Chapter 3 and regularization parameter etc).

It should be noted that the required spacing depends on the complexity of the flow. The proposed method will interpolate the flow field taking into account mass conservation, and recover out of plane velocities to generate a full 3D velocity field, but it would be impossible for the method to retrieve information about a complex local vortex flow if it is not being sampled in the imaging planes. Therefore we have to sample sufficiently dense (i.e., use smaller spacing) to visualize complex and localized vortex flow, at the cost of slower reconstruction and UIV acquisition. In Case 3 of this study we tested the plane spacing at 5 mm for real UIV measurement and the results show the experimental feasibility of the proposed method. UIV accuracy depends on the acquisition angle and higher UIV accuracy is achieved when the angle between flow direction and the acquisition plane is smaller. However, for real acquisition it is difficult to know the flow direction beforehand and thus in

this study we use two perpendicular acquisition angles to maximize the independence of flow information acquired from two angles. UIV experimental error can be below or around 10% according to (184), and by optimizing UIV parameters such as the number of plane waves compounded, plane wave tilting angle, frequency, acoustic pressure, and pulse length UIV accuracy can be improved.

RBF kernels also influence accuracy and previous study has compared Gaussian and TPS kernels with 3D velocity input (192). This study shows Gaussian, IM and MU have higher accuracy than TPS, because TPS has fixed spatial support while the spatial support of the other three kernels can be larger by adjusting the shape parameter. In this study we added 5% Gaussian noise in Case 1 and 2 (17 numerical experiments in total, with different spacing), and the results show the robustness of the method to noise. Shape parameter of RBF is an open research topic and difficult to optimize. It is usually found by trial and error or by cross validation (193, 194) and is chosen as the reciprocal of the in-plane resolution in this chapter. The regularization parameter also plays a vital importance in increasing accuracy and suppressing the influence of noise, and it can be explicitly optimized by the L-curve method.

B). Reconstruction time

For the first two cases the reconstruction time takes less than 3 minutes (spacing=2 mm, number of input vectorial points <1000) when data acquisition and UIV post processing are not included. CFD using Star-CCM+ in the same workstation for Case 2 takes around one hour.

The reconstruction time of the third case (spacing=5 mm, number of input vectorial points <3000) takes less than 15 minutes, including UIV experiment and UIV post processing. Image acquisition of each plane takes a matter of microseconds and total US acquisition of the full length takes around 1 minute (including movement of stage controller, data transfer and beamforming). UIV post-processing is accelerated by GPU and takes several seconds at each plane. SVD of matrix G involves $O((3m)^3)$ flops and the reconstruction time could be significantly reduced if m is reduced, i.e., the vectorial data is downsampled or spacing is increased. In addition a highly optimized iterative solver with proper regularization (e.g., Generalized Minimal Residual method with early stopping) has the potential to reduce the reconstruction time to a matter of seconds, and this will be introduced in Chapter 3.

C). Limitation

One limitation of UADFI is the mechanical translation of the ultrasound probe, as this may be not feasible in some applications, and makes acquisition slow and may cause spatial registration error, particularly when the target moves during the translation. To solve this problem it is possible to electrically rotate and wobble the US imaging plane, or use a matrix probe.

Only steady flow is studied. For periodic physiological flow a method is to use one probe to acquire flow for a full cardiac cycle (~1s) at each location and then the probe translates to the next location. After acquiring 2D flow information of a full cardiac at each location, the velocity field is registered temporally, e.g., aligning the peak 2D flow rate at different locations temporally or using ECG gating.

Chapter 3 Optimization of VFI Augmented DFI reconstruction

3.1.   Introduction

Chapter 2 proposes a new UADFI algorithm which is capable of reconstructing full field 3D velocities from 3D distribution of multi-planar 2D velocities using a 2D US system. 2D velocities are acquired by High Frame Rate UIV using a linear probe with a frame rate up to tens of thousands per second. The algorithm assumes the measured 2D velocities are merely planar projection of the 3D flow velocity vectors, and solves the unknown weighting coefficients of divergence free basis functions of the resulting singular system using Truncated Singular Value Decomposition (TSVD) and the L-curve method (195). As a result, a cost-effective 1D linear US probe can be used to recover the out of plane velocity components, with the mass conservation imposed to filter measurement noises. The feasibility of the technique to reconstruct 3D vectors from 2D ultrasound UIV measurements has been demonstrated in Chapter 2. However the reconstruction for a typical vascular flow field takes several minutes and the shape parameter is chosen to be reciprocal of the in-plane resolution for simplicity and has not been optimized.

DFI constructs a linear vector space using divergence free functions containing the shape parameter ε. ε is related with the variance (or spatial width) of the basis function and its selection is a trade-off between accuracy and robustness, and to achieve higher accuracy shape parameter optimization is necessary. With a small shape parameter each basis function has a larger spatial support and is more accurate for interpolation, but more numerically ill-conditioned and thus sensitive to measurement noise. Radial Basis functions with a large shape parameter are compactly supported, robust to measurement noise but lack essential accuracy for interpolating. An optimal shape parameter is required to balance robustness and accuracy. Shape parameter optimization has been well studied for Radial Basis Function (in this study we use Gaussian function) for scalar interpolation (193, 196-199) but not for DFI or UADFI which is more complex. Another problem related with shape parameter of DFI is the numerical oscillation between the interpolation points especially near the interpolation domain boundary (196), and is termed the Runge-Phenomenon (RP). Suppression of RP reduces interpolation error especially near the flow boundary.

This chapter aims to significantly improve the accuracy and reduce the time-to-solution of our previous approach thereby paving the way for clinical translation. More specifically, accuracy is improved by optimising the divergence free basis to reduce Runge-phenomena near domain boundaries, and time-to-solution is reduced by demonstrating that under certain conditions the resulting singular system can be solved using widely available and highly optimized Generalized Minimum Residual (GMRES) algorithms.

3.2. Hold out validation for shape parameter optimization

The work reported in (200) shows that RP can be completely suppressed with the optimal selection of shape parameter. In this study we aim at finding the optimal shape parameter using the holdout validation to construct the optimal divergence free basis and suppress near-boundary numerical oscillation due to RP.

The holdout validation is a kind of cross validation and has the advantage of low computational overheads. It assigns data into the fitting set and the testing set, solves the system using the fitting set and validates the results on the testing set. US B-mode resolution in axial resolution is higher than lateral resolution and so is UIV vector resolution (184). In this study the axial resolution and lateral resolution of UIV are ~0.25 mm and ~0.5 mm respectively, and are both downsampled to ~ 1 mm for simplicity. The downsampled velocities are used as the fitting set and the rest velocities are the testing set. The optimal shape parameter is chosen with three steps:

1.   loop different values of ε in a range (around the reciprocal of the in-plane resolution), and solve λ_{reg} in Eq. 2.19 using the fitting set;

2.   use the solution λ_{reg} obtained from Step 1 to compute the 3D velocity at the testing set using Eq. 2.13;

3.   conduct corresponding projection for every 3D velocities of the testing set obtained from Step 2 to calculate their 2D projected velocities onto the imaging plane (different velocity vectors may have different projection matrices depending on the experimental setup), and compare with those obtained from experiments to evaluate the accuracy.

The optimal shape parameter is the one that has the highest accuracy from Step 3, denoted as:

$$\underset{\varepsilon}{\mathbf{argmin}}\left|\hat{\mathbf{R}} * \hat{\mathbf{v}}(\boldsymbol{\varepsilon}) - \hat{\mathbf{v}}_{\mathbf{m}}\right|_{\mathbf{2}} \qquad (3.1)$$

where $\hat{R}$ is the projection matrix for the testing set, $\hat{v}(\varepsilon)$ is the interpolated 3D velocity with shape parameter ε and $\hat{v}_m$ is 2D velocity at the testing set from US measurements. After

finding the optimal shape parameter for the particular in-plane resolution and spacing, ε is fixed for further reconstruction.

3.3. GMRES and Krylov subspace

GMRES is a widely used method for the numerical solution of a nonsingular system of linear equations. The cost of the GMRES is $O(m^2)$, compared with $O(m^3)$ for SVD. GMRES approximates the kth iterate x_k in a finite linear space, i.e., $x_k \in K_k = \text{span}\{b, Ab, A^2b, \cdots, A^{k-1}b\}$, to a nonsingular and square system $Ax = b$ by minimizing the residual $r_n = |Ax_n - b|_2$ (201). K_k is usually constructed by Arnoldi' iteration (202) or Householder transform (203). For a nonsingular system the condition $\text{dim}A(K_k) = \text{dim}K_k = k$ always stands and GMRES does not break down. (204) proves that GMRES breaks down at the kth iteration for (nearly) singular systems in two situations: 1. $\text{dim}A(K_k) < k$; 2. $\text{dim}K_k < k$.

In this study matrix A in Eq. 2.19 is singular and GMRES cannot be used directly unless the condition $\text{dim}A(K_k) = \text{dim}K_k = k$ is satisfied before a solution is found. Below we prove that for 3D reconstruction using 2D UIV it stands naturally iff the iteration step $k < 2m_1 + 3m_2$, and proposed some matrix transformations/permutations to improve the sparsity of the problem system and thus the efficiency of GMRES for our particular study.

A). Rank analysis of UADFI Krylov subspace

It has been proven in Section 2.5.3 that for 3D reconstruction with 2D UIV, $\text{rank}(R) = 2m_1 + 3m_2$ by analyzing the eigenvalues of matrix R, and $\text{rank}(A) = 2m_1 + 3m_2$. Below we conduct rank analysis of A^k with $k>1$. Based on Eqs.2.20-2.23 we conduct the following permutation:

$$\mathbf{R} = \widehat{\mathbf{Q}} * \mathbf{P}^{-1} * \mathbf{P} * \widehat{\mathbf{D}} * \mathbf{P} * \mathbf{P}^{-1} * \widehat{\mathbf{Q}}^{\mathbf{T}} = \widehat{\mathbf{Q}} * \widehat{\mathbf{D}} * \widehat{\mathbf{Q}}^{\mathbf{T}} \qquad (3.2)$$

where P is the permutation matrix and $P^{-1} = P^T$. By conducting permutation the m_1 zero diagonal elements of D are moved to the end of the diagonal positions. After permutation $\widehat{D}$ is a diagonal matrix whose first $2m_1 + 3m_2$ diagonal elements are 1 and the last m_1 diagonal elements are 0. We have $Q = \widehat{Q} * P^{-1}$, and the orthogonal matrix Q has $Q^T = Q^{-1}$. Then we have:

$$\mathbf{A} = \mathbf{R} * \mathbf{G} = \widehat{\mathbf{Q}} * \widehat{\mathbf{D}} * \widehat{\mathbf{Q}}^{\mathbf{T}} * \mathbf{G} \qquad (3.3)$$

and

$$A^k = \widehat{Q} \underbrace{\widehat{D}\widehat{Q}^TG * \widehat{Q}\widehat{D}\widehat{Q}^TG * ... * \widehat{Q}\widehat{D}}_{M^{k-1}} \widehat{Q}^T * G \qquad (3.4)$$

$M = \widehat{D}\widehat{Q}^TG\widehat{Q}\widehat{D}$ and can be denoted as a block matrix. Left and right multiplying the nonsingular matrix $\widehat{Q}^TG\widehat{Q}$ by $\widehat{D}$ is equal to setting its last m_1 rows and m_1 columns to all zeros. As a result the $2m_1 + 3m_2$th order principle submatrix of M, denoted by M_{sub}, is a nonsingular matrix. According to block matrix multiplication:

$$M^{k-1} = \begin{bmatrix} M_{sub} & 0_{m_1+m_2} \\ 0_{m_1+m_2} & 0_{m_1+m_2} \end{bmatrix}^{k-1} = \begin{bmatrix} M_{sub}^{k-1} & 0_{m_1+m_2} \\ 0_{m_1+m_2} & 0_{m_1+m_2} \end{bmatrix} \qquad (3.5)$$

we have $\text{rank}(M^{k-1}) = \text{rank}(M_{sub}^{k-1}) = 2m_1 + 3m_2$ and thus $\text{rank}(A^k) = 2m_1 + 3m_2$.

B) break-down free GMRES with conditions

Below we calculate $\dim A(K_k)$ and $\dim K_k$ for UADFI. Using the conclusion $\text{rank}(A^k) = 2m_1 + 3m_2$ we have:

1. $A(K_k) = \text{span}\{Ab, A^2b, \cdots, A^kb\}$ and $\dim A(K_k) = k$ when $k \leq 2m_1 + 3m_2$;

2. $\dim K_k = \dim \text{span}\{b, Ab, \cdots, A^{k-1}b\} = k$ when $k \leq 2m_1 + 3m_2$.

In conclusion $\dim A(K_k) = \dim K_k = k$ when the iteration step k satisfies $k \leq 2m_1 + 3m_2$ and GMRES for UADFI will not break down.

C) Further speed up for GMRES

First we conducted a series of matrix transformations/permutations to increase the sparsity of the system. We combine Eqs. 2.19 and 3.2 and have:

$$\widehat{D} * \widehat{Q}^T * G * \lambda = \widehat{Q}^{-1} * b \qquad (3.6)$$

$\widehat{D} * \widehat{Q}^T * G$ is a square matrix whose last m_1 rows are 0. We also set the last m_1 elements of $\widehat{Q}^{-1} * b$ to be 0 as they do not influence the solution of λ. Then a similar method to Algorithm 2.2 in (203) was applied to Eq. 3.6, where the maximum iteration is $2m_1 + 3m_2$. GMRES

applied to Eq. 3.6 is more computational efficient than directly applied to Eq. 2.19 given the sparsity of matrix $\widehat{D} * \widehat{Q}^T * G$.

3.4. regularization of GMRES on UADFI

We prove in Section 3.3 that GMRES is effective for Eq. 2.19 (or Eq. 3.6) when the maximum iteration step is $2m_1 + 3m_2$. For a circular vessel with an intraventricular diameter of 5 mm and length 20 mm, the typical values of m_1 and m_2 are both ~1000 if the UIV resolution is 1 mm and spacing is 1 mm. Another problem is, the input velocity vector b contains input errors and to minimize the impact of input errors (i.e., to avoid overfitting), we have to find the regularized solution to Eq. 2.19 (or Eq. 3.6) by early stopping as follows:

$$\left| A\lambda_{reg} - b \right|_2 < \zeta \qquad (3.7)$$

where ζ is the relative residual and used as the regularization parameter in this section. The residual ζ is defined using the similar equation to the RMS error in Eq. 2.27. ζ is empirically determined by a trade-off between accuracy and robustness, and quantifies the extent that US input deviates from the DFI model. The input error may result from VFI measurements, spatial and temporal registration, model simplification (i.e., DFI ignores the physics of momentum conservation), image segmentation, floating points computation etc. With fixed shape parameter, solving the system using the fitting set and evaluating the reconstruction accuracy on the unseen validation set, the accuracy of GMRES decreases first and then increases dramatically. This indicates that apart from the impact of shape parameter, the accuracy is limited by potential overfitting. Large ζ leads to suboptimal accuracy but faster convergence because a smaller number of iterations is needed. On the contrary small ζ leads to potential overfitting and thus is unstable due to overfitting and is also slower. To force GMRES to stop before overfitting, the regularization parameter ζ is usually slightly larger than the relative error of the input 2D velocities (vector b), and is termed the 'early stopping' principle. However in practice the input error level is usually unknown and ζ has to be determined empirically.

In summary a regularized solution λ_{reg} of Eq. 2.19 is calculated using the more efficient GMRES by setting the maximum iteration step $2m_1 + 3m_2$, tolerance ζ, after some linear transformation applied to the system to increase sparsity, with the optimal shape parameter chosen by the holdout validation. After solving λ_{reg} the 3D velocity can be calculated by Eq. 2.13 using λ_{reg} as the coefficients of the basis functions.

Although the reconstruction algorithm can be potentially applied for the WSS evaluation after 3D flow reconstruction, WSS is highly sensitive to the normal direction of the surface constructed from US and can be very noisy. We did not use the UADFI algorithm for WSS evaluation.

3.5. Testing cases

First two *in silico* reconstructions are conducted in this section, where 3D reconstructions are conducted with 3D distribution of 2D velocity components. Then two 3D *in vitro* constructions are conducted, where the input is 2D velocities from US UIV measurements. Given that the input of the four cases are from different sources, the relative input errors of the four cases are different and so is ζ. The interested flow domains of the four cases are small to avoid large memory usage, as constructing a large and dense matrix A can be memory expensive.

A). 3D flow reconstruction using 2D in-plane velocities *in silico*

Case1&2 are 3D reconstruction using artificial 2D in-plane velocities to simulate UIV input. We first extract the 3D velocities at the fictitious imaging plane from reference (analytical solution or CFD obtained by STAR-CCM+), and then project them onto two sets of equal-interval orthogonal planes as UADFI input.

Case 1: pulsatile water Womersley flow in a straight circular vessel with diameter 10 mm and reconstruction domain length of 25 mm. The dimensionless Womersley number is 6.78. The

3D velocity at 50 different time steps in a full cardiac cycle (1 second) is calculated by Womersley's arterial flow model (205), which represents flow velocity as an infinite Fourier series. In this study we truncate it to 8th order and set $\zeta=8\%$. Analytical solution is used as reference so that no boundary conditions are needed.

Case 2: steady water flow in a helical vessel with diameter 10 mm and reconstruction domain length of 20 mm (Fig. 2.5a). We set $\zeta=10\%$. We set a volume flow rate (equivalent to a Reynolds number of 1000) at the inlet and atmospheric pressure at the outlet.

B) 3D flow reconstruction using 2D in-plane velocities in phantoms

Two *in vitro* experiments were conducted, where UIV is used to acquire the 2D in-plane velocities and the experimental setup is the same with Chapter 2. Consider the registration error of stage motion system and UIV post-processing, input has larger error than the *in silico* cases and to avoid overfitting we set $\zeta=15\%$.

Case 3: steady water flow in a straight vessel phantom with diameter 6 mm and reconstruction domain length of 25 mm, with a long entrance length so that inlet flow velocity profile is fully developed. Analytical solution is used as reference.

Case 4: steady water flow in a bifurcation phantom. The inlet diameter is 5.3 mm and reconstruction domain length is 30 mm. The geometry of the bifurcation phantom is shown in Fig. 2.5b.

3.6.   Results

The influence of the relative spacing $\delta d = \frac{d}{D}$ (probe motion spacing normalized by vessel diameter) on accuracy and reconstruction time for Cases 1 & 2 is shown in Fig. 3.1a and Fig. 3.1b. It shows a trade-off between accuracy and reconstruction time, where faster reconstruction is achieved with larger relative spacing at lower accuracy. Whereas less US

acquisitions are of more clinical interest it is necessary to find the largest possible acquisition spacing with accepted accuracy. An accepted relative error level of 5% can be achieved with reconstruction time <4 s for the two cases when δd = 0.2, and in this study we set δd = 0.2 for all cases. The angles between the US probe and the probe motion direction are the same with Chapter 2, i.e., ±45º.

(a)   (b)
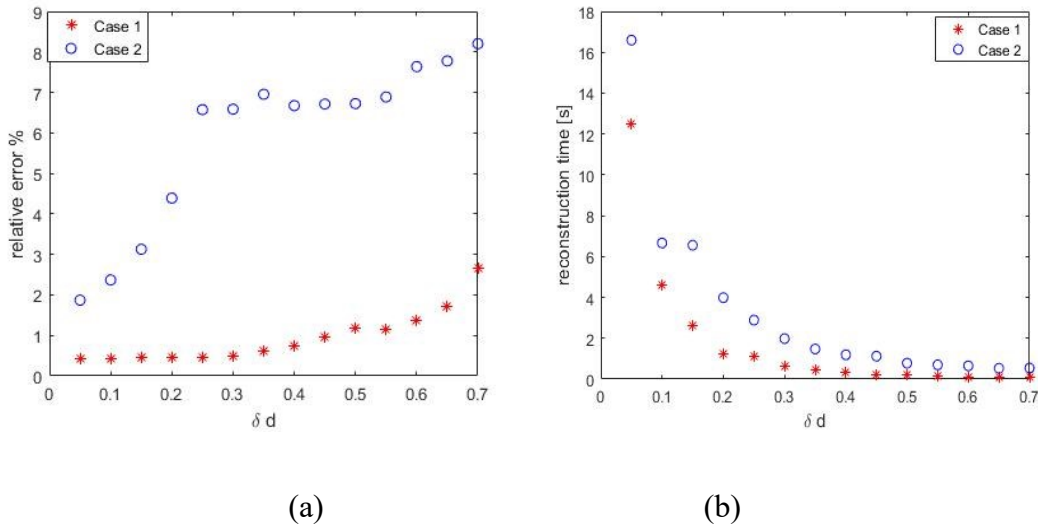
Figure 3-1 . (a) reconstruction error vs relative spacing for Case 1, (b) reconstruction time vs relative spacing for Case 2

3.6.1   Case 1: *in silico* Pulsatile Womersley flow

Fig. 3.2 shows the mean flow rate change in a cardiac cycle from reconstruction and reference, where the average relative reconstruction error calculated by Eq. 2.27 is 0.82% and $\varepsilon = 0.1$. The accuracy is higher than Chapter 2, because: 1. The shape parameter has been optimized; 2. a different regularization method, i.e., early-stopping, is used which is free of any unphysical assumptions (as mentioned in Chapter 2 the L-curve method assumes the input noise is white and this assumption may not be true). Fig. 3.3 shows the velocity magnitude distribution at the central cutting plane at 3 different time steps. The error near the wall is small which indicates successful suppression of RP. The very high accuracy implies that for simple flows, e.g., Womersley flow, the error caused by model simplification is negligible. The third row in Fig. 3.3 shows the error distribution is subject to sampling

location of the fictitious imaging plane, which indicates that lack of sampling may be the main error source in Case 1.
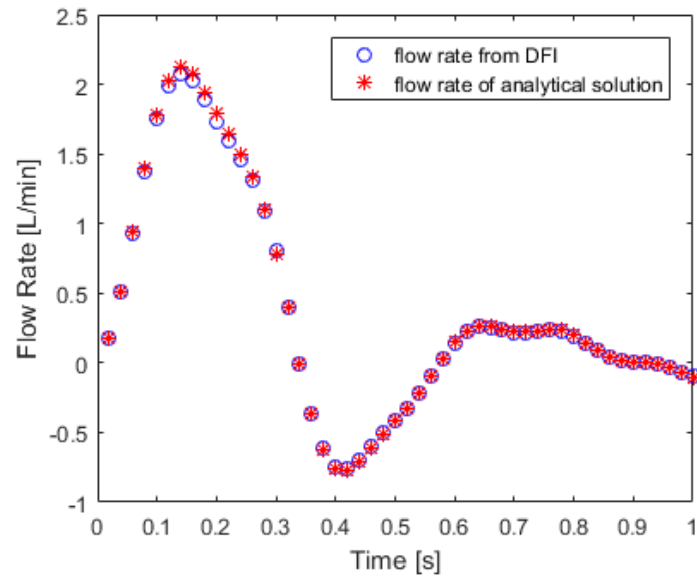
Figure 3-2 Flow rate at 50 different times in a full cardiac cycle
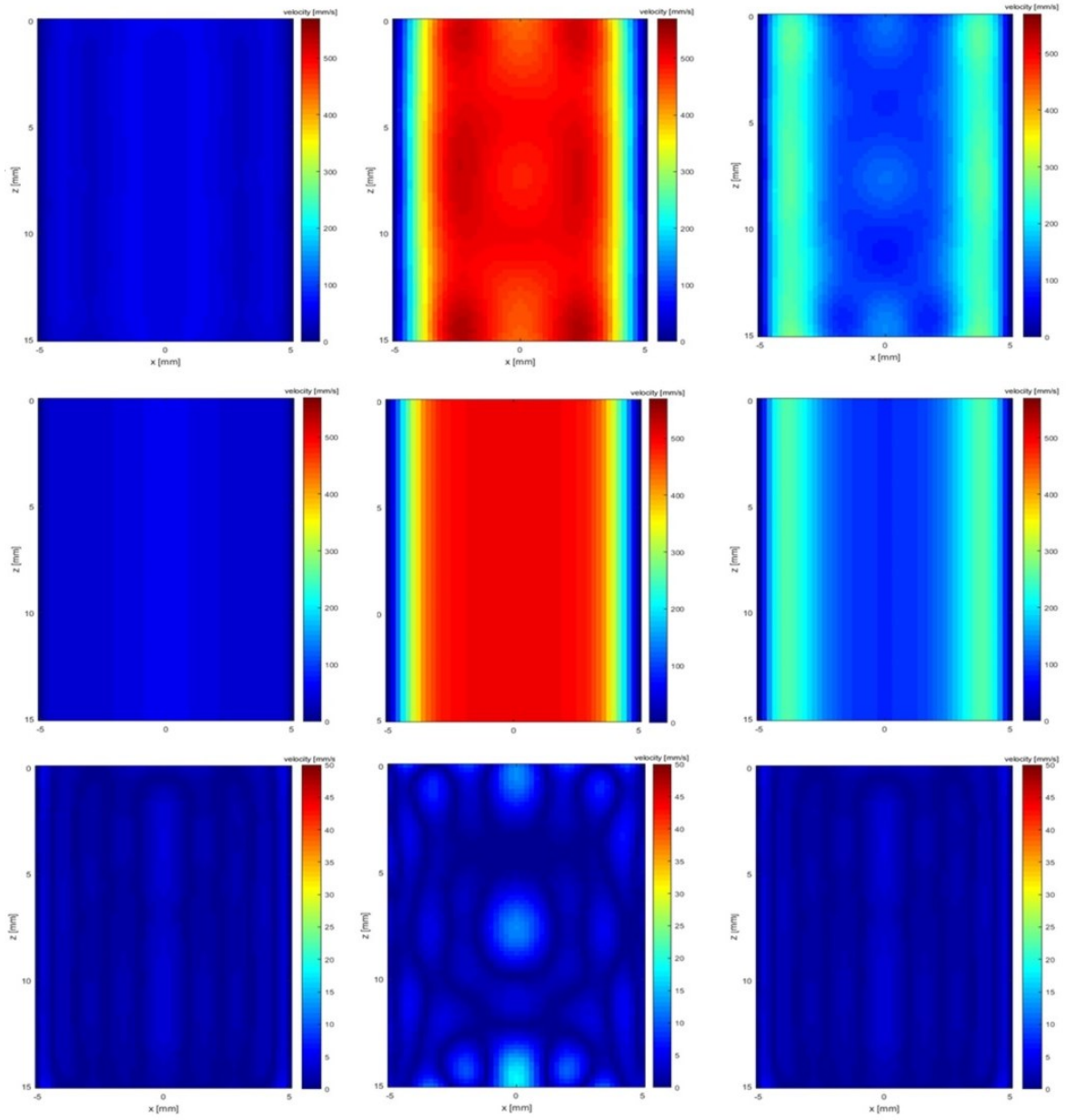
Figure 3-3 velocity magnitude distribution at different time. First row: reconstructed flow; second row: analytical solution; third row: the absolute difference between first and second row. Different columns are Womersley flow velocity distribution at different time (t=0 s, 0.25s, 0.6s).

3.6.2   Case 2: *in silico* Steady helical flow

Fig. 3.4 shows the reconstructed 3D flow from projected 2D CFD data and CFD results, and

the error of reconstructed 3D flow is 4.8%. We set $\varepsilon = 0.08$.

reconstructed velocity

()

velocity from CFD
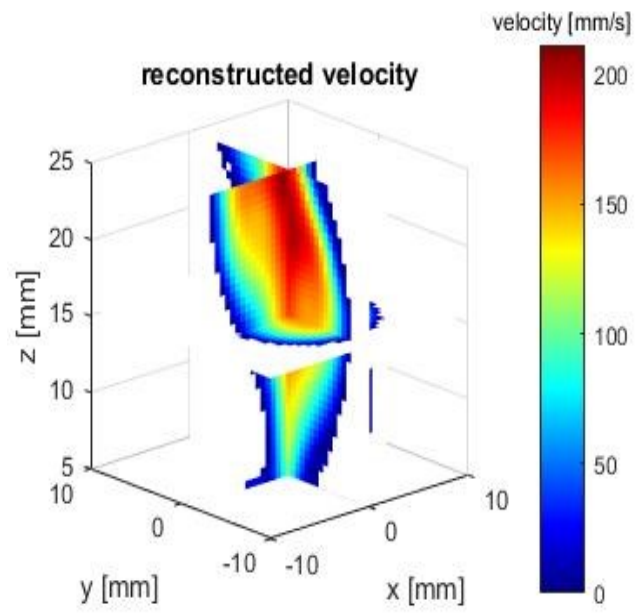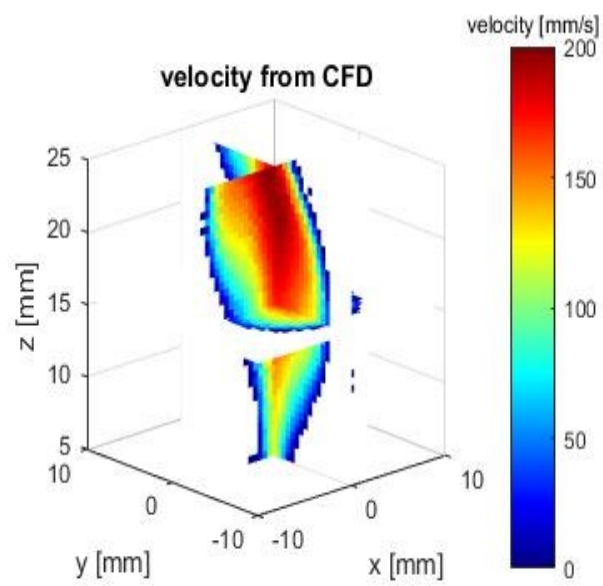
(b)
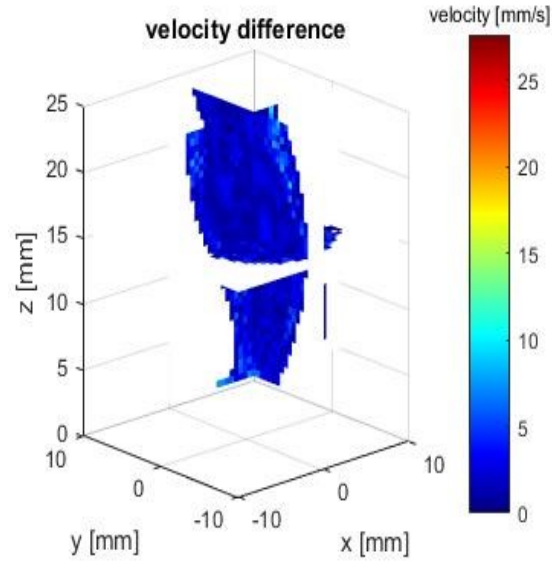
(c)

Figure 3-4 3D flow velocity magnitude: (a) reconstructed from simulation data; (b) CFD; (c) difference between (a) and (b)

3.6.3 Case 3: *in vitro* ultrasound measurement of steady flow in straight tube

Fig. 3.5 shows the reconstructed 3D flow of straight vessel from UIV, and the average relative error of reconstruction is 6.01%. $\varepsilon = 0.08$. The error is much higher compared with the *in silico* Womersley flow case, which indicates that measurement error (including UIV error, probe registration error, temporal registration error etc) in Case 3 may be the main error source.
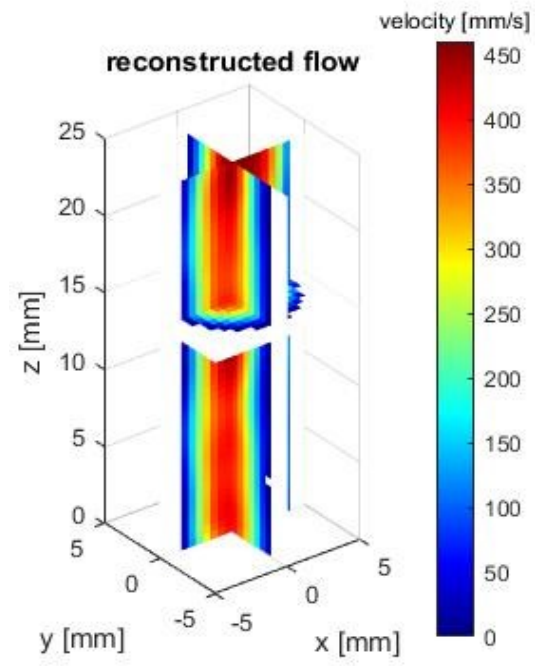
(a)
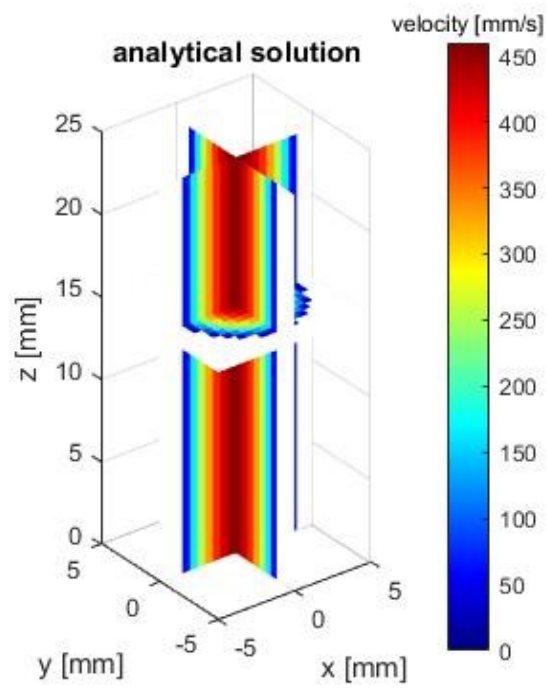
(b)
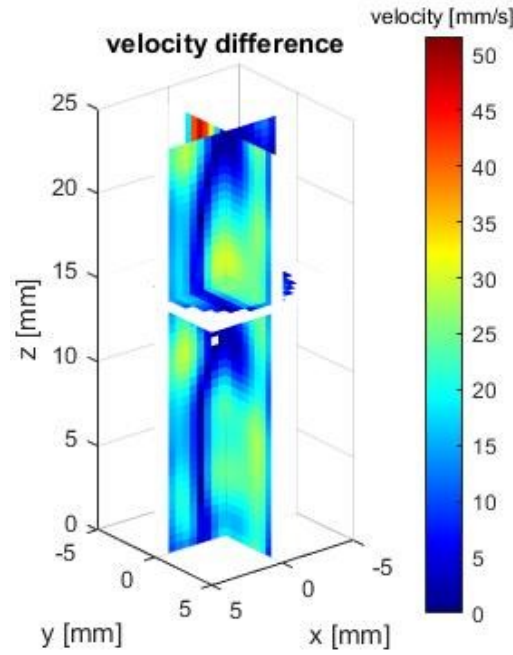
(c)

Figure 3-5 3D flow velocity magnitude: (a) reconstructed from UIV; (b) analytical solution; (c) difference between (a) and (b)

3.6.4 Case 4: *in vitro* ultrasound measurement of steady flow in bifurcation phantom

Fig. 3-6 shows the 3D flow reconstructed from UIV and CFD, with an overall error of 12.8%.

$\varepsilon = 0.12$. CFD is conducted with STAR-CCM+ by using inlet mass flow rate and outlet mass split ratio as the boundary condition with a Laminar incompressible solver. The inlet/outlet flow rates are measured by containers collecting fluids at the two outlets of the bifurcation. The results show that the method matches well with CFD simulation, and prove that method has a high accuracy for *in vitro* reconstruction.

73
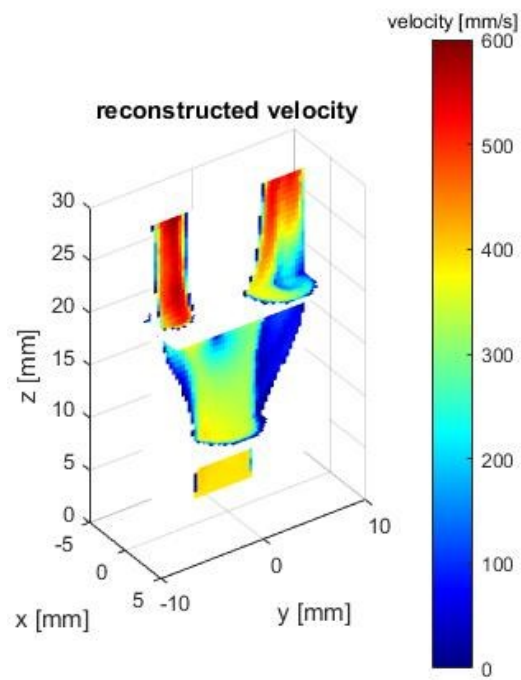
(a)
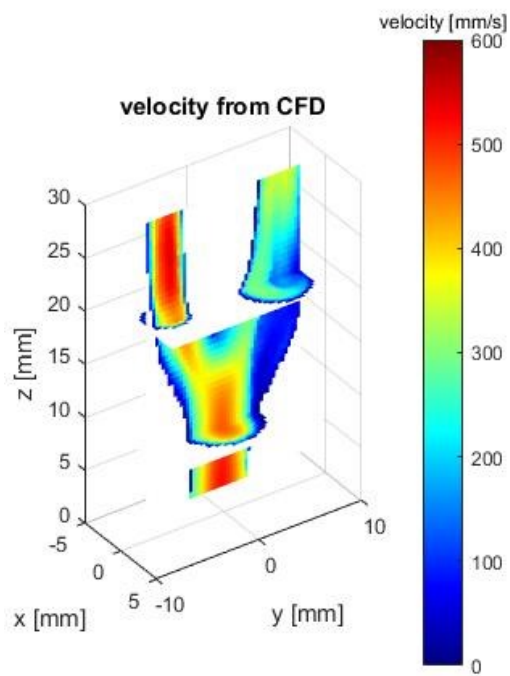
(b)
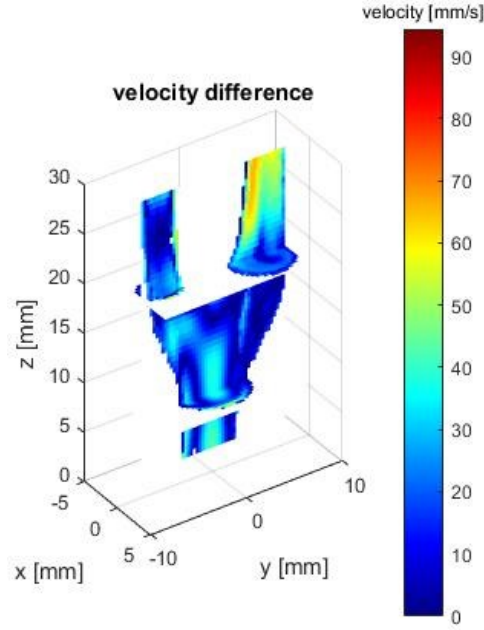
(c)

Figure 3-6 3D flow velocity magnitude: (a) reconstruction from UIV; (b) from CFD; (c) difference

3.6.5   Comparison of 3D reconstruction using 2D US VFI with previous work

To evaluate the improvements of reconstruction accuracy and time-to-solution over the previous TSVD method in Chapter 2, 3D flows using 2D velocities of the four cases are constructed separately using the method in (195) with the identical 2D in-plane velocity input and compared in Table 3-1. To make fair comparison we use the sparse SVD solver in Matlab for TSVD. The results show that with GMRES the optimized UADFI makes the reconstruction 45~728 fold faster than using Matlab sparse TSVD solver (195). In addition because of the shape parameter optimization using the holdout validation and new regularization method, the reconstruction accuracy is significantly improved.

Table 3-1 also shows that Case 3&4 (*in vitro* cases) have lower accuracy than Case 1&2 (simulated cases), but with shorter time-to-solution. This is because velocity samples from Case 1&2 are free of measurement noise and registration error, the iteration tolerance ζ is

much smaller compared with Case 1&2, so that convergence of iteration is achieved with larger number of iterations and thus is faster than Case 1&2. In this chapter Cases 3&4 stop with <50 iterations, while Case1&2 stops with ~100 iterations.

Table 3-1 Improvement of reconstruction time and accuracy over the existing method

	Reconstruction time (s)			Relative error		
	Previous	Optimized	Improvement	Previous	Optimized	Improvement
Case 1 (m=2043)	56.63	1.24	45.67 fold	3.22%	0.82%	74.53%
Case 2 (m=3259)	174.55	3.91	44.64 fold	8.20%	4.80%	41.46%
Case 3 (m=3392)	191.84	1.06	180.98 fold	10.10%	6.01%	40.5%
Case 4 (m=5487)	575.25	0.79	728.16 fold	15.10%	12.8%	15.23%

3.7. Summary

In Chapter 3 the shape parameter of the divergence free basis functions is optimized by the holdout validation, and the UADFI system is solved by the modified sparse break-down free GMRES. The results from both simulation and phantom studies show an improvement of 15%-75% in reconstruction accuracy and 45-728 fold acceleration in speed when comparing to the TSVD method used sparse SVD solver (195). Then the results show higher accuracy (15%-74% improvement).

The error of reconstruction in this work may be caused by several factors: 1) Model simplification, i.e., momentum conservation is not considered in DFI; 2) Dependence on experimental sampling. Fig. 3-1 shows that error increases with increased relative spacing δd; 3) Registration error due to stage motion, i.e., the probe stepping is controlled by a stage which may move out of alignment with the original position after scanning for the first angle; 4) Temporal registration; 5) Limitations in the measurement accuracy of 2D VFI due to US

clutter, signal noise and lost track etc. For UIV accuracy of tracking complex flow with localized vortex may decrease due to lost track, e.g., for flow at post-stenosis. Lost track is a result of averaging complex flow in a finite UIV interrogation window (184), and cross correlation fails to track fast flows with a small interrogation window. 6). Segmentation error. Tissue/phantom may be falsely segmented as fluid and vice versa. On the other hand it should be noted that CFD, here used as reference for Case 2&4, also has errors due to input uncertainty and representativeness errors. Representativeness error is induced in STAR-CCM+ when 3D velocities at the CFD grids are interpolated to the fictitious imaging planes.

Methods to increase accuracy include: 1). Increasing sampling by reducing the relative spacing δd, at the cost of increasing US acquisitions and reconstruction time. However the reconstruction is a trade-off between acquisition cost and accuracy, δd is selected according to the in-plane resolution and the desired accuracy, and $\delta d = 0.2$ is a suggested choice for real blood flow reconstruction; 2) Building a more accurate stage motion system; 3). Reducing the error of HFR UIV by locally adjusting the size of interrogation window and reducing US grating lobe; 4). Segmenting the flow domain using more accurate algorithms, e.g., deep neural network (45). Reconstruction for *in vivo* may suffer from reduced accuracy due to less accurate input caused by clutter, larger noise and registration error, however the solution of Eq. 2.19 is a "least square error solution with regularization" where regularization is used to increase the robustness to input errors against ill-posedness. In addition by imposing the physical divergence free condition the interpolation has the capability of filter out input noise. With reasonable input containing acceptable input error, we have to adjust the regularization parameter ζ and the solution is still stable and reasonable (but the accuracy will decrease). With unacceptable input error level from ultrasound acquisition, we cannot count on reconstruction only and have to increase the input accuracy.

Chapter 3 is based on data acquired with the presence of microbubbles which significantly increase echoes from within the blood by RBC alone. While the use of microbubbles increases the measurement SNR and generate better velocity field data, the injection of microbubbles may limit the wider use of the technique. However in principle the 3D flow reconstruction approach proposed in this chapter can use velocity field samples generated by other experimental methods, including e.g. Doppler measurements without microbubbles although the reduced SNR makes it more challenging. These need further *in vitro* and *in vivo* validation.

Comparing to our previous method in Chapter 2, Chapter 3 provides a more accurate and computationally more efficient way (by using highly optimized GMRES and increasing the sparsity of the resulting system) for estimating 3D flow field and can be potentially used for artery volume flow measurements. These make it a potentially suitable tool for clinical measurements of intravascular flow, and in the detection of heart failure, renal failure and atherosclerosis related diseases which are closely correlated with 3D blood flow patterns. The results of 3D reconstruction with 1D velocity components acquired by a matrix probe or tilting a linear probe also show clinical potential of UADFI for intraventricular flow reconstruction.

Chapter 4 Ultrasound VFI Augmented LBM Flow Simulation

4.1. Introduction

Chapters 2&3 introduce the UADFI, which fast reconstructs 3D full flow field from sparse 3D distribution of 2D measurements. Although it is expected to have high accuracy as it reconstructs flow from real measurements incorporating the divergence free condition, the method requires 3D distribution of US VFI measurements, and its accuracy is largely affected by the sampling density and input accuracy. To obtain high accuracy to fully resolve complex flows, high US measurement sampling density is required but is expensive.

Alternatively patient specific CFD obtains full flow information with lower cost but is limited by high computational overheads and compromised accuracy due to: 1. CFD input uncertainty/error (especially in initial/boundary condition). Blood CFD errors include coding/user errors (incorrect use of code and mistakes/bugs of code), numerical errors (round-off error, iterative convergence errors, discretization errors, numerical diffusion/dispersion error), image registration error (during boundary condition and geometry acquisition using MRI/US/CT). Blood CFD uncertainties are caused by lack of knowledge due to the complexity of blood flow, including: incomplete information on Boundary condition (BC) or Initial Condition (IC), discrepancy between real and CFD geometries (206), pulsatile artery motion; incomplete information on blood properties (e.g., non-Newtonian rheology); physical model uncertainty caused by negligence of turbulence, unsteadiness, compressibility, and chemical reactions. 2. uncertainties/error propagation and amplification as a result of inherent CFD nonlinearity. CFD involves strong nonlinearity and its accuracy is sensitive to input uncertainty/error, and thus has a lower fidelity than its input (e.g., boundary condition) from which uncertainty/error propagates and amplifies.

On the other hand, measurements within the flow domain acquired by US VFI (or MRI) only contain measurement/registration error, and with the current advances in medical US highly accurate and widely accessible flow measurements are available (the US VFI method used in this study has a validated *in vivo* accuracy of >90%). This study assumes the internal measurements and BCs have similar levels of fidelity, while the fidelity of CFD simulation using such BCs reduces due to many other uncertainty sources and nonlinearity. Previously the internal measurements are either used to validate CFD accuracy, or for data assimilation to reduce the impact of input uncertainties/errors. Measurement assimilated CFD combines experimental observations with CFD, based on inversion modelling where an iterative Ensemble Kalman method is used to infer and refine uncertain model parameters in CFD input (e.g., uncertainty in BCs) and calibrate numerical models (207-211). Although these methods have higher fidelity by assimilating real measurements, three problems exist: 1. Data assimilation are more computationally expensive than conventional CFD because a successive set of CFD simulations are required until an accepted match between CFD and measurements is reached; 2. Blood CFD involves very complicated error/uncertainty sources. To refine such a large parameter space (for example geometry, BCs, ICs, viscosity, CFD models all contain uncertainties), a large number of assimilated measurements are required for high fidelity, which restricts its application in blood flow CFD; 3. Such methods refine CFD input (e.g. BCs) with internal measurements, assuming that internal flow measurements have higher fidelity over CFD input (for example data assimilation refines inlet BC while keeping measurements fixed) which may not be true. Based on the assumption that BCs and internal measurements have similar level of fidelity, the flow information arriving at the BCs and the internal measurements during CFD simulation has lower fidelity than local information from measurements as a result of CFD nonlinearity and many other uncertainty

sources. As such it is not reasonable to refine one while keeping another fixed (for example measurement assimilation usually refines BCs using internal measurements).

Based on the assumption in this study it is reasonable to treat BCs and internal measurements equally, rather than refining one using the other. This study proposed a direct measurement augmented Lattice Boltzmann Method (LBM) CFD algorithm, where internal measurements are used as internal "BCs" to constrain CFD. Although such internal "BCs" are inconsistent with the CFD simulation obtained with only normal BCs, this is equivalent to physically setting the flow particles at the imaging plane to move at the measured velocities, and such particles impose a viscosity force to nearby flow particles (the non-slip condition). This is a physically consistent and will converge, although not consistent with the original CFD.

Lattice Boltzmann Method (LBM) is a relatively new CFD method and has some advantageous properties including easy implementation of complex geometries and full parallelization. By adopting the CUDA parallel computing power LBM has the capability of conducting blood flow simulation in several seconds. In this chapter a GPU version LBM code is first developed, and then we propose a direct measurement augmented LBM CFD algorithm. Different from the previous data assimilation method which corrects CFD parameters iteratively, this study is the first application of direct measurements integration to CFD to the best of our knowledge. As such it is equivalent to setting internal "BCs" to reduce the local and overall CFD uncertainty. Our method has the advantage of suppressing various CFD uncertainty errors and speeding up convergence compared with conventional LBM, while fully utilizes the parallel computing power of GPU.

The merits of the proposed approach were evaluated in two *in silico* benchmarks, i.e., lid driven square cavity flow and fully developed circular Poiseulle flow, and experimentally on a steady-state *in vitro* carotid bifurcation phantom. For *in silico* cases and *in vitro* case 3D

and 2D velocities extracted from reference and US VFI are used respectively to augment CFD with perturbed BCs. The results show the method can speed up LBM convergence and reduce LBM error significantly due to BC error when measurements have higher fidelity than the compromised CFD. The two *in silico* cases show up to an order of magnitude and more than threefold faster using velocities from seven planes, while using velocities from one plane the overall error reduced from 15.59% to 10.41% and from 14.86% to 5.57%. The *in vitro* results show ~27% acceleration using 2D components of velocities from one plane, and an overall error reduced from 15.90% to 12.17% and from 13.09% to 11.36%. The results indicate the potential clinical application of the proposed method.

4.2. LBM-BGK

LBM has comparable accuracy with other CFD algorithms (e.g., Finite Element/Volume Method) and is easy for GPU parallelization and complex geometry handling. This section briefly introduces the standard D3Q19 Bhatnagar-Gross-Krook (BGK) LBM model (212) for 3D flow simulation, then the data augmentation method, followed by selection of regularization/relaxation parameter. Three testing cases are introduced in the end.

In BGK-LBM model the 3D flow at each lattice location is represented by particle probability density function (pdf) f which evolves by:

$$\frac{Df}{Dt} = \partial_t f + (\vec{\xi} \cdot \nabla)f = \frac{f - f^0}{\tau} \tag{4.1}$$

where $\vec{\xi}$ is the lattice velocity, τ is the relaxation time. f^0 is the Maxwell-Boltzmann equilibrium pdf. The macroscopic properties $\vec{u}$ and ρ (3D velocity and density) are given by:

$$\rho(\vec{x}, t) = \int f(\vec{x}, \vec{\xi}, t)d\vec{\xi} \tag{4.2}$$

$$\rho(\vec{x}, t)\vec{u}(\vec{x}, t) = \int \vec{\xi}f(\vec{x}, \vec{\xi}, t)d\vec{\xi} \tag{4.3}$$

Using the D3Q19 BGK-LBM model the velocity space is discretized into 19 components and Eq. 4.1 becomes:

$$f_i(\vec{x} + \vec{c}_i dt, t + dt) - f_i(\vec{x}, t) = -\frac{1}{\tau}\left(f_i - f_i^0\right), i \in [0:18] \qquad (4.4)$$

where the equilibrium pdf f_i^0 is:

$$f_i^0 = w_i \rho[1 + 3\vec{c}_i \cdot \vec{u} + 4.5(\vec{c}_i \cdot \vec{u})^2 - 1.5\vec{u} \cdot \vec{u}] \qquad (4.5)$$

The discretisation of velocity at each computational cell using the D3Q19 model is illustrated in Fig. 4-1, where the velocity of the artificial particle is discretised to 19 velocity components weighted by w_i.
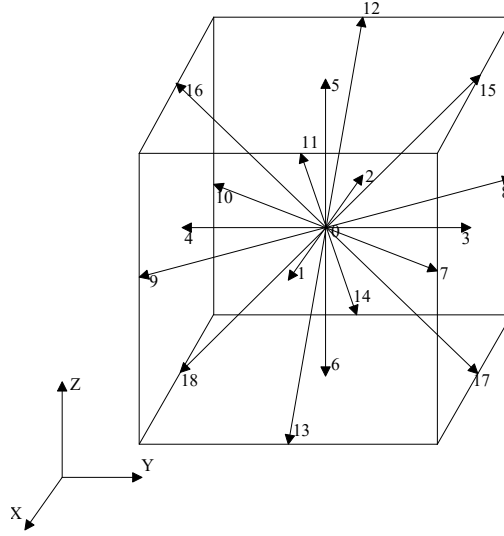
Figure 4-1 Lattice cell with 19 discrete directions in D3Q19 model

$\vec{u}$ and ρ are macroscopic properties (3D velocity and density) of flows and are updated in each time-step by:

$$\rho = \sum_{i=0}^{18} f_i \qquad (4.6)$$

$$\rho\vec{u} = \sum_{i=0}^{18} \vec{c}_i f_i \qquad (4.7)$$

and $\vec{c}_i$ is the discretised lattice speed. The weighting factors w_i are:

$$w_i = \begin{cases} \frac{1}{3}, & i = 0 \\ \frac{1}{18}, & i = 1, 2, 3, 4, 5, 6 \\ \frac{1}{36}, & i = 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18 \end{cases} \qquad (4.8)$$

$\vec{c}_i = 0$ for i=0, $\vec{c}_i = (\pm 1, 0, 0)$ for i=1:2, $\vec{c}_i = (0, \pm 1, 0)$ for i=3:4, $\vec{c}_i = (0, 0, \pm 1)$ for i=5:6, $\vec{c}_i = (\pm 1, \pm 1, 0)$ for i=7:10, $\vec{c}_i = (\pm 1, 0, \pm 1)$ for i=11:14, $\vec{c}_i = (0, \pm 1, \pm 1)$ for i=15:18. The kinematic shear viscosity is:

$$\nu = c_s^2 \left(\tau - \frac{1}{2} \right) \qquad (4.9)$$

Eq. 4.4 shows that after collision the pdf returns to the local equilibrium counterpart f_i^0 which is a function of macroscopic properties $\vec{u}$ and ρ. The macroscopic properties in the flow domain are then updated iteratively, with information propagating from the boundary into the interior of the flow domain at the lattice speed $\vec{c}_i$ until convergence. Eq. 4.4 can be split into collision and streaming:

A) Collision:

$$\tilde{f}_i(\vec{x}, t) = f_i(\vec{x}, t) - \frac{1}{\tau}\left(f_i - f_i^0\right) \qquad (4.10)$$

B) Streaming:

$$f_i(\vec{x} + \vec{c}_i dt, t + dt) = \tilde{f}_i(\vec{x}, t) \qquad (4.11)$$

where $\tilde{f}_i$ is the post-collision pdf. Collision and streaming happen at all fluid lattices in each time-step until convergence. The handling of BCs used in this study is the hall-way bounce back to specify the velocity of vessel walls (i.e., artificial LBM particles bounce back at the vessel walls, but the vessel velocity is either static or obtained from measurements), and non-equilibrium extrapolation for velocity and pressure boundary (213). Body force is not considered for simplicity.

4.3. Ultrasound Augmented LBM

The LBM simulates the streaming and collision of particle pdfs, while measurements obtain macroscopic properties (pressure, velocity). Eqs. 4.6 and 4.7 show the link between the macroscopic parameters and the pdf. To translate the velocity or pressure measurements into the LBM language, measurements are fed into Eq. 4.5 to obtain the equilibrium pdf. In LBM pressure difference δp is first transformed to a density difference $\delta\rho$ by:

$$\delta p = \delta\rho c_s{}^2 \tag{4.12}$$

At the two "BCs" two equilibrium pdfs f_i^0 can be obtained: f_{1i}^0 using macroscopic properties from measurements (with partial flow information, e.g., 1D or 2D measurement, only the prior known information is used), and f_{2i}^0 using macroscopic properties updated using flow information propagating from adjacent computing cells using Eqs. 4.10 and 4.11. The final local f_i^0 is calculated by a linear combination of f_{1i}^0 and f_{2i}^0 weighted by a relaxation parameter α ($0 \leq \alpha \leq 1$):

$$f_i^0 = (1 - \alpha)f_{1i}^0 + \alpha f_{2i}^0 \tag{4.13}$$

α quantifies the relative relaxation between flow information propagating from the other "BC" and local measurements. Large α ($\alpha \rightarrow 1$) is given when the measurements bear lower fidelity. $\alpha = 0$ means measurements are fully trusted and $\alpha = 1$ means no relaxation is utilized. f_{2i}^0 is updated in each time-step so that the convergence becomes slower with $\alpha \rightarrow 1$.

As explained in the introduction, this study assumes the two BCs have higher fidelity level than compromised CFD. f_{2i}^0 propagating from adjacent cells using CFD suffers from increased uncertainty due to nonlinearity and various CFD uncertainty, while f_{1i}^0 is from local measurements and only contains measurement/registration uncertainties. It is reasonable to set $\alpha = 0$ for both BCs, so that the internal "BCs" (augmentation data) and LBM BC are both fully trusted, over the local flow information propagating to them. In this way two benefits can be obtained: 1. Convergence is faster. Flow information propagates away from both BCs

to the spatial gap between them until the full domain is filled, leading to faster information propagation and thus convergence; 2. In flow domain between the two "BCs", the local and thus overall uncertainty/error can be suppressed by higher fidelity information propagating from the two "BCs".

4.4.   GPU architecture and programming

General Purpose GPU (GPGPU) is an energy efficient and cost-effective parallel many-core computational hardware accelerator, and can accelerate the computational time for a suitable problem. Compared with CPU which dedicates a lot of transistors to perform data caching and control operations, GPU has a large number of transistors dedicated to arithmetic logic units (ALUS), as illustrated in Fig. 4-2.

(a) CPU	(b) GPU
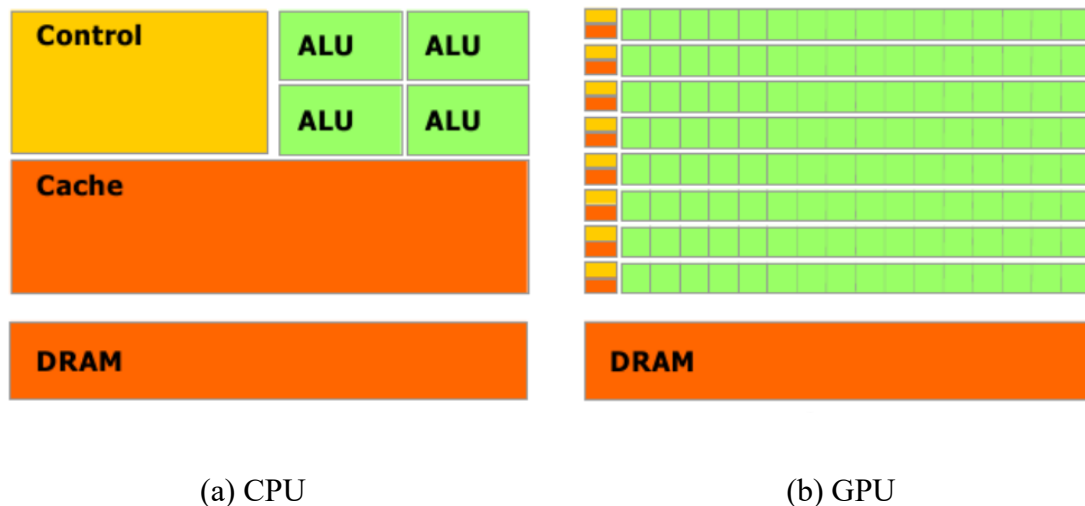
Figure 4-2 CPU and GPU transistor allocation

GPU comprises multiple Streaming Multiprocessors (SMs), which execute threads in parallel groups of 32 threads, called a warp. To allow GPU for general purpose processing, Compute Unified Device Architecture (CUDA) is developed by NVIDIA as a parallel computing platform and application programming interface, which gives direct access to GPU's virtual instruction set and parallel computational elements. CUDA provides a compiler called nvcc,

and is accessible to software developers through CUDA-accelerated libraries, compiler directives, and industry-standard programming languages including C, C++ and Fortran.

GPU memory is managed in a hierarchical scheme, with each thread having a local memory and each block having the shared memory. All threads have access to the global memory (also called the device memory), which is large but has a longer access latency. Two other types of read-only memories, i.e., constant memory and texture memory, are also available. Constant memory and texture memory use a cache for quick access, increasing the GPU performance in reading data.

To develop a piece of software with CUDA, both CPU and GPU are used while CPU transfers data to/from GPU, and call CUDA kernel functions. Kernel functions run on GPU, using the Single Instruction Multiple Data (SIMD) scheme, while each thread executes the same instruction directed by the common Instruction Unit on its own data streaming from the device memory to the on-chip cache memories and registers. Each data element occupies a CUDA thread. To optimize a GPU code different memories and data structures should be used, and the GPU throughput should be used as much as possible, while memory transfer, warp divergence, uncoalesced memory access should be minimized to fully utilize the parallel computing capability of GPUs.

The GPU used in this study is a Nvidia Geforce GTX 1050ti, with a base clock of 1,290MHz (boost 1,392MHz). It has two Graphics Processing Clusters (GPCs) subdivided into six Streaming Multiprocessors (SMs) for a total of 768 CUDA cores and 48 texture units. The render output unit count stands at 32, with eight each assigned to one of the four 32-bit memory controllers (128-bit interface in total) and a 256KB slice of L2 cache (1MB total). The GPU ships with 4GB of GDDR5 for a total memory bandwidth of 112 GB/sec. Single precision floating point algorithm is used throughout this chapter.

4.5. Parallel version LBM

LBM is data intensive and memory bound, and relies on large and expensive computational clusters. As GPU performs significantly more floating point operations per unit time than CPUs, with GPU the number of parallel cores increases at slightly increased price. This tendency may bring down the cost of parallel LBM. To fit the LBM algorithm to gain benefits from computational possibilities with the GPU hardware, Section 4.5 introduces the implementation of a Newtonian fluid LBM code with GPU, by the use of the NVIDIA CUDA C++ language.

A) kernel functions

Two kernels, i.e., collision_streaming_kernel for the streaming and collision operations, and the boundary_kernel for the boundary condition processing. In each time-step three procedures take place: 1). For each computational grid within flow domain, the collision_streaming_kernel loads the pdf at the neighbour cells from the source_grid, conducts the collision and updates the density/velocity, and writes the post-collision pdfs to dest_grid; 2) For each computational grid located on the flow boundary the boundary_kernel update the pdfs, including inlet, outlet and wall; 3) the pointers of source_grid and dest_grid are swapped for the next time-step.

B) index transformation

GPU has a limited global memory and each computational cell requires Q+4 floating variables, where Q is 19 for D3Q19 model and 4 accounts for three velocity components and the density. To reduce the total number of computational cells we conduct the index transformation in CPU first. Index transformation maps each GPU thread arranged in structured manner to a LBM grid in an arbitrary geometry, using a lookup table. This reduces memory usage for flows as the flow domain is usually not rectangular. Index transformation is performed in CPU before initialization. After index transformation the total number of

computational cells is the total number of fluid cells and boundary cells, rather than NX*NY*NZ where NX, NY and NZ are number of grids in x, y and z coordinates. Each flow and boundary cell has an index to map it from the location in the source_grid and dest_grid to the thread index, and the look-up table is stored in an 1D array of index_grid.

C) data organization

When a running group of threads (or WARP) encounters a cache miss, the threads are delayed for the next few hundred cycles. This is termed uncoalesced memory access. The LBM streaming phase incurs a lot of uncoalesced GPU memory access, which imposes a negative impact on the overall performance.

Two 1D arrays are used to represent the pdfs before and after collision, i.e., source_grid and dest_grid, as shown in Fig. 4-3. To alleviate the problem of uncoalesced memory access, we use the Array of Structure (AOS) scheme, where the same pdf of all computation cells occupy consecutive elements of the 1D array. By using the AOS scheme different threads access consecutive memory elements simultaneously, which results in reduced uncoalesced memory access. Similarly the 1D arrays density_grid and velocity_grid are also organized by the AOS scheme.
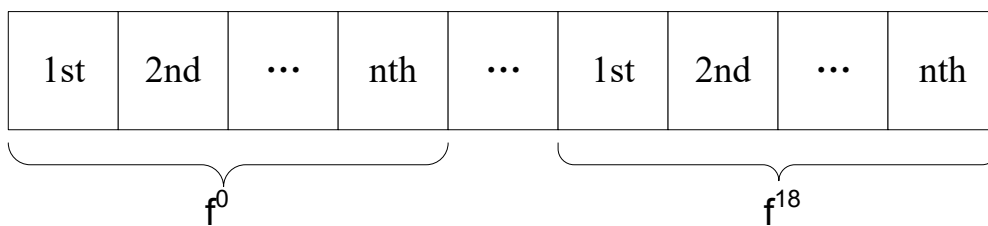
Figure 4-3 AOS Scheme

D) neighbour grid data access scheme

During the streaming phase information (i.e., pdfs) at the neighbour grids needs to be stored in source_grid and then passed to each grid. Uncoalesced writing to device memory is much slower than uncoalesced reading, and thus a pull scheme is used (214), i.e., during the

streaming phase the data from the adjacent 18 cells (if D3Q19 model is used) is pulled to the current cell before collision happens, and only the local post-collision pdfs in dest_grid are updated.

E) texture memory

The texture memory is a special memory spaces residing in device memory and are read-only. It reduces memory latency (i.e., the data transfer time from memory) when the data to be fetched have spatial locality. In LBM some variables are read only and to speed up memory access, we store them in the texture memory to speed up memory access. An example is the inlet/outlet BC and augmentation data, if steady flow simulation is conducted.

Other tricks to optimize the CUDA version LBM include reduction of register use, i.e., overuse the local memory as much as possible, and avoid warp divergence.

In summary, several 1D arrays are used in different memory, where m is the number of velocities acquired by US VFI, NLATTICE is the number of total computation cells after index transformation and $N \leq NX * NY * NZ$:

- index_grid: the lookup table to map each GPU thread in structured manner to a LBM grid in an arbitrary geometry. It is a 1D array with $NX * NY * NZ$ elements, stored in texture memory, and its value is between 0 and $NLATTICE - 1$;
- source_grid: a 1D array with NLATTICE *19 elements, stored in global memory, used to store the pdfs of all computation cells before collision
- dest_grid: a 1D array with NLATTICE *19 elements, stored in global memory, used to store the pdfs of all computation cells after collision
- density_grid: a 1D array with NLATTICE elements, stored in global memory, used to store the density of all computation cells after collision
- velocity_grid: a 1D array with NLATTICE *3 elements, stored in global memory, used to store the three velocity components of all computation cells after collision
- vfi_array: a 1D array with m*2 elements (if VFI is 2D US, and m is the number of measured velocity vectors), stored in texture memory (for steady flow) or global

memory (for unsteady flow), used to store the two velocity components of velocities acquired by 2D UIV

- inlet_array: a 1D array with mi*2 elements (mi is the number of measured velocity vectors at the inlet), stored in texture memory (for steady flow), or global memory (for unsteady flow), used to store the two velocity components of velocities acquired by 2D UIV

- outlet_array: a 1D array with mo*2 elements (mo is the number of measured velocity vectors at the outlet) or a scalar (if pressure outlet is given), stored in texture memory (for steady), or global memory (for unsteady flow), used to store the two velocity components of velocities acquired by 2D UIV

- label_array: a 1D array with NLATTICE elements, stored in texture memory (if vessel lumen is static) or global memory (if vessel lumen is changing), used to label each computational grid with unsigned integers.

4.6. Error and residual metrics

To check convergence the residual was defined as:

$$\mathbf{Err} = \frac{\left\| \sqrt{\sum u_{i,t+1}^2 + v_{i,t+1}^2 + w_{i,t+1}^2} - \sqrt{\sum u_{i,t}^2 + v_{i,t}^2 + w_{i,t}^2} \right\|}{\sqrt{\sum u_{i,t+1}^2 + v_{i,t+1}^2 + w_{i,t+1}^2}}$$ (4.14)

$u_{i,t}$, $v_{i,t}$ and $w_{i,t}$ are velocity components at the t-th time-step. The error metric is the same with Eq. 2.27.

4.7. Workflow of GPU based LBM

The overall workflow of GPU version LBM consists of the steps shown in Fig. 4-4. The geometry preprocessing includes four steps: 1) segment each US B-mode images from geometry scan, in order to obtain the scattered surface point cloud; 2) reconstruct surface with triangulated mesh using a crust based method (215) and smooth the surface using a curvature flow smoothing (216); 3) generate a set of uniform computational cells, and label each cell using the alpha shape algorithm (217); 4) conduct distance transformation on the binary 3D geometry generated in Step 3, so that different cells in inlet, outlet, flow domain,

wall, imaging plane, and outside of the flow domain are labelled with different integers, e.g., outside-0, wall-1, inlet-2, outlet-3, flow domain-4, augmented locations-5.
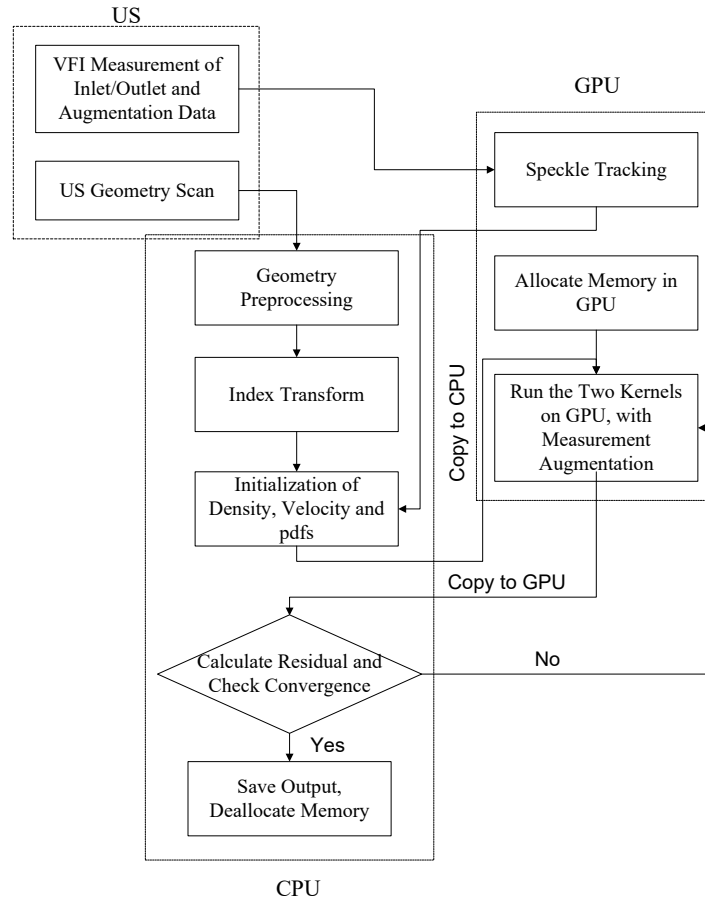
Figure 4-4 Flowchart of parallel version LBM

4.8. Testing cases

First the proposed US augmented LBM method is tested on two steady-state *in silico* benchmark cases (lid driven cavity flow and circular Poiseulle flow) and then on a steady-state *in vitro* case (carotid bifurcation phantom flow). For *in silico* cases we first simulate the flow using conventional LBM as reference, and extract 3D velocities at artificial 2D imaging planes from the reference as augmentation data. For the *in vitro* case the augmentation data is 2D velocities obtained by interpolating 2D ultrafast UIV velocity data to LBM lattices using the nearest interpolation, and the reference is LBM simulation with BCs acquired by measurements. After obtaining the reference and augmentation data, we simulated the three

testing cases using measurement augmented LBM. For each testing case BCs with certain levels of artificial error are applied. The simulation results are compared with reference for accuracy evaluation. For simplicity the working fluids of the three cases are Newtonian and we only study steady flows.

A) *In silico* Newtonian fluid simulations

Case 1: steady flow in a square Lid Driven Cavity (LDC, 4mm-4mm-4mm) filled with the Newtonian blood. The right side of the cube slides constantly at 0.15 m/s (along y-axis), serving as the moving plane. Density and kinematic viscosity of blood are 1060 kg/m^3 and 3.2547×10^{-6} m^2/s, $Re = 222.2$. The total number of computational lattices is $262144 (64 \times 64 \times 64)$.

Case 2: steady fully developed Poiseulle flow of Newtonian blood in a straight circular vessel with diameter 4 mm and length 4 mm. The maximum velocity at the centre of the vessel is 0.15 m/s, $Re = 222.2$. The total number of computational lattices is 208632. The boundary conditions are parabolic velocity at both inlet and outlet, and atmospheric pressure at the outlet.

B) *In vitro* Newtonian fluid simulations

Case 3: steady Newtonian water flowing in a carotid bifurcation phantom. The inlet diameter is 7.6 mm and length of simulation is 20 mm. The maximum velocity at the inlet is ~0.5 m/s. $Re = 379.1$ and the total number of lattices is 65820. The geometry of the bifurcation phantom is shown in Fig. 4-5c. 2D in-plane velocity at the sagittal plane was acquired by HFR UIV for data augmentation. The volume flow rates at inlet/outlet are measured by flow volume collected by a container at the two outlets. Velocity profiles at the BCs are acquired from UIV and then normalized by the measured volume flow rate to ensure mass conservation.

We use UIV as the VFI method, and the US probe was placed to scan the sagittal plane of the flow phantom (XY plane, see Fig. 4-5c and the acquired B-mode image and tracked 2D velocities are shown in Fig. 4-6, and then registered into the LBM model for augmentation.

(a) (b) (c)
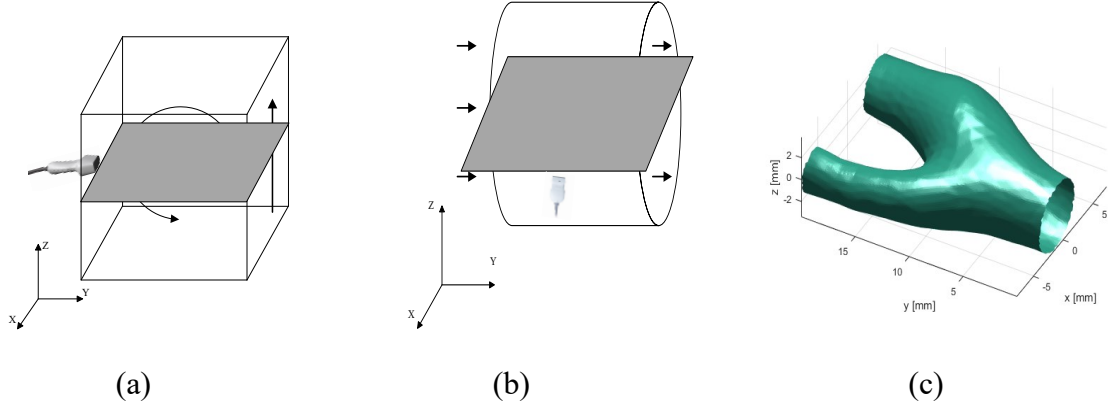
Figure 4-5 Geometry of three cases
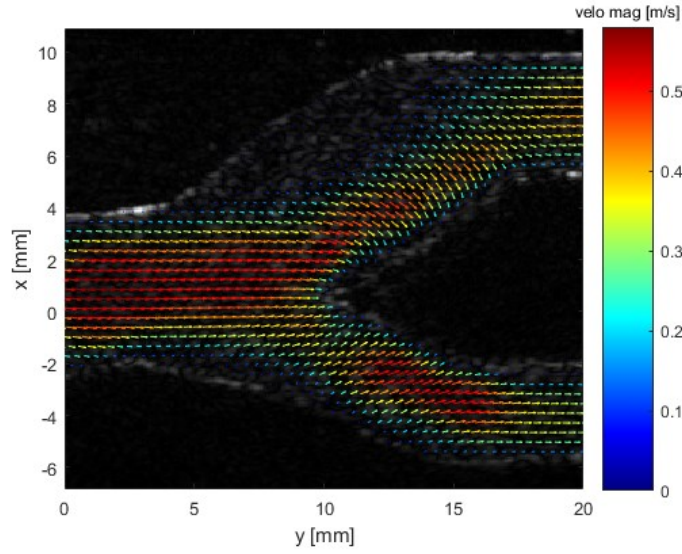
Figure 4-6 B-mode image and in plane velocity vectors acquired from 2D UIV on Case 3

4.9. Results

First we compare the efficiency of GPU version LBM with serial CPU version LBM, and the accuracy with the refence (STAR-CCM+ for Case 1 and analytical solution for Case 2). Then we aim to study two effects with data augmentation: A) convergence acceleration of LBM-BGK; B) LBM-BGK error suppression.

4.9.1. Comparison with serial implementation

The performance of CPU version and GPU version of LBM-BGK solvers were compared on Case 1 (computational domain size 64^3) using both single precision floating algorithms. The performance is 2.9572 Million Lattice Updates per Second (MLUPS) on CPU and 391.86 MLUPS on GPU, which shows up to 132-fold speedup of parallel implementation using a single NVIDIA 1050ti GPU compared with the serial implementation.

4.9.2. Validation with commercial package

The LBM-BGK solver built in this study is validated with the reference for Cases 1&2. In STAR-CCM+ we use a steady incompressible laminar flow model after grid independence is tested. The velocity field obtained at non-uniform grids from STAR-CCM+ was interpolated to the uniform lattices of LBM using the nearest interpolation, and compared with the results obtained with LBM for error analysis. The velocity field of STAR-CCM+ for Case 1 is slightly blurred due to the nearest interpolation, as shown in Fig. 4-7a left.

The errors of LBM for Case1&2 after convergence ($\sim 1 \times 10^5$ time-step) compared with the reference are <2%. However it should be noted that STAR-CCM+ uses double precision while the GPU LBM uses single precision for efficiency, and this may cause a difference between the two simulation results. The results show that the LBM-BGK script used in this study is stable with high accuracy. Fig. 4-7a and Fig. 4-8a show the velocity distributions of LBM and the reference for Cases 1&2, and their velocity plots are compared in Fig. 4-7b and Fig. 4-8b. For Case 2 the circular vessel is approximated by staircase and this induces error (mainly near the wall).

STAR-CCM+

LBM-BGK

vel (m/s)

0.0e+00   1.5e-01

(a)

velocity magnitude plot along x-axis

velocity magnitude plot along y-axis
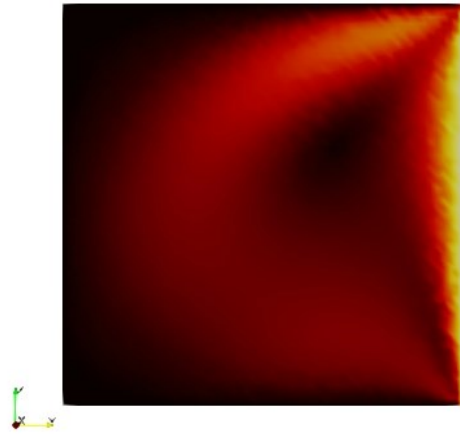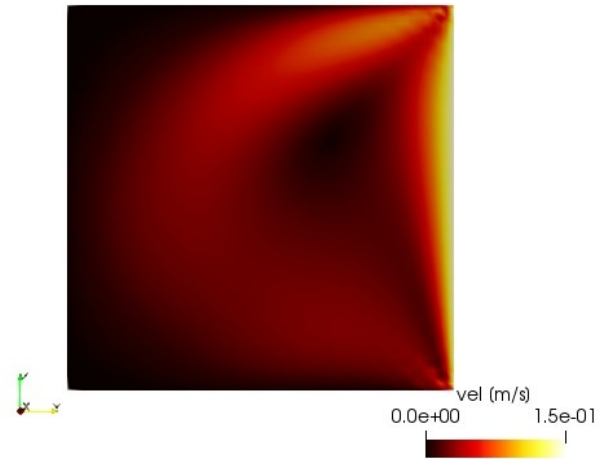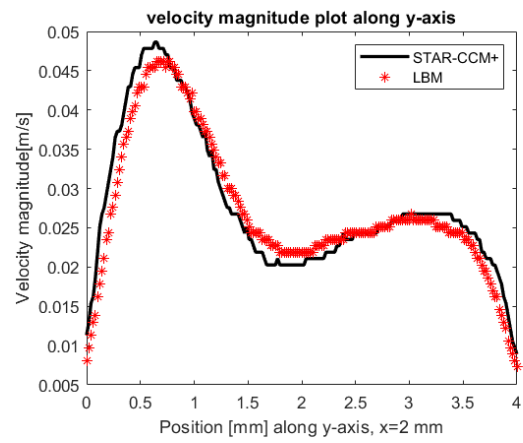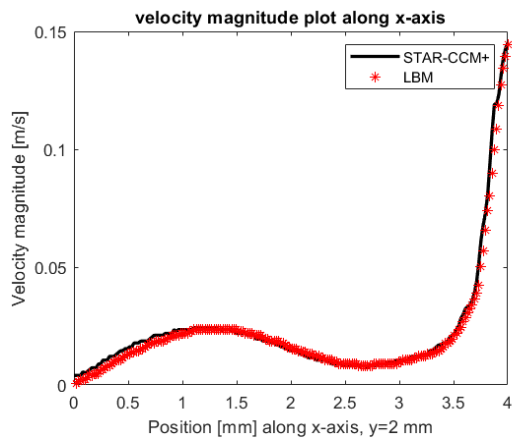
(b)

Figure 4-7 (a) Velocity magnitude at the middle cutting plane for Case 1; (b) velocity plot along x-axis (y=2 mm) and y-axis (x=2 mm)

analytical

LBM-BGK

velo (m/s)
0.0e+00 0.05   0.1   1.5e-01

(a)

velocity magnitude plot along y-axis

(b)
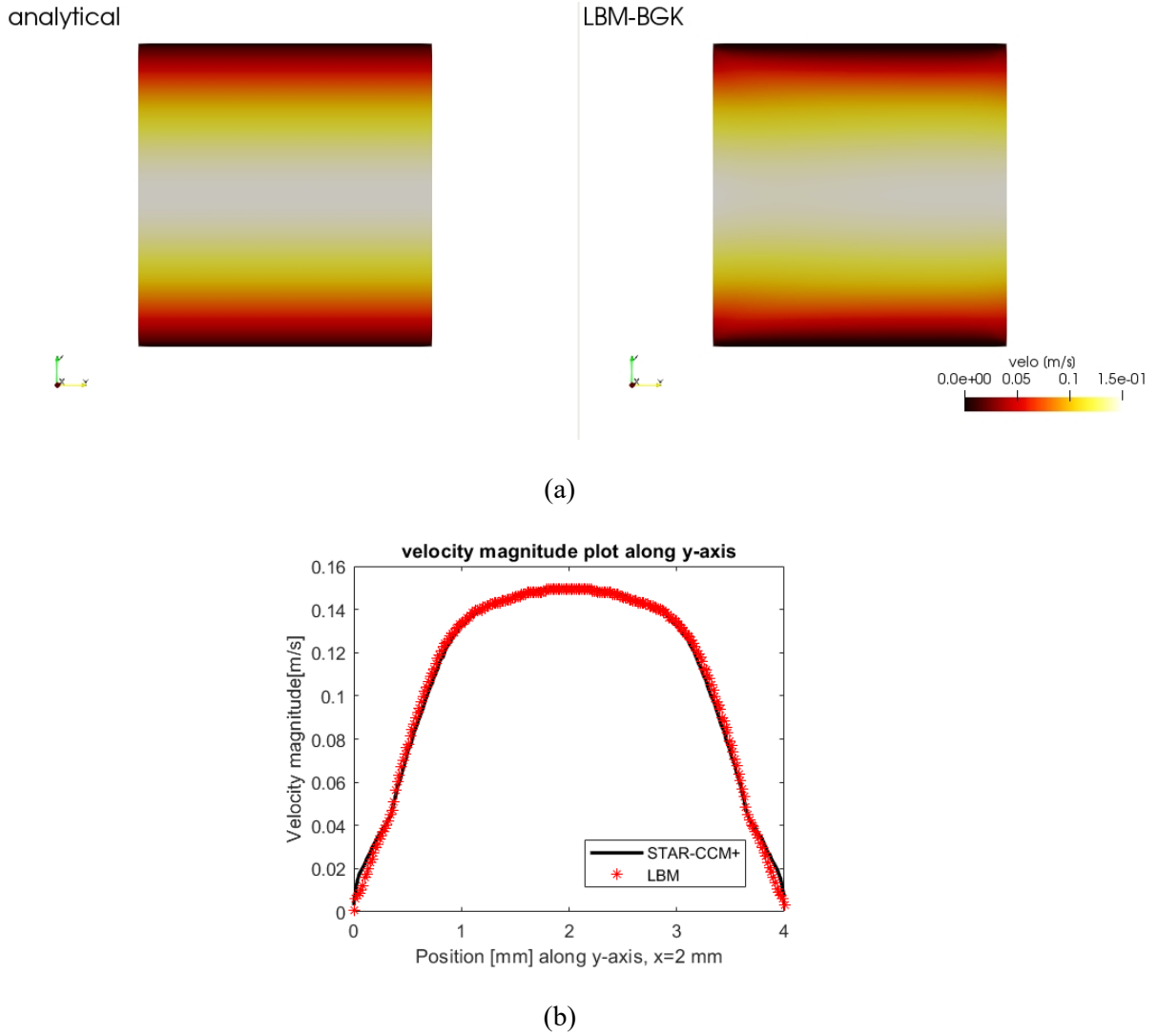
Figure 4-8 (a) Velocity magnitude at the middle cutting plane for Case 2; (b) velocity plot along y-axis (x=2 mm)

4.9.3. Validation of the Selection of α

We validate the selection of α=0 only on the bifurcation flow case (Case 3), which has larger Re number than Case 1&2 and thus a stronger nonlinearity effect. We run LBM with the right BC without data augmentation as the reference, and then intentionally rotate both the BC velocity and augmentation velocity vectors (extracted from the reference) the same angle ß around the z-axis, and evaluate the accuracy of LBM with or without augmentation data using $\alpha = 0$. In this way, the two BCs contain the same registration error.

The rotation angle ß is between -35° and 35° and larger angles are not considered. The results in Fig. 4-9 show that when internal "BC" and normal BC have similar error level, the overall

error will be reduced by augmentation and the selection of $\alpha = 0$ is valid. Fig. 4-9 shows that the overall error of rotating anti-clockwise is larger than rotating clockwise, because more fluid flows into the smaller branch of the bifurcation when rotating the BCs anti-clockwise and causes a larger error in the mass split ratio of the two outlets.
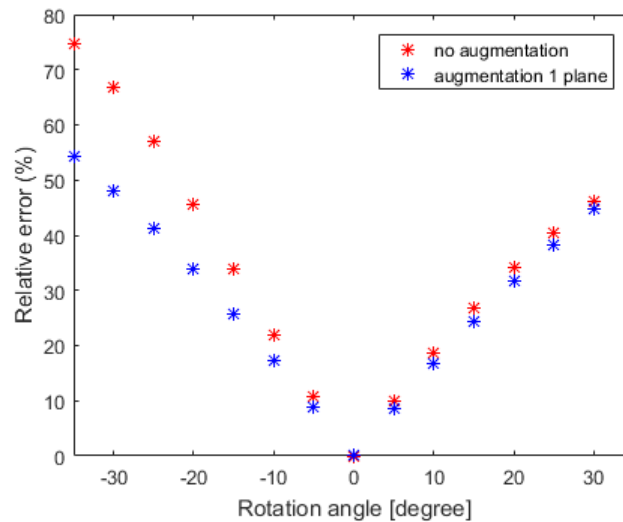
Figure 4-9 Overall CFD Error for bifurcation flow, for which two "BCs" contain same registration error and α=0

4.9.4.　Case 1: steady LDC flow

A) Convergence acceleration with data augmentation

Three simulations are conducted: 1. LBM without data augmentation; 2. LBM with augmentation data on one imaging plane at z=32 (computational domain size 64-64-64); 3. LBM with augmentation data on seven artificial imaging planes at z=8, 16, 24, 32, 40, 48, 56. No artificial error is added to BC. Convergence is achieved when relative error stably decreases to a plateau as shown in Fig. 4-10.

Table 4-1 simulation error after convergence and time

	No augmentation	Augmentation 1 plane	Augmentation 7 planes
Relative Error (%)	4.46×10^{-3}	4.12×10^{-3}	2.98×10^{-3}
Time-Step	~30000	~12000	~2000
Running Time (s)	28.0	11.0	1.7
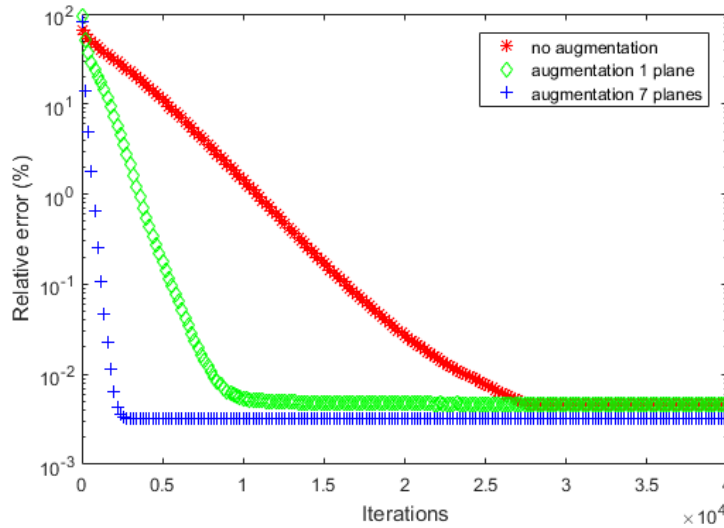
Figure 4-10 convergence acceleration with data augmentation for Case 1

Table 4-1 and Fig. 4-10 show the augmented LBM converges faster to a smaller residual, and increasing number of data fed into the model makes the solver converge faster. With augmentation data from seven planes the LBM converges to a much smaller relative error of $3 \times 10^{-3}\%$ with ~2000 time-steps, more than an order of magnitude faster that without augmentation (a relative error of $4.5 \times 10^{-3}\%$ achieved within $\sim 3 \times 10^4$ time-steps). The velocity magnitude at the middle plane of the three simulations is shown in Fig. 4-11 for sanity check.
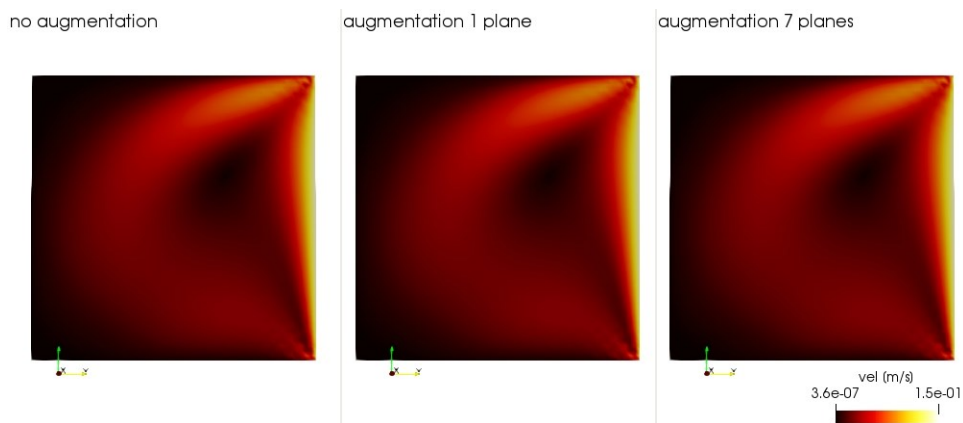
Figure 4-11 velocity magnitude at the middle plane (yz plane) after convergence

B) BC error suppression by data augmentation

99

We added 15% error to the BC, i.e., the sliding wall velocity was 0.1725 m/s. Three simulations are conducted: 1. LBM without data augmentation; 2. LBM with one imaging plane at z=32, and the augmentation data have 10% error, i.e., multiplied by 1.1; 3. LBM with one imaging plane at z=32, and the augmentation data are extracted from reference and error free.

Table 4-2 simulation error after convergence (4×10^4 time-steps)

	wrong BC, no augmentation	wrong BC, 1 plane with 10% error	wrong BC, 1 plane augmentation with no error
Relative Error (%)	15.59	12.67	10.41

Table 4-2 shows the error distribution, which shows that when LBM is compromised with BC error, data with higher fidelity can suppress the local and overall error.
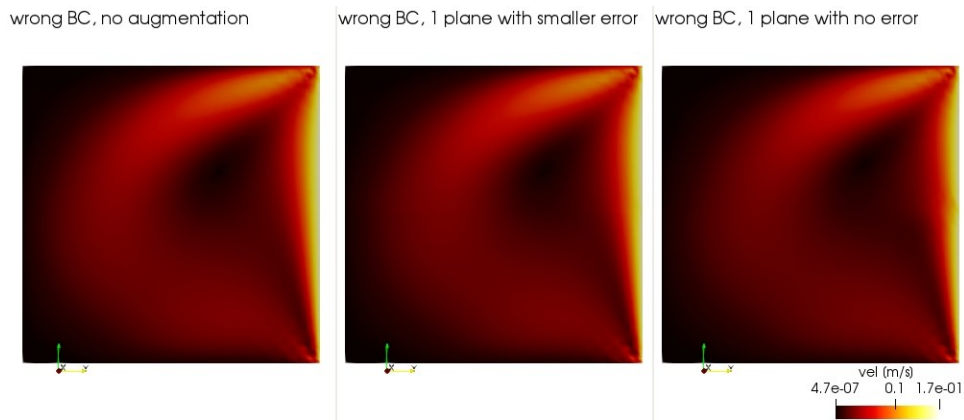
(a) velocity magnitude at the middle plane (yz plane), when artificial error is added to BC
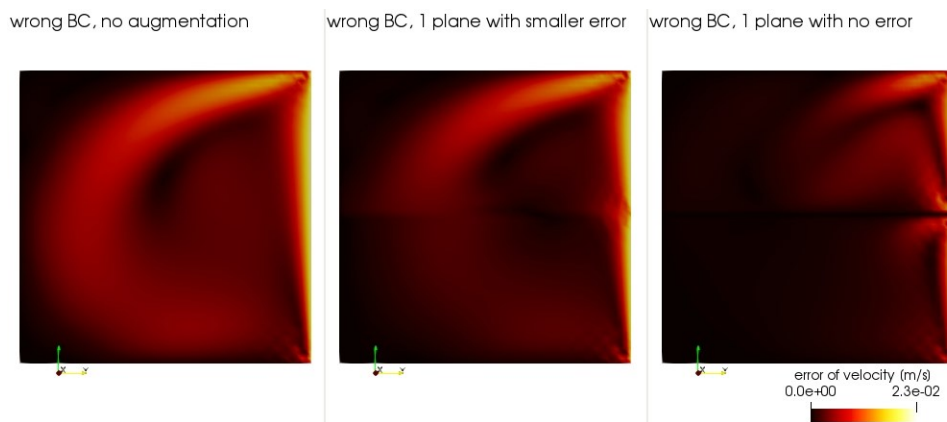
(b) velocity error magnitude distribution at the middle plane (yz plane), when artificial error is added to BC

Figure 4-12 velocity and velocity error distribution after convergence (4×10^4 time-steps)

As introduced in Section 4.3, the internal augmentation data are usually inconsistent with the BCs, but will correct the flow simulation based on the assumption that the augmentation data have higher fidelity than CFD. Fig. 4-12 shows that the local error will be reduced by augmentation data, and so is the overall error (shown in Table 4-2).

4.9.5. Case 2: fully-developed Poiseulle flow

A) Convergence acceleration with data augmentation

Three simulations are conducted: 1. LBM without data augmentation; 2. LBM with data from one imaging plane at z=32 (computational domain size $1 \leq x, y, z \leq 64$); 3. LBM with data from seven artificial imaging planes at z=8, 16, 24, 32, 40, 48, 56. The relative errors of three simulations are shown in Table 4-3, which shows augmented LBM converges faster, and more data make the solver converge faster to a smaller error residual. Fig. 4-13 shows with augmentation data from seven planes the LBM converges to a relative error of $3.00 \times 10^{-3}\%$ with ~1900 time-steps, while without data augmentation the LBM converges to a relative error of $2.3 \times 10^{-3}\%$ with ~6200 time-steps. Convergence is achieved when relative error decrease to a stable plateau as shown in Fig. 4-13. Fig. 4-14 shows the velocity magnitude at the central plane for sanity check.

Table 4-3 simulation error after convergence and time

	No augmentation	Augmentation 1 plane	Augmentation 7 planes
Relative Error (%)	2.26×10^{-3}	2.20×10^{-3}	3.00×10^{-3}
Time-Step	~6200	~4500	~1900
Running Time (s)	8.6	7.1	2.7
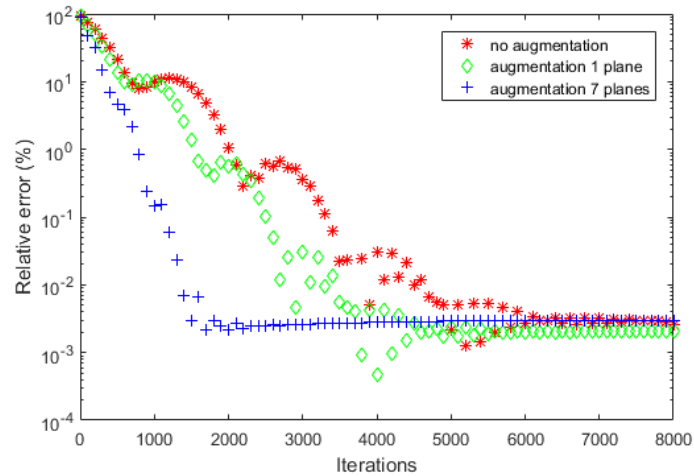
Figure 4-13 Convergence acceleration with data augmentation for Case 2
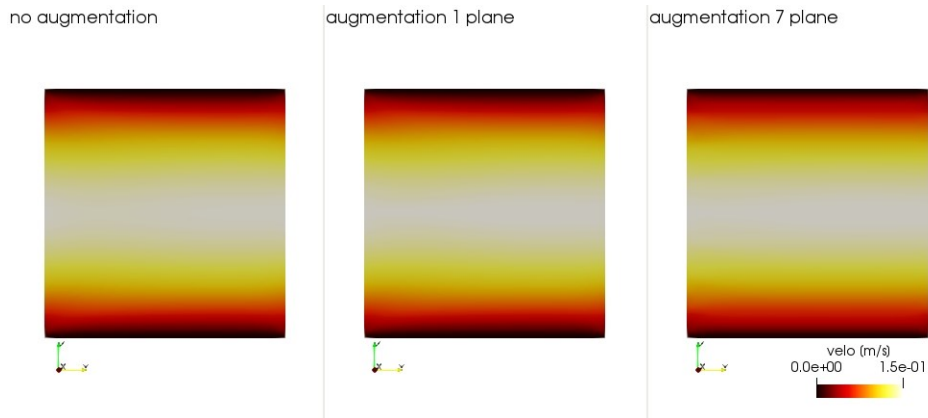
Figure 4-14 velocity magnitude at the middle plane (yz plane) after convergence

B) BC error suppression by data augmentation

We added 15% error to the BC, i.e., the inlet and outlet velocity was multiplied by 1.15. Three simulations are conducted: 1. LBM without data augmentation; 2. LBM with one fictious imaging plane at z=32, and the augmentation data has 10% error, i.e., multiplied by 1.1; 3. LBM with one fictious imaging plane at z=32, and the augmentation data are from reference and thus error free.

Table 4-4 simulation error after convergence (6200 time-steps)

	wrong BC, no augmentation	wrong BC, 1 plane with smaller error	wrong BC, 1 plane augmentation with no error
Relative Error (%)	14.86	9.94	5.57

A similar conclusion to Case 1 can be drawn, i.e., measurements with smaller error than compromised LBM can suppress the overall error. The velocity magnitude and velocity error magnitude at the middle plane of the three simulations are shown in Fig. 4-15, which shows that the local error is suppressed.

(a) velocity magnitude at the middle plane (yz plane), when artificial error is added to BC

(b) velocity error magnitude at the middle plane (yz plane), when artificial error is added to BC
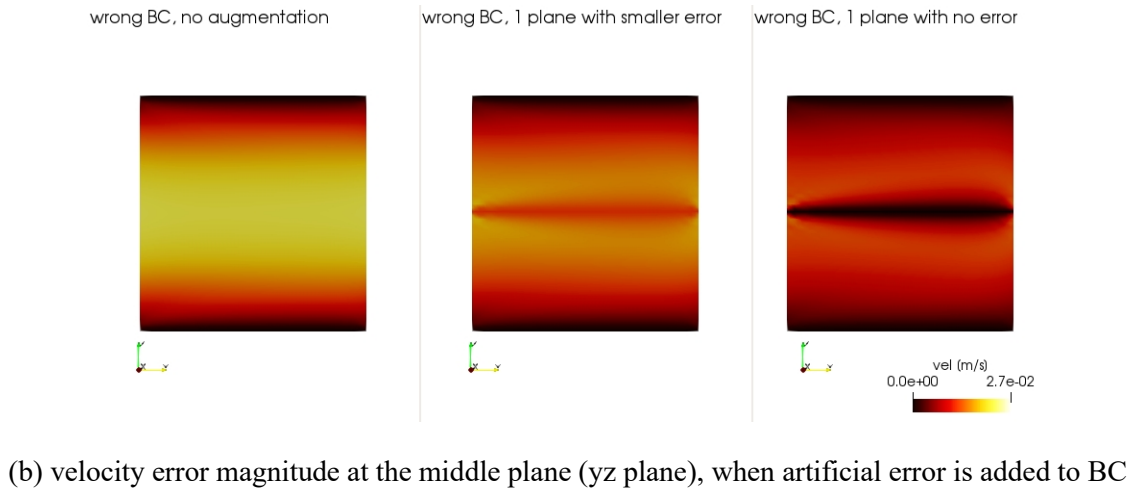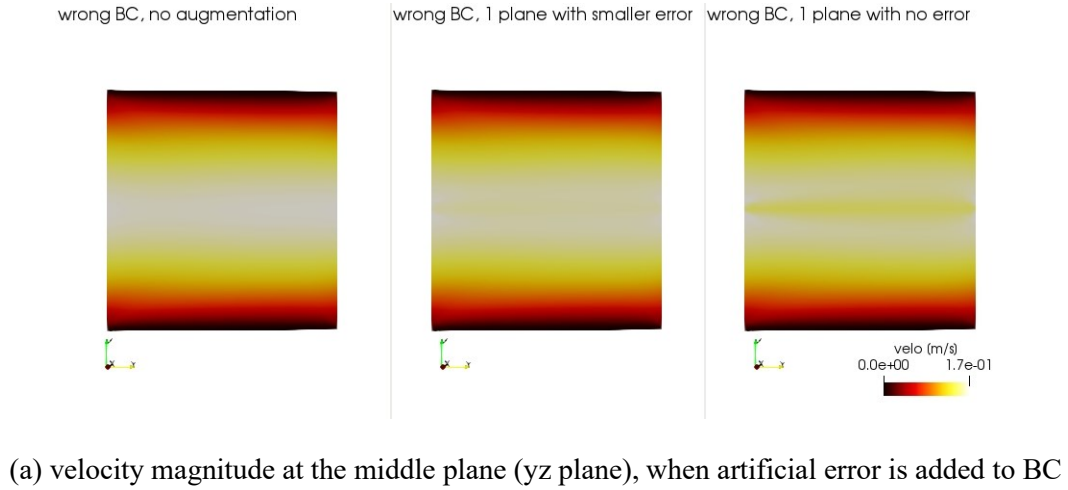
Figure 4-15 velocity and velocity error distribution after 6200 time-steps

4.9.6. Case 3: steady *in vitro* bifurcation phantom flow

The computational domain size is $1 \leq x \leq 64, 1 \leq y \leq 83, 1 \leq z \leq 32$. The 2D velocities from CFD (without augmentation) and UIV (z=14, the augmentation data) are shown in Fig. 4-16, where the relative error of CFD (without augmentation) is 9.19% compared with measurements.
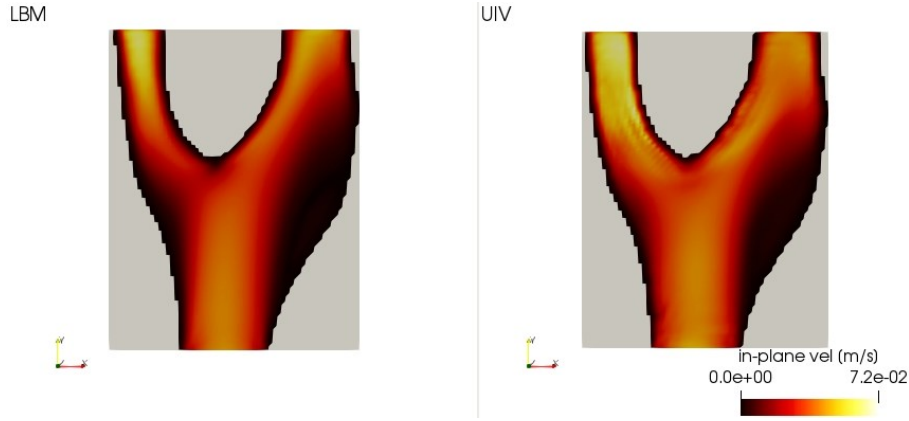
Figure 4-16 In plane velocity magnitude of the reference and UIV

A) Convergence acceleration with data augmentation

Two simulations are conducted: 1. LBM without data augmentation; 2. LBM augmented by 2D velocities from UIV measurements (Fig. 4-6). Convergence is achieved when residual is stable and small. Fig. 4-17 shows with one plane of data augmentation the LBM converges to a residual of 1.5×10^{-6}% with ~3500 time-steps, while without data augmentation the LBM converges to a residual of the same level with ~4800 time-steps (~27% improvement). The velocity magnitude at the middle plane (z=16) of the two simulations is shown in Fig. 4-18 for sanity check. The absolute difference between the augmented LBM and LBM without augmentation is shown in Fig. 4-18, and due to the lack of reference, we only show the *in vitro* feasibility.
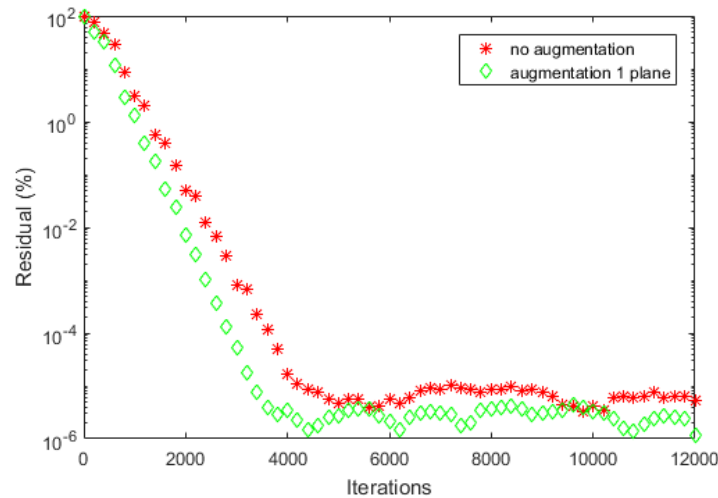
Figure 4-17 Convergence acceleration with data augmentation for Case 3
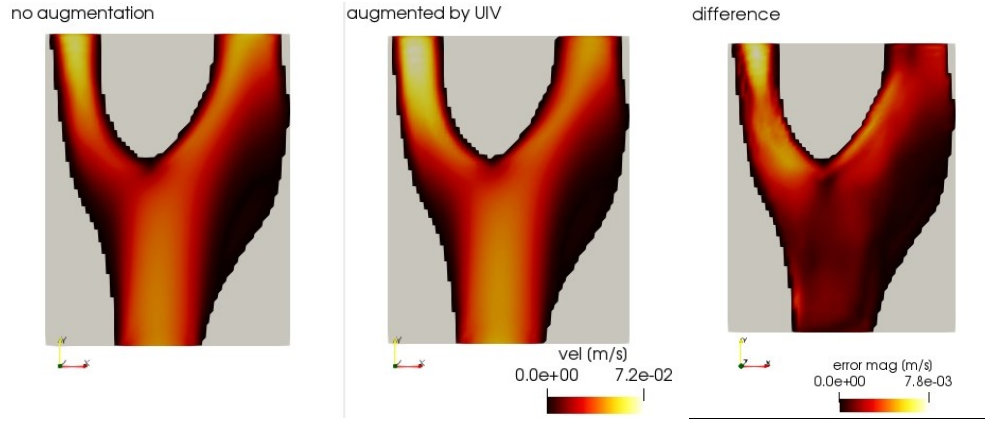
Figure 4-18 velocity magnitude (left and middle) and difference (right) caused by augmentation at the middle plane (yz plane, z=16) after convergence

B) BC error suppression by data augmentation

The outlet mass split ratio of the reference simulation is 37.55%-62.45%. We conducted two sets of LBM simulations with and without UIV augmentation, for each set outlet BC with a certain level of artificial error is applied: 1) LBM with the outlet mass split ratio 27.55%-72.45%; 2) LBM with the outlet mass split ratio 47.55%-52.45%.

Table 4-5 simulation relative error after convergence (5000 time-steps)

	Simulation 1 (27.55%-72.45%)		Simulation 2 (47.55%-52.45%)	
Augmentation	NO	UIV	NO	UIV
Relative Error (%)	15.90	12.17	13.09	11.36

Results in Table 4-5 show that UIV measurements augmented LBM suppress the overall error of the compromised LBM if large perturbation exists in the BCs. However it should be noted that apart from BC uncertainty, the CFD simulation has other uncertainties/errors including geometry, registration error etc, and thus we only use the outlet mass flow rates measured at the two outlets for accuracy evaluation.

4.10. Summary

This chapter first introduces the traditional Lattice Boltzmann Method (LBM) for Newtonian flows modelling. Then an US VFI augmented LBM method is implemented on a General Purpose (GP) GPU, where high-fidelity VFI (e.g., UIV) is used to acquire flow velocities non-invasively. The method uses measured velocities at inlet/outlet as "normal" Boundary

Condition (BC) and those within flow domain as "internal" BC by directly integrating US VFI measurements for equilibrium distribution function calculation into GPU version LBM. As the LBM input (including geometry, inlet/outlet acquired from VFI, viscosity etc) suffers from uncertainties from various factors and nonlinearity, the overall uncertainty of LBM simulation results is larger than input uncertainty (218). We expect such "internal" BCs, which have similar level of uncertainty with inlet/outlet, can suppress local and thus overall LBM uncertainty. Another benefit of the "internal" BCs is to accelerate LBM convergence, as during simulation flow information propagates locally from both the "normal" and "internal" BCs until the full flow domain is filled.

To study convergence acceleration and error suppressing of measurement augmented LBM, we simulate steady flow fields on three cases, where the results show a significant increase in convergence speed and improved accuracy compared with those obtained with conventional LBM. The *in vitro* case shows the experimental feasibility of the algorithm with an ultrafast UIV experimental system, where the results show the LBM augmented by UIV converges faster and the measurements improve the LBM accuracy if large perturbation exists in BCs. Convergence acceleration of Case 3 by 2D velocities is less significant than Case 1&2, because less data (2D velocity components) is integrated. Although the results only show correction of LBM error due to BC error, similarly the method can correct all CFD errors resulting from various uncertainties including fluid boundary segmentation, CFD model, elasticity/pulsatility of the arteries and non-Newtonian rheology of blood etc.

Compared with previous data assimilation CFD, the proposed method is faster as it only requires one CFD simulation. The convergence rate of the measurements augmented LBM depends on the number and position of measurements, the number of computational lattices, the regularization parameter α and the complexity of flow field. LBM converges faster with more augmentation data. Consider the extreme case where 3D velocity and pressure at all

LBM lattices are acquired by measurements LBM is not needed and convergence time is 0. The position of experimental acquisition is vital for convergence acceleration, and depends on the flows. For blood flow in a circular vessel velocity acquired at the centre sagittal plane may provide most information for augmentation but this needs further study.

Limitations of the work is, LBM is memory bound and data intensive while GPU has a rather limited global memory, which restricts the application of the algorithm in complex flow simulation in large geometry. Finite difference LBM (219) or LBM on multiple GPUs (220) are possible methods to overcome this limitation.

The proposed method, by integrating experimental measurements into CFD, is shown to be faster and more accurate than existing physiological CFD approach if highly accurate flow measurements are available. The method has potential clinical application for fast and accurate blood flow quantification, and this will be studied in the future.

Chapter 5   Conclusions

5.1   Summary of the present research

This thesis has proposed two new algorithms for 3D flow reconstruction and simulation, i.e., US augmented DFI reconstruction with 2D VFI velocity measurements, and US VFI augmented LBM simulation. In addition the feasibilities of 3D DFI using 1D US VFI and non-Newtonian LBM with GPU are studied in the Appendix. Knowledge gaps in terms of 3D flow reconstruction with 2D and 1D flow measurements, effects of measurement noise on reconstruction robustness, reconstruction accuracy and efficiency, and LBM error and convergence rate with/without measurement augmentation are investigated.

Chapter 2 proposes a new algorithm, UADFI, to investigate the capabilities of 3D DFI to reconstruct 3D full field flow with 2D US VFI measurements. Chapter 2 first introduces the existing 3D DFI algorithm, where the input is 3D velocity measurements. Then a combined projection matrix is constructed and then multiplied to both sides of the 3D DFI linear system, resulting in an under-determined linear system which accepts 2D US VFI as input. To increase the robustness and inverse stability, the under-determined linear system is solved by TSVD, where a low rank approximate solution is found. In the end Chapter 2 compares three different methods to optimize the regularization parameter, to maximally suppress the influence of measurement noise, and the results for the Poiseulle flow reconstruction show that the regularization parameter found by the L-curve method leads to the highest accuracy.

Chapter 3 optimizes the accuracy and efficiency of UADFI. The shape parameter of DFI RBFs affects both the accuracy and inverse stability of the solution space of DFI and its optimization is challenging and never studied before. The hold-out validation is proposed to find the optimal shape parameter, so that an optimal set of linear basis can be found. To

increase the computational efficiency Chapter 3 then aims to utilize a projection based iterative method, i.e., the GMRES to solve the ill-posed UADFI system. GMRES will breakdown for singular linear systems. To prove GMRES is suitable for the particular UADFI system, the Krylov subspace of the UADFI is analysed and it is found that with certain conditions the GMRES does not breakdown. We further conduct a series of matrix transforms to increase the sparsity of the system and thus computational efficiency. The results show that, with careful consideration, UADFI is capable of reconstructing 3D full field flow in various scenarios with a typical reconstruction time of ~1s and high accuracy (error ~10% for *in vitro* case).

Chapter 4 develops a parallel GPU version LBM solver, where different optimization techniques are used to optimize the use of GPU for computational and memory efficiency. We assume that BC and the internal flow measurements have higher fidelity than compromised CFD. The results show that the parallel version LBM is 132-fold faster than the serial CPU based LBM using the 768-core GPU in this study. Then we integrate the velocity measurement from US VFI as "internal boundary condition" to LBM with perturbed BCs, and the results show that with such augmentation data, LBM converges at least ~30% faster and suppresses error compared with no measurement augmentation. Using the GPU in this thesis, the typical flow simulation time (for steady *in vitro* bifurcation flow) with measurement augmentation is ~1s, without consideration of US data processing, speckle tracking using UIV.

Appendix A1 extends the optimized UADFI to conduct 3D reconstruction from 3D distribution of 1D velocity measurements, and the results show the method has the potential for fast and accurate (~91% accuracy) intraventricular flow reconstruction. Appendix A2 studies the GPU version non-Newtonian LBM using the Carreu model, and the results show high accuracy compared with commercial software STAR-CCM+.

5.2 Main conclusions

The main conclusions of this research are summarised as follows:

- The UADFI algorithm is a novel, fast and accurate method for 3D full field flow reconstruction from 2D US VFI measurements. The accuracy of the reconstructed flow is largely affected by US sampling density especially spacing, measurement error, the selection of regularization parameter, the shape parameter etc. To reconstruct small-scale flow features, dense US sampling is necessary. The reconstruction time is mostly affected by the number of US VFI input velocities.

- After optimization UADFI is solved by GMRES with conditions to prevent GMRES from breakdown, and has a speedup up to ~728-fold compared with TSVD. The speedup is also highly dependent on the relative error of the input and the selection of ζ, which has to be determined empirically. Its accuracy is improved by optimizing the shape parameter using hold-out validation. The early-stopping is used as the regularization method. The optimized UADFI has a high reconstruction accuracy and efficiency.

- UADFI can also be extended for 3D reconstruction with 1D US Doppler input, which makes it potentially useful for intraventricular flow quantification. The reconstruction accuracy is largely affected by the location of the US probe, the measurement accuracy, the sampling density and location, and the number of wall velocities from vessel tracking. *In silico* simulation shows the algorithm has accuracy (error <10%) with a reconstruction time <1s.

- Compared with serial implementation using CPU, the GPU based LBM developed in this study is 132-fold faster and can be even faster with a more powerful GPU or GPU clusters. It has comparable accuracy with commercial software, with a high accuracy (error <2%) on a series of benchmark problems. The solver can also be extended to simulate the non-Newtonian flow by locally adjusting the LBM relaxation rate, using the Carreau constitutive model. In this model the flow viscosity is shear rate dependent and thus spatially non-uniform. This can be potentially applied to clinical situations where non-Newtonian blood rheology cannot be neglected.

- Highly accurate flow measurements can be used to augment LBM for convergence acceleration and error suppression. Based on the assumption that CFD boundary condition and augmentation data are acquired by US UIV with the same level of accuracy, augmentation data is expected to have higher fidelity than CFD which is

compromised by uncertainties/errors from sources other than boundary conditions. In this study no relaxation is used so that the measurements are treated as "internal" boundary conditions. As such, LBM converges faster by >30%. With more US measurements, LBM converges faster. In the extreme scenario where 3D velocities at all CFD grids are acquired experimentally, LBM probability distribution functions relax to its equilibrium value without simulation so that LBM simulation is not required. In addition the accuracy of LBM can also be improved as more measurements reduce the CFD uncertainty. The effects of measurement augmentation depend on the locations and number of US velocity measurements, the accuracy of US VFI, the US resolution, and all factors that influence the LBM accuracy. The findings indicate the potential clinical application of the measurement augmented LBM.

In summary to address the challenges of patient specific blood flow quantification listed in Chapter 1, the two algorithms developed in this thesis can advance the current 3D blood flow reconstruction/simulation techniques in terms of efficiency and accuracy, and can make patient specific blood flow quantification faster, cheaper, and more accurate.

5.3 Future research

Based upon the main conclusions of this research, further research and optimization can advance the understanding in regard to the UADFI and ultrasound augmented LBM, and thus potentially paves the way for their potential clinical translations. Several topics worthy of investigation are recommended under the following categories.

5.3.1 UADFI

In this thesis the UADFI algorithm is only validated *in silico* and *in vitro*, and the merits of the UADFI algorithm for *in vivo* flow reconstruction have to be validated, as *in vivo* US VFI may suffer higher level of measurement error. For 3D reconstruction using 1D US VFI (Appendix A1) the input contains less information and the results may suffer from reduced accuracy, and this method needs both *in vitro* and *in vivo* validation, especially for complex intraventricular flow validation.

This merit of UADFI study for pulsatile flow is only validated with *in silico* case, i.e., Womersley flow, and has to be validated *in vivo* where temporal registration, e.g., using ECG, needs to be performed.

One challenge of UADFI is the characterisation of turbulent flow, which may happen in disease situations or complex geometries, e.g., stented vessel, ventricles or flow after stenosis. Turbulent flow is chaotic and usually exhibits unsteady vortices in different spatial and temporal scales, and reconstructing turbulent flow requires very dense velocity samples both spatially and temporally. This is challenging for US VFI but may be of some research and clinical interest, especially when the interested flow is in a small volume and in a short time period.

To further speed up reconstruction GPU may be used for DFI reconstruction including building the DFI interpolation matrix G and solving the UADFI system using GMRES. This is especially helpful when the number of sampled velocities from US VFI is large, and may potentially further accelerate UADFI towards real-time. In addition UADFI preprocessing, i.e., US B-mode image segmentation, is computationally extensive using traditional method and can be accelerated using deep neural network.

5.3.2 US augmented LBM

For the US augmented LBM we did not regularize/relax measurements in this study for simplicity, as: 1. US VFI is shown to be accurate. Ultrafast UIV has a validated *in vitro* error of <10% (184); 2. BCs usually contain same level of errors with VFI measurements, and given the nonlinearity of CFD errors propagate and amplify during simulation resulting in lower CFD fidelity than VFI measurements. However it should be noted that when CFD and augmentation data has similar level of fidelity, a potentially higher accuracy can be achieved by adjusting the weights α between CFD and experimental measurements. Appropriate regularization/relaxation to reflect the balance between the relative noise level in measurements and the errors in CFD boundary conditions requires further study.

Another challenging task is turbulence modelling at high Reynolds number using RANS, LES and DNS, which may be potentially augmented using the method. However US VFI should have a spatial and temporal resolution higher enough to fully resolve the spatiotemporal scales of the turbulence using different methods. This is imperative for subsequent numerical analysis of blood flow in ventricles, where the flow is usually turbulent.

One more practical issue is the numerical stability of the LBM solver used in this study. This study is based on BGK-BLM model and is not stable at high/middle Re number. Other LBM model with improved stability, e.g., the entropic LBM (221, 222), multiple relaxation time LBM (223, 224), may also be augmented by US measurements. As such, the proposed measurement augmented LBM can be more numerical stable for high/middle Re number flow, and can be a promising tool for applications in intraventricular flow, aeronautics, hydrodynamics or automobile engineering where interested flow has a higher Re number.

To optimize the workflow of the measurement augmented LBM, the preprocessing of LBM, e.g., segmentation and VFI, can be optimized by deep neural network (45) in terms of both accuracy and efficiency.

This research only adopted one GPU with 768 cores, and the simulation for *in vitro* steady bifurcation flow takes ~1s. However with a more powerful GPU or with multiple GPUs, the simulation may be further accelerated, which may be of interest for blood flow study and potential acute CVD detection.

A coupled FSI modelling LBM algorithm augmented by measurements can be developed for simulating the intravascular flows where large vessel deformation exists. This can be realised through directly tracking the motion of vessel walls from consecutive US B-mode images, or coupling with a solid mechanics solver for vessel walls but requires further research.

As an initial study to show the effectiveness of measurement augmentation, we did not show unsteady flow results. However to bring the US augmented LBM to clinical applications, the future research will focus on studying the unsteadiness and applying this method to real pulsatile cardiovascular flow modelling. Given that the experimental setup has an imaging time-step down to 0.0001s, and the time-step of LBM is determined by the Courant–Friedrichs–Lewy (CFL) condition and may be smaller than imaging time-step. For this the

augmentation can be conducted when measurements are available, or augmentation data can be fixed during each imaging interval. In addition, more accurate probe motion stepping system is required to record the location of the US probe. An integrated hardware and software system (the US system, a post-processing workstation for UIV, segmentation, flow reconstruction and simulation, visualization etc) has to be built. Human experiments also have to be conducted in the future.

References

1.   Townsend N, Wilson L, Bhatnagar P, Wickramasinghe K, Rayner M, Nichols M. Cardiovascular disease in Europe: epidemiological update 2016. European heart journal. 2016;37(42):3232-45.

2.   Gaziano TA. Cardiovascular disease in the developing world and its cost-effective management. Circulation. 2005;112(23):3547-53.

3.   Gaziano T, Reddy KS, Paccaud F, Horton S, Chaturvedi V. Cardiovascular disease. Disease Control Priorities in Developing Countries 2nd edition: The International Bank for Reconstruction and Development/The World Bank; 2006.

4.   Feigin VL, Barker-Collo S, Krishnamurthi R, Theadom A, Starkey N. Epidemiology of ischaemic stroke and traumatic brain injury. Best Practice & Research Clinical Anaesthesiology. 2010;24(4):485-94.

5.   Mokdad AH, Ford ES, Bowman BA, Dietz WH, Vinicor F, Bales VS, et al. Prevalence of obesity, diabetes, and obesity-related health risk factors, 2001. Jama. 2003;289(1):76-9.

6.   Vasan RS, Larson MG, Leip EP, Evans JC, O'Donnell CJ, Kannel WB, et al. Impact of high-normal blood pressure on the risk of cardiovascular disease. New England journal of medicine. 2001;345(18):1291-7.

7.   Hertog MG, Kromhout D, Aravanis C, Blackburn H, Buzina R, Fidanza F, et al. Flavonoid intake and long-term risk of coronary heart disease and cancer in the seven countries study. Archives of internal medicine. 1995;155(4):381-6.

8.   Kurian AK, Cardarelli KM. Racial and ethnic differences in cardiovascular disease risk factors: a systematic review. Ethnicity and Disease. 2007;17(1):143.

9.   Kannel WB, D'Agostino RB, Belanger AJ. Fibrinogen, cigarette smoking, and risk of cardiovascular disease: insights from the Framingham Study. American heart journal. 1987;113(4):1006-10.

10.   Cecchi E, Giglioli C, Valente S, Lazzeri C, Gensini GF, Abbate R, et al. Role of hemodynamic shear stress in cardiovascular disease. Atherosclerosis. 2011;214(2):249-56.

11.   Nambi V, Chambless L, Folsom AR, He M, Hu Y, Mosley T, et al. Carotid intima-media thickness and presence or absence of plaque improves prediction of coronary heart disease risk: the ARIC (Atherosclerosis Risk In Communities) study. Journal of the American College of Cardiology. 2010;55(15):1600-7.

12.   Zhao S, Xu X, Hughes A, Thom S, Stanton A, Ariff B, et al. Blood flow and vessel mechanics in a physiologically realistic model of a human carotid arterial bifurcation. Journal of biomechanics. 2000;33(8):975-84.

13.   Texon M. Hemodynamic basis of atherosclerosis with critique of the cholesterol-heart disease hypothesis. Cardiovascular Engineering: An International Journal. 2001;1(1):57-8.

14.   Meng H, Tutino V, Xiang J, Siddiqui A. High WSS or low WSS? Complex interactions of hemodynamics with intracranial aneurysm initiation, growth, and rupture: toward a unifying hypothesis. American Journal of Neuroradiology. 2014;35(7):1254-62.

15.   Nguyen KT, Clark CD, Chancellor TJ, Papavassiliou DV. Carotid geometry effects on blood flow and on risk for vascular disease. Journal of biomechanics. 2008;41(1):11-9.

16.   Levesque M, Nerem R. The elongation and orientation of cultured endothelial cells in response to shear stress. Journal of biomechanical engineering. 1985;107(4):341-7.

17.   Okano M, Yoshida Y. Junction complexes of endothelial cells in atherosclerosis-prone and atherosclerosis-resistant regions on flow dividers of brachiocephalic bifurcations in the rabbit aorta. Biorheology. 1994;31(2):155-61.

18.   Tian F-B, Zhu L, Fok P-W, Lu X-Y. Simulation of a pulsatile non-Newtonian flow past a stenosed 2D artery with atherosclerosis. Computers in biology and medicine. 2013;43(9):1098-113.

19.   Peiffer V, Sherwin SJ, Weinberg PD. Does low and oscillatory wall shear stress correlate spatially with early atherosclerosis? A systematic review. Cardiovascular research. 2013;99(2):242-50.

20.   White CJ. Catheter-based therapy for atherosclerotic renal artery stenosis. Circulation. 2006;113(11):1464-73.

21.   Tang D, Yang C, Mondal S, Liu F, Canton G, Hatsukami TS, et al. A negative correlation between human carotid atherosclerotic plaque progression and plaque wall stress: in vivo MRI-based 2D/3D FSI models. Journal of biomechanics. 2008;41(4):727-36.

22.   Taylor CA, Figueroa C. Patient-specific modeling of cardiovascular mechanics. Annual review of biomedical engineering. 2009;11:109-34.

23.   Srichai MB, Lim RP, Wong S, Lee VS. Cardiovascular applications of phase-contrast MRI. American Journal of Roentgenology. 2009;192(3):662-75.

24.   Stalder AF, Russe M, Frydrychowicz A, Bock J, Hennig J, Markl M. Quantitative 2D and 3D phase contrast MRI: optimized analysis of blood flow and vessel wall parameters. Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine. 2008;60(5):1218-31.

25.   Markl M, Chan FP, Alley MT, Wedding KL, Draney MT, Elkins CJ, et al. Time‐resolved three‐dimensional phase‐contrast MRI. Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine. 2003;17(4):499-506.

26.   Leow CH, Tang M-X. Spatio-temporal flow and wall shear stress mapping based on incoherent ensemble-correlation of ultrafast contrast enhanced ultrasound images. Ultrasound in medicine & biology. 2018;44(1):134-52.

27.   McDicken W, Sutherland G, Moran C, Gordon L. Colour Doppler velocity imaging of the myocardium. Ultrasound in medicine & biology. 1992;18(6-7):651-4.

28.   Loupas T, Powers J, Gill RW. An axial velocity estimator for ultrasound blood flow imaging, based on a full evaluation of the Doppler equation by means of a two-dimensional autocorrelation approach. IEEE transactions on ultrasonics, ferroelectrics, and frequency control. 1995;42(4):672-88.

29.   Pairleitner H, Steiner H, Hasenoehrl G, Staudach A. Three‐dimensional power Doppler sonography: imaging and quantifying blood flow and vascularization. Ultrasound in Obstetrics and Gynecology: The Official Journal of the International Society of Ultrasound in Obstetrics and Gynecology. 1999;14(2):139-43.

30.   Leow CH, Bazigou E, Eckersley RJ, Alfred C, Weinberg PD, Tang M-X. Flow velocity mapping using contrast enhanced high-frame-rate plane wave ultrasound and image tracking: Methods and initial in vitro and in vivo evaluation. Ultrasound in medicine & biology. 2015;41(11):2913-25.

31.   Salles S, Chee AJ, Garcia D, Alfred C, Vray D, Liebgott H. 2-D arterial wall motion imaging using ultrafast ultrasound and transverse oscillations. IEEE transactions on ultrasonics, ferroelectrics, and frequency control. 2015;62(6):1047-58.

32.   Matias A, Huggon I, Areias J, Montenegro N, Nicolaides K. Cardiac defects in chromosomally normal fetuses with abnormal ductus venosus blood flow at 10–14 weeks. Ultrasound in Obstetrics and Gynecology: The Official Journal of the International Society of Ultrasound in Obstetrics and Gynecology. 1999;14(5):307-10.

33.   Yock PG, Popp RL. Noninvasive estimation of right ventricular systolic pressure by Doppler ultrasound in patients with tricuspid regurgitation. Circulation. 1984;70(4):657-62.

34.   Wladimiroff J, Tonge H, Stewart P. Doppler ultrasound assessment of cerebral blood flow in the human fetus. BJOG: An International Journal of Obstetrics & Gynaecology. 1986;93(4):471-5.

35.   Crosby J, Amundsen BH, Hergum T, Remme EW, Langeland S, Torp H. 3-D speckle tracking for assessment of regional left ventricular function. Ultrasound in medicine & biology. 2009;35(3):458-71.

36.   Long Q, Xu XY, Ariff B, Thom SA, Hughes AD, Stanton AV. Reconstruction of blood flow patterns in a human carotid bifurcation: a combined CFD and MRI study. Journal of Magnetic Resonance Imaging: An Official Journal of the International Society for Magnetic Resonance in Medicine. 2000;11(3):299-311.

37.   Jung J, Hassanein A. Three-phase CFD analytical modeling of blood flow. Medical engineering & physics. 2008;30(1):91-103.

38.   Leuprecht A, Kozerke S, Boesiger P, Perktold K. Blood flow in the human ascending aorta: a combined MRI and CFD study. Journal of engineering mathematics. 2003;47(3-4):387-404.

39.   Fadlun E, Verzicco R, Orlandi P, Mohd-Yusof J. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. Journal of computational physics. 2000;161(1):35-60.

40.   Patankar S. Numerical heat transfer and fluid flow: CRC press; 2018.

41.   Huebner KH, Dewhirst DL, Smith DE, Byrom TG. The finite element method for engineers: John Wiley & Sons; 2001.

42.   Karniadakis G, Sherwin S. Spectral/hp element methods for computational fluid dynamics: Oxford University Press; 2013.

43.   Monaghan JJ. Smoothed particle hydrodynamics. Annual review of astronomy and astrophysics. 1992;30(1):543-74.

44.   Yu D, Mei R, Luo L-S, Shyy W. Viscous flow computations with the method of lattice Boltzmann equation. Progress in Aerospace Sciences. 2003;39(5):329-67.

45.   Ronneberger O, Fischer P, Brox T. U-net: Convolutional networks for biomedical image segmentation. International Conference on Medical image computing and computer-assisted intervention; 2015: Springer.

46.   Dong C, Loy CC, He K, Tang X. Learning a deep convolutional network for image super-resolution. European conference on computer vision; 2014: Springer.

47.   Christensen-Jeffries K, Browning RJ, Tang M-X, Dunsby C, Eckersley RJ. In vivo acoustic super-resolution and super-resolved velocity mapping using microbubbles. IEEE transactions on medical imaging. 2014;34(2):433-40.

48.   Huynh H, Wang ZJ, Vincent PE. High-order methods for computational fluid dynamics: a brief review of compact differential formulations on unstructured grids. Computers & Fluids. 2014;98:209-20.

49.   Wei F, Westerdale J, McMahon EM, Belohlavek M, Heys JJ. Weighted least-squares finite element method for cardiac blood flow simulation with echocardiographic data. Computational and mathematical methods in medicine. 2012;2012.

50.   D'Elia M, Veneziani A. Methods for assimilating blood velocity measures in hemodynamics simulations: Preliminary results. Procedia Computer Science. 2010;1(1):1231-9.

51.   Nickolls J, Dally WJ. The GPU computing era. IEEE micro. 2010;30(2):56-69.

52.   Vermeire BC, Witherden FD, Vincent PE. On the utility of GPU accelerated high-order methods for unsteady flow simulations: A comparison with industry-standard tools. Journal of Computational Physics. 2017;334:497-521.

53.   Mittal R, Seo JH, Vedula V, Choi YJ, Liu H, Huang HH, et al. Computational modeling of cardiac hemodynamics: Current status and future outlook. Journal of Computational Physics. 2016;305:1065-82.

54.   Feichtinger C, Habich J, KöStler H, Hager G, RüDe U, Wellein G. A flexible Patch-based lattice Boltzmann parallelization approach for heterogeneous GPU–CPU clusters. Parallel Computing. 2011;37(9):536-49.

55.   Myre J, Walsh SD, Lilja D, Saar MO. Performance analysis of single‐phase, multiphase, and multicomponent lattice‐Boltzmann fluid flow simulations on GPU clusters. Concurrency and Computation: Practice and Experience. 2011;23(4):332-50.

56.   Peskin CS. The immersed boundary method. Acta numerica. 2002;11:479-517.

57.   McQueen DM, Peskin CS. Heart simulation by an immersed boundary method with formal second-order accuracy and reduced numerical viscosity. Mechanics for a New Mellennium: Springer; 2001. p. 429-44.

58.   Adib MAHM, Ii S, Watanabe Y, Wada S. Minimizing the blood velocity differences between phase-contrast magnetic resonance imaging and computational fluid dynamics simulation in cerebral arteries and aneurysms. Medical & biological engineering & computing. 2017;55(9):1605-19.

59.   Cho YI, Kensey KR. Effects of the non-Newtonian viscosity of blood on flows in a diseased arterial vessel. Part 1: Steady flows. Biorheology. 1991;28(3-4):241-62.

60.   Ku D, Liepsch D. The effects of non-Newtonian viscoelasticity and wall elasticity on flow at a 90 bifurcation. Biorheology. 1986;23(4):359-70.

61.   Wang S-H, Lee L, Lee J. A linear relation between the compressibility and density of blood. The Journal of the Acoustical Society of America. 2001;109(1):390-6.

62.   Carson P, Li XLX, Pallister J, Moskalik A, Rubin J, Fowlkes J. Approximate quantification of detected fractional blood volume and perfusion from 3-D color flow and Doppler power signal imaging. 1993 Proceedings IEEE Ultrasonics Symposium; 1993: IEEE.

63.   Chang FM, Hsu KF, Ko HC, Yao BL, Chang CH, Yu CH, et al. Fetal heart volume assessment by three‐dimensional ultrasound. Ultrasound in Obstetrics and Gynecology: The Official Journal of the International Society of Ultrasound in Obstetrics and Gynecology. 1997;9(1):42-8.

64.   Dabrowski W, Dunmore-Buyze J, Cardinal HN, Fenster A. A real vessel phantom for flow imaging: 3-D Doppler ultrasound of steady flow. Ultrasound in medicine & biology. 2001;27(1):135-41.

65.   Fenster A, Downey DB, Cardinal HN. Three-dimensional ultrasound imaging. Physics in medicine & biology. 2001;46(5):R67.

66.   Downey DB, Fenster A. Vascular imaging with a three-dimensional power Doppler system. AJR American journal of roentgenology. 1995;165(3):665-8.

67.   Picot PA, Rickey DW, Mitchell R, Rankin RN, Fenster A. Three-dimensional colour Doppler imaging. Ultrasound in medicine & biology. 1993;19(2):95-104.

68.   Guo Z, Moreau M, Rickey DW, Picot PA, Fenster A. Quantitative investigation of in vitro flow using three-dimensional colour Doppler ultrasound. Ultrasound in medicine & biology. 1995;21(6):807-16.

69.   Sklansky MS, Nelson T, Strachan M, Pretorius D. Real‐time three‐dimensional fetal echocardiography: initial feasibility study. Journal of ultrasound in medicine. 1999;18(11):745-52.

70.   Scharf A, Geka F, Steinborn A, Frey H, Schlemmer A, Sohn C. 3D real-time imaging of the fetal heart. Fetal diagnosis and therapy. 2000;15(5):267-74.

71.   Deng J, Sullivan ID, Yates R, Vogel M, Mcdonald D, Linney AD, et al. Real-time three-dimensional fetal echocardiography-optimal imaging windows. Ultrasound in medicine & biology. 2002;28(9):1099-105.

72.   Maulik D, Nanda NC, Singh V, Dod H, Vengala S, Sinha A, et al. Live three‐dimensional echocardiography of the human fetus. Echocardiography. 2003;20(8):715-21.

73.   Provost J, Papadacci C, Arango JE, Imbault M, Fink M, Gennisson J-L, et al. 3D ultrafast ultrasound imaging in vivo. Physics in Medicine & Biology. 2014;59(19):L1.

74.   Provost J, Papadacci C, Demene C, Gennisson J-L, Tanter M, Pernot M. 3-D ultrafast Doppler imaging applied to the noninvasive mapping of blood vessels in vivo. IEEE transactions on ultrasonics, ferroelectrics, and frequency control. 2015;62(8):1467-72.

75.   Yiu BY, Lai SS, Alfred C. Vector projectile imaging: Time-resolved dynamic visualization of complex flow patterns. Ultrasound in medicine & biology. 2014;40(9):2295-309.

76.   Falahatpisheh A, Kheradvar A. Ultrasound-based volumetric particle tracking method. Google Patents; 2018.

77.   Voorneveld J. High Frame Rate Ultrasound Velocimetry of Fast Blood Flow Dynamics. 2019.

78.   Holbek S, Pihl MJ, Ewertsen C, Nielsen MB, Jensen JA. In vivo 3-D vector velocity estimation with continuous data. 2015 IEEE International Ultrasonics Symposium (IUS); 2015: IEEE.

79.   Poelma C, Mari J, Foin N, Tang M-X, Krams R, Caro C, et al. 3D Flow reconstruction using ultrasound PIV. Experiments in fluids. 2011;50(4):777-85.

80.   Grønli T, Smistad E, Nyrnes SA, Gomez A, Lovstakken L. Reconstruction of in vivo flow velocity fields based on a rapid ultrasound image segmentation and B-spline regularization framework. 2016 IEEE International Ultrasonics Symposium (IUS); 2016: IEEE.

81.   Grønli T, Wigen M, Segers P, Lovstakken L. A Fast 4D B-Spline Framework for Model-Based Reconstruction and Regularization in Vector Flow Imaging. 2018 IEEE International Ultrasonics Symposium (IUS); 2018: IEEE.

82.   Garcia D, del Álamo JC, Tanné D, Yotti R, Cortina C, Bertrand É, et al. Two-dimensional intraventricular flow mapping by digital processing conventional color-Doppler echocardiography images. IEEE transactions on medical imaging. 2010;29(10):1701-13.

83.   Narcowich FJ, Ward JD. Generalized Hermite interpolation via matrix-valued conditionally positive definite functions. Mathematics of Computation. 1994;63(208):661-87.

84.   Lowitzsch S. Approximation and interpolation employing divergence-free radial basis functions with applications: Texas A&M University; 2004.

85.   Lowitzsch S. Error estimates for matrix-valued radial basis function interpolation. Journal of Approximation Theory. 2005;137(2):238-49.

86.   Chan KG, Liebling M. Estimation of divergence-free 3D cardiac blood flow in a zebrafish larva using multi-view microscopy. Biomedical Imaging (ISBI), 2015 IEEE 12th International Symposium on; 2015: IEEE.

87.   Bohs LN, Friemel BH, McDermott BA, Trahey GE. A real time system for quantifying and displaying two-dimensional velocities using ultrasound. Ultrasound in medicine & biology. 1993;19(9):751-61.

88.   Bohs LN, Friemel BH, Trahey GE. Experimental velocity profiles and volumetric flow via two-dimensional speckle tracking. Ultrasound in medicine & biology. 1995;21(7):885-98.

89.   Hein I, O'brien WD. Current time-domain methods for assessing tissue motion by analysis from reflected ultrasound echoes-a review. IEEE transactions on ultrasonics, ferroelectrics, and frequency control. 1993;40(2):84-102.

90.   Mailloux G, Langlois F, Simard PY, Bertrand M. Restoration of the velocity field of the heart from two-dimensional echocardiograms. IEEE Transactions on Medical Imaging. 1989;8(2):143-53.

91.   Friemel BH, Bohs LN, Trahey GE. Relative performance of two-dimensional speckle-tracking techniques: normalized correlation, non-normalized correlation and sum-absolute-difference. 1995 IEEE Ultrasonics Symposium Proceedings An International Symposium; 1995: IEEE.

92.   Yeung F, Levinson SF, Parker KJ. Multilevel and motion model-based ultrasonic speckle tracking algorithms. Ultrasound in medicine & biology. 1998;24(3):427-41.

93.   Céspedes I, Huang Y, Ophir J, Spratt S. Methods for estimation of subsample time delays of digitized echo signals. Ultrasonic imaging. 1995;17(2):142-71.

94.   Pan B, Qian K, Xie H, Asundi A. Two-dimensional digital image correlation for in-plane displacement and strain measurement: a review. Measurement science and technology. 2009;20(6):062001.

95.   Kim H-B, Hertzberg JR, Shandas R. Development and validation of echo PIV. Experiments in fluids. 2004;36(3):455-62.

96.   Raffel M, Willert CE, Scarano F, Kähler CJ, Wereley ST, Kompenhans J. Particle image velocimetry: a practical guide: Springer; 2018.

97.   Garcia D. A fast all-in-one method for automated post-processing of PIV data. Experiments in fluids. 2011;50(5):1247-59.

98.   Vlasenko A, Schnorr C. Physically consistent and efficient variational denoising of image fluid flow estimates. IEEE Transactions on Image Processing. 2009;19(3):586-95.

99.   Falahatpisheh A, Kheradvar A. Volumetric echocardiographic particle image velocimetry (V-Echo-PIV). Circulation. 2014;130(suppl_2):A14952-A.

100.   Falahatpisheh A, Kheradvar A. A framework for synthetic validation of 3D echocardiographic particle image velocimetry. Meccanica. 2017;52(3):555-61.

101. Nam K-H, Yeom E, Ha H, Lee S-J. Velocity field measurements of valvular blood flow in a human superficial vein using high-frequency ultrasound speckle image velocimetry. The international journal of cardiovascular imaging. 2012;28(1):69-77.

102. Qian M, Niu L, Wang Y, Jiang B, Jin Q, Jiang C, et al. Measurement of flow velocity fields in small vessel-mimic phantoms and vessels of small animals using micro ultrasonic particle image velocimetry (micro-EPIV). Physics in Medicine & Biology. 2010;55(20):6069.

103. Engelhard S, Voorneveld J, Vos HJ, Westenberg JJ, Gijsen FJ, Taimr P, et al. High-frame-rate contrast-enhanced US particle image velocimetry in the abdominal aorta: first human results. Radiology. 2018;289(1):119-25.

104. Poelma C, Van der Mijle R, Mari J-M, Tang M-X, Weinberg PD, Westerweel J. Ultrasound imaging velocimetry: Toward reliable wall shear stress measurements. European Journal of Mechanics-B/Fluids. 2012;35:70-5.

105. Zhang F, Lanning C, Mazzaro L, Barker AJ, Gates PE, Strain WD, et al. In vitro and preliminary in vivo validation of echo particle image velocimetry in carotid vascular imaging. Ultrasound in medicine & biology. 2011;37(3):450-64.

106. Hong G-R, Pedrizzetti G, Tonti G, Li P, Wei Z, Kim JK, et al. Characterization and quantification of vortex flow in the human left ventricle by contrast echocardiography using vector particle image velocimetry. JACC: Cardiovascular Imaging. 2008;1(6):705-17.

107. Kheradvar A, Houle H, Pedrizzetti G, Tonti G, Belcik T, Ashraf M, et al. Echocardiographic particle image velocimetry: a novel technique for quantification of left ventricular blood vorticity pattern. Journal of the American Society of Echocardiography. 2010;23(1):86-94.

108. Gao H, Bijnens N, Coisne D, Lugiez M, Rutten M, D'hooge J. 2-D left ventricular flow estimation by combining speckle tracking with Navier–Stokes-based regularization: An in silico, in vitro and in vivo study. Ultrasound in medicine & biology. 2015;41(1):99-113.

109. Sengupta PP, Khandheria BK, Korinek J, Jahangir A, Yoshifuku S, Milosevic I, et al. Left ventricular isovolumic flow sequence during sinus and paced rhythms: new insights from use of high-resolution Doppler and ultrasonic digital particle imaging velocimetry. Journal of the American College of Cardiology. 2007;49(8):899-908.

110. Voorneveld J, Muralidharan A, Hope T, Vos HJ, Kruizinga P, van der Steen AF, et al. High frame rate ultrasound particle image velocimetry for estimating high velocity flow patterns in the left ventricle. IEEE transactions on ultrasonics, ferroelectrics, and frequency control. 2017;65(12):2222-32.

111. Voorneveld J, Keijzer LB, Strachinaru M, Bowen DJ, Goei JS, Ten Cate F, et al. High-Frame-Rate Echo-Particle Image Velocimetry Can Measure the High-Velocity Diastolic Flow Patterns. Circulation: Cardiovascular Imaging. 2019;12(4):e008856.

112. Smith N, de Vecchi A, McCormick M, Nordsletten D, Camara O, Frangi AF, et al. euHeart: personalized and integrated cardiac care using patient-specific cardiovascular modelling. Interface focus. 2011;1(3):349-64.

113. Gasser TC, Nchimi A, Swedenborg J, Roy J, Sakalihasan N, Böckler D, et al. A novel strategy to translate the biomechanical rupture risk of abdominal aortic aneurysms to their equivalent diameter risk: method and retrospective validation. European Journal of Vascular and Endovascular Surgery. 2014;47(3):288-95.

114. Vitti P, Rago T, Mazzeo S, Brogioni S, Lampis M, De Liperi A, et al. Thyroid blood flow evaluation by color-flow Doppler sonography distinguishes Graves' disease from Hashimoto's thyroiditis. Journal of endocrinological investigation. 1995;18(11):857-61.

115. Byun HS, Rhee K. CFD modeling of blood flow following coil embolization of aneurysms. Medical engineering & physics. 2004;26(9):755-61.

116. Deschamps T, Schwartz P, Trebotich D, Colella P, Saloner D, Malladi R. Vessel segmentation and blood flow simulation using level-sets and embedded boundary methods. International Congress Series; 2004: Elsevier.

117.   Oshima M, Torii R, Kobayashi T, Taniguchi N, Takagi K. Finite element simulation of blood flow in the cerebral artery. Computer methods in applied mechanics and engineering. 2001;191(6-7):661-71.

118.   Leuprecht A, Perktold K, Kozerke S, Boesiger P. Combined CFD and MRI study of blood flow in a human ascending aorta model. Biorheology. 2002;39(3, 4):425-9.

119.   Fedosov DA, Noguchi H, Gompper G. Multiscale modeling of blood flow: from single cells to blood rheology. Biomechanics and modeling in mechanobiology. 2014;13(2):239-58.

120.   Nitä C, Itu LM, Suciu C. GPU accelerated blood flow computation using the lattice Boltzmann method. 2013 IEEE High Performance Extreme Computing Conference (HPEC); 2013: IEEE.

121.   Caballero A, Mao W, Liang L, Oshinski J, Primiano C, McKay R, et al. Modeling left ventricular blood flow using smoothed particle hydrodynamics. Cardiovascular engineering and technology. 2017;8(4):465-79.

122.   Amritkar A, Deb S, Tafti D. Efficient parallel CFD-DEM simulations using OpenMP. Journal of Computational Physics. 2014;256:501-19.

123.   Hoeflinger J, Alavilli P, Jackson T, Kuhn B. Producing scalable performance with OpenMP: Experiments with two CFD applications. Parallel Computing. 2001;27(4):391-413.

124.   Resch M, Sander B, Loebich I. A comparison of OpenMP and MPI for the parallel CFD test case. Proc of the First European Workshop on OpenMP; 1999.

125.   Corrigan A, Camelli FF, Löhner R, Wallin J. Running unstructured grid‐based CFD solvers on modern graphics hardware. International Journal for Numerical Methods in Fluids. 2011;66(2):221-9.

126.   Kraus J, Schlottke M, Adinetz A, Pleiter D. Accelerating a C++ CFD code with OpenACC. 2014 First Workshop on Accelerator Programming using Directives; 2014: IEEE.

127.   Jacobsen D, Thibault J, Senocak I. An MPI-CUDA implementation for massively parallel incompressible flow computations on multi-GPU clusters. 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition; 2010.

128.   Duffy AC, Hammond DP, Nielsen EJ. Production level CFD code acceleration for hybrid many-core architectures. 2012.

129.   Birchall D, Zaman A, Hacker J, Davies G, Mendelow D. Analysis of haemodynamic disturbance in the atherosclerotic carotid artery using computational fluid dynamics. European radiology. 2006;16(5):1074-83.

130.   Morris PD, Ryan D, Morton AC, Lycett R, Lawford PV, Hose DR, et al. Virtual fractional flow reserve from coronary angiography: modeling the significance of coronary lesions: results from the VIRTU-1 (VIRTUal Fractional Flow Reserve From Coronary Angiography) study. JACC: Cardiovascular Interventions. 2013;6(2):149-57.

131.   Nørgaard BL, Leipsic J, Gaur S, Seneviratne S, Ko BS, Ito H, et al. Diagnostic performance of noninvasive fractional flow reserve derived from coronary computed tomography angiography in suspected coronary artery disease: the NXT trial (Analysis of Coronary Blood Flow Using CT Angiography: Next Steps). Journal of the American College of Cardiology. 2014;63(12):1145-55.

132.   Tu S, Barbato E, Köszegi Z, Yang J, Sun Z, Holm NR, et al. Fractional flow reserve calculation from 3-dimensional quantitative coronary angiography and TIMI frame count: a fast computer model to quantify the functional significance of moderately obstructed coronary arteries. JACC: Cardiovascular Interventions. 2014;7(7):768-77.

133.   Brown AG, Shi Y, Arndt A, Müller J, Lawford P, Hose DR. Importance of realistic LVAD profiles for assisted aortic simulations: evaluation of optimal outflow anastomosis locations. Computer methods in biomechanics and biomedical engineering. 2012;15(6):669-80.

134.   Farag MB, Karmonik C, Rengier F, Loebe M, Karck M, von Tengg-Kobligk H, et al. Review of recent results using computational fluid dynamics simulations in patients receiving mechanical assist devices for end-stage heart failure. Methodist DeBakey cardiovascular journal. 2014;10(3):185.

135.   Basri AA, Zubair M, Aziz AF, Ali RM, Tamagawa M, Ahmad KA. Computational fluid dynamics study of the aortic valve opening on hemodynamics characteristics. 2014 IEEE Conference on Biomedical Engineering and Sciences (IECBES); 2014: IEEE.

136.   Zakaria M, Ismail F, Wiriadidjaja S, Aziz M, Tamagawa M, Basri A, et al. Numerical simulation of mechanical heart valve with coherent vortex shedding in OpenFOAM. Proceedings of Mechanical Engineering Research Day 2017. 2017;2017:68-9.

137.   Biasetti J, Hussain F, Gasser TC. Blood flow and coherent vortices in the normal and aneurysmatic aortas: a fluid dynamical approach to intra-luminal thrombus formation. Journal of The Royal Society Interface. 2011;8(63):1449-61.

138.   Ab Naim WNW, Ganesan PB, Sun Z, Liew YM, Qian Y, Lee C-J, et al. Prediction of thrombus formation using vortical structures presentation in Stanford type B aortic dissection: a preliminary study using CFD approach. Applied Mathematical Modelling. 2016;40(4):3115-27.

139.   Long Q, Xu X, Ramnarine K, Hoskins P. Numerical investigation of physiologically realistic pulsatile flow through arterial stenosis. Journal of Biomechanics. 2001;34(10):1229-42.

140.   Khader AS, Shenoy SB, Pai RB, Kamath GS, Sharif NM, Rao V. Effect of increased severity in patient specific stenosis of common carotid artery using CFD—a case study. World Journal of Modelling and Simulation. 2011;7(2):113-22.

141.   Sui B, Gao P, Lin Y, Jing L, Sun S, Qin H. Hemodynamic parameters distribution of upstream, stenosis center, and downstream sides of plaques in carotid artery with different stenosis: a MRI and CFD study. Acta Radiologica. 2015;56(3):347-54.

142.   Zhang Y, Furusawa T, Sia SF, Umezu M, Qian Y. Proposition of an outflow boundary approach for carotid artery stenosis CFD simulation. Computer methods in biomechanics and biomedical engineering. 2013;16(5):488-94.

143.   Frauenfelder T, Boutsianis E, Schertler T, Husmann L, Leschka S, Poulikakos D, et al. In-vivo flow simulation in coronary arteries based on computed tomography datasets: feasibility and initial results. European radiology. 2007;17(5):1291-300.

144.   Cebral JR, Castro MA, Burgess JE, Pergolizzi RS, Sheridan MJ, Putman CM. Characterization of cerebral aneurysms for assessing risk of rupture by using patient-specific computational hemodynamics models. American Journal of Neuroradiology. 2005;26(10):2550-9.

145.   Pekkan K, Whited B, Kanter K, Sharma S, De Zelicourt D, Sundareswaran K, et al. Patient-specific surgical planning and hemodynamic computational fluid dynamics optimization through free-form haptic anatomy editing tool (SURGEM). Medical & biological engineering & computing. 2008;46(11):1139-52.

146.   Pelliccioni O, Cerrolaza M, Herrera M. Lattice Boltzmann dynamic simulation of a mechanical heart valve device. Mathematics and Computers in Simulation. 2007;75(1-2):1-14.

147.   Krafczyk M, Cerrolaza M, Schulz M, Rank E. Analysis of 3D transient blood flow passing through an artificial aortic valve by Lattice–Boltzmann methods. Journal of Biomechanics. 1998;31(5):453-62.

148.   Zhang X-j, Li X, He F. Numerical simulation of blood flow in stented aneurysm using lattice Boltzmann method. 7th Asian-Pacific Conference on Medical and Biological Engineering; 2008: Springer.

149.   Ouared R, Chopard B, Stahl B, Rüfenacht DA, Yilmaz H, Courbebaisse G. Thrombosis modeling in intracranial aneurysms: a lattice Boltzmann numerical algorithm. Computer Physics Communications. 2008;179(1-3):128-31.

150.   Hirabayashi M, Ohta M, Rüfenacht DA, Chopard B. A lattice Boltzmann study of blood flow in stented aneurism. Future Generation Computer Systems. 2004;20(6):925-34.

151.   Ouared R, Chopard B. Lattice Boltzmann simulations of blood flow: non-Newtonian rheology and clotting processes. Journal of statistical physics. 2005;121(1-2):209-21.

152.   Fang H, Lin Z, Wang Z. Lattice Boltzmann simulation of viscous fluid systems with elastic boundaries. Physical Review E. 1998;57(1):R25.

153.   Krafczyk M, Tölke J, Rank E, Schulz M. Two-dimensional simulation of fluid–structure interaction using lattice-Boltzmann methods. Computers & Structures. 2001;79(22-25):2031-7.

154.   Lallemand P, Luo L-S. Lattice Boltzmann method for moving boundaries. Journal of Computational Physics. 2003;184(2):406-21.

155.   Li H, Fang H, Lin Z, Xu S, Chen S. Lattice Boltzmann simulation on particle suspensions in a two-dimensional symmetric stenotic artery. Physical review E. 2004;69(3):031919.

156.   Guo Z, Zheng C, Shi B. An extrapolation method for boundary conditions in lattice Boltzmann method. Physics of Fluids. 2002;14(6):2007-10.

157.   Fang H, Wang Z, Lin Z, Liu M. Lattice Boltzmann method for simulating the viscous flow in large distensible blood vessels. Physical Review E. 2002;65(5):051925.

158.   Artoli A, Hoekstra AG, Sloot PM. Accuracy of 2D pulsatile flow in the lattice Boltzmann BGK method. International Conference on Computational Science; 2002: Springer.

159.   Cosgrove J, Buick J, Tonge S, Munro C, Greated C, Campbell D. Application of the lattice Boltzmann method to transition in oscillatory channel flow. Journal of Physics A: Mathematical and General. 2003;36(10):2609.

160.   Artoli AM, Hoekstra AG, Sloot PM. Mesoscopic simulations of systolic flow in the human abdominal aorta. Journal of biomechanics. 2006;39(5):873-84.

161.   Boyd J, Buick J, Green S. A second-order accurate lattice Boltzmann non-Newtonian flow model. Journal of physics A: Mathematical and General. 2006;39(46):14241.

162.   Boyd J, Buick JM, Green S. Analysis of the Casson and Carreau-Yasuda non-Newtonian blood models in steady and oscillatory flows using the lattice Boltzmann method. Physics of Fluids. 2007;19(9):093103.

163.   Hyakutake T, Matsumoto T, Yanase S. Lattice Boltzmann simulation of blood cell behavior at microvascular bifurcations. Mathematics and Computers in Simulation. 2006;72(2-6):134-40.

164.   Le G, Zhang J. Boundary slip from the immersed boundary lattice Boltzmann models. Physical Review E. 2009;79(2):026701.

165.   Qian Y-H, d'Humières D, Lallemand P. Lattice BGK models for Navier-Stokes equation. EPL (Europhysics Letters). 1992;17(6):479.

166.   Jensen JA, Munk P. A new method for estimation of velocity vectors. IEEE transactions on ultrasonics, ferroelectrics, and frequency control. 1998;45(3):837-51.

167.   Aderson M. Multi-dimensional velocity estimation with ultrasound using spatial quadrature. IEEE transactions on ultrasonics, ferroelectrics, and frequency control. 1998;45(3):852-61.

168.   Jensen JA, Brandt AH, Nielsen MB. Convex array vector velocity imaging using transverse oscillation and its optimization. IEEE transactions on ultrasonics, ferroelectrics, and frequency control. 2015;62(12):2043-53.

169.   Jensen JA Improved vector velocity estimation using directional transverse oscillation. Ultrasonics Symposium (IUS), 2015 IEEE International; 2015: IEEE.

170.   Dotti D. Blood flow measurements by ultrasound correlation techniques. Energ Nucl. 1979;23:571-5.

171.   Bassini M, Dotti D, Gatti E, Pizzolati P, Svelto V. An ultrasonic non-invasive blood flowmeter based on cross-correlation techniques. Ultrason Int Proc. 1979:273-8.

172.   Newhouse VL, Varner LW, Bendick PJ. Geometrical spectrum broadening in ultrasonic Doppler systems. IEEE Transactions on Biomedical Engineering. 1977(5):478-80.

173.   Newhouse VL, Ehrenwald A. The effect of Rayleigh scattering and frequency dependent absorption on the output spectrum of Doppler flowmeters: School of Electrical Engineering, Purdue University; 1976.

174.   Ohtsuki S, Tanaka M. The flow velocity distribution from the Doppler information on a plane in three-dimensional flow. Journal of visualization. 2006;9(1):69-82.

175.   Uejima T, Koike A, Sawada H, Aizawa T, Ohtsuki S, Tanaka M, et al. A new echocardiographic method for identifying vortex flow in the left ventricle: numerical validation. Ultrasound in medicine & biology. 2010;36(5):772-88.

176.   Jensen JA. Estimation of blood velocities using ultrasound: a signal processing approach: Cambridge University Press; 1996.

177.   Stewart SF. Effects of transducer, velocity, Doppler angle, and instrument settings on the accuracy of color Doppler ultrasound. Ultrasound in medicine & biology. 2001;27(4):551-64.

178.   Kripfgans OD, Rubin JM, Hall AL, Fowlkes JB. Vector Doppler imaging of a spinning disc ultrasound Doppler phantom. Ultrasound in medicine & biology. 2006;32(7):1037-46.

179.   Hall AL, Bernardi RB. Method for detecting two-dimensional flow for ultrasound color flow imaging. Google Patents; 1995.

180.   Steel R, Fish PJ. Velocity bias and fluctuation in the standard dual beam Doppler reconstruction algorithm. IEEE transactions on ultrasonics, ferroelectrics, and frequency control. 2002;49(10):1375-83.

181.   Dunmire B, Beach K, Labs K, Plett M, Strandness Jr D. Cross-beam vector Doppler ultrasound for angle-independent velocity measurements. Ultrasound in medicine & biology. 2000;26(8):1213-35.

182.   Fox MD. Multiple crossed-beam ultrasound Doppler velocimetry. IEEE Transactions on Sonics and Ultrasonics. 1978;25(5):281-6.

183.   Trahey GE, Allison JW, Von Ramm OT. Angle independent ultrasonic detection of blood flow. IEEE Transactions on Biomedical Engineering. 1987(12):965-7.

184.   Leow CH, Bazigou E, Eckersley RJ, Alfred C, Weinberg PD, Tang M-X. Flow velocity mapping using contrast enhanced high-frame-rate plane wave ultrasound and image tracking: Methods and initial in vitro and in vivo evaluation. Ultrasound in Medicine and Biology. 2015;41(11):2913-25.

185.   Leow CH, Tang M-X. Spatio-Temporal Flow and Wall Shear Stress Mapping Based on Incoherent Ensemble-Correlation of Ultrafast Contrast Enhanced Ultrasound Images. Ultrasound in Medicine and Biology. 2018;44(1):134-52.

186.   O'Donnell M, Skovoroda AR, Shapo BM, Emelianov SY. Internal displacement and strain imaging using ultrasonic speckle tracking. IEEE transactions on ultrasonics, ferroelectrics, and frequency control. 1994;41(3):314-25.

187.   Sheeran PS, Luois S, Dayton PA, Matsunaga TO. Formulation and acoustic studies of a new phase-shift agent for diagnostic and therapeutic ultrasound. Langmuir. 2011;27(17):10412-20.

188.   Lee WF, Papadopoulou V, Corbett RW, Leow CH, Duncan N, Cosgrove D, et al.. Automated segmentation of blood vessel in contrast enhanced plane wave ultrasound images. Ultrasonics Symposium (IUS), 2016 IEEE International; 2016: IEEE.

189.   Chung J, Nagy JG, O'Leary DP. A weighted GCV method for Lanczos hybrid regularization. Electronic Transactions on Numerical Analysis. 2008;28:149-67.

190.   Hansen PC, Kilmer ME, Kjeldsen RH. Exploiting residual information in the parameter choice for discrete ill-posed problems. BIT Numerical Mathematics. 2006;46(1):41-59.

191.   Bassrei A The L-curve Method for the Regularization Parameter Search in Cross-Well Attenuation Geophysical Tomography2011.

192.   Skrinjar O, Bistoquet A, Oshinski J, Sundareswaran K, Frakes D, Yoganathan A. A divergence-free vector field model for imaging applications. Biomedical Imaging: From Nano to Macro, 2009 ISBI'09 IEEE International Symposium on; 2009: IEEE.

193.   Fasshauer GE, Zhang JG. On choosing "optimal" shape parameters for RBF approximation. Numerical Algorithms. 2007;45(1-4):345-68.

194.   Wang J, Liu G. On the optimal shape parameters of radial basis functions used for 2-D meshless methods. Computer methods in applied mechanics and engineering. 2002;191(23-24):2611-30.

195.   Xinhuan Zhou VP, Chee Hau Leow, Peter Vincent, Meng-Xing Tang. 3D Flow Reconstruction Using Divergence Free Interpolation of Multiple 2D Contrast Enhanced Ultrasound Particle Imaging Velocimetry Measurements Ultrasound in Medicine and Biology. 2018;In Press.

196.   Fornberg B, Zuev J. The Runge phenomenon and spatially variable shape parameters in RBF interpolation. Computers & Mathematics with Applications. 2007;54(3):379-98.

197.   Fornberg B, Piret C. A stable algorithm for flat radial basis functions on a sphere. SIAM Journal on Scientific Computing. 2007;30(1):60-80.

198.   Fornberg B, Larsson E, Flyer N. Stable computations with Gaussian radial basis functions. SIAM Journal on Scientific Computing. 2011;33(2):869-92.

199.   Rippa S. An algorithm for selecting a good value for the parameter c in radial basis function interpolation. Advances in Computational Mathematics. 1999;11(2-3):193-210.

200.   Boyd JP. Six strategies for defeating the Runge Phenomenon in Gaussian radial basis functions on a finite interval. Computers & Mathematics with Applications. 2010;60(12):3108-22.

201.   Saad Y, Schultz MH. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. SIAM Journal on scientific and statistical computing. 1986;7(3):856-69.

202.   Arnoldi WE. The principle of minimized iterations in the solution of the matrix eigenvalue problem. Quarterly of applied mathematics. 1951;9(1):17-29.

203.   Walker HF. Implementation of the GMRES method using Householder transformations. SIAM Journal on Scientific and Statistical Computing. 1988;9(1):152-63.

204.   Brown PN, Walker HF. GMRES on (nearly) singular systems. SIAM Journal on Matrix Analysis and Applications. 1997;18(1):37-51.

205.   Womersley JR. Method for the calculation of velocity, rate of flow and viscous drag in arteries when the pressure gradient is known. The Journal of physiology. 1955;127(3):553-63.

206.   Morris PD, Narracott A, von Tengg-Kobligk H, Soto DAS, Hsiao S, Lungu A, et al. Computational fluid dynamics modelling in cardiovascular medicine. Heart. 2016;102(1):18-28.

207.   D'Elia M, Perego M, Veneziani A. A variational data assimilation procedure for the incompressible Navier-Stokes equations in hemodynamics. Journal of Scientific Computing. 2012;52(2):340-59.

208.   Moore AM, Arango HG, Broquet G, Edwards C, Veneziani M, Powell B, et al. The Regional Ocean Modeling System (ROMS) 4-dimensional variational data assimilation systems: part II– performance and application to the California Current System. Progress in Oceanography. 2011;91(1):50-73.

209.   D'Elia M, Mirabella L, Passerini T, Perego M, Piccinelli M, Vergara C, et al. Applications of variational data assimilation in computational hemodynamics. Modeling of Physiological Flows: Springer; 2012. p. 363-94.

210.   Janjić T, McLaughlin D, Cohn SE, Verlaan M. Conservation of mass and preservation of positivity with ensemble-type Kalman filter algorithms. Monthly Weather Review. 2014;142(2):755-73.

211.   Kato H, Yoshizawa A, Ueno G, Obayashi S. A data assimilation methodology for reconstructing turbulent flows around aircraft. Journal of Computational Physics. 2015;283:559-81.

212.   Chen S, Doolen GD. LATTICE BOLTZMANN METHOD FOR FLUID FLOWS.30(1):329-64.

213.   Zhao-Li G, Chu-Guang Z, Bao-Chang S. Non-equilibrium extrapolation method for velocity and pressure boundary conditions in the lattice Boltzmann method. Chinese Physics. 2002;11(4):366.

214.   Tran N-P, Lee M, Choi DH. Memory-efficient parallelization of 3D lattice Boltzmann flow solver on a GPU. 2015 IEEE 22nd International Conference on High Performance Computing (HiPC); 2015: IEEE.

215.   Amenta N, Bern MW, Kamvysselis M. A new voronoi-based surface reconstruction algorithm. Siggraph; 1998.

216.   Desbrun M, Meyer M, Schröder P, Barr AH. Implicit fairing of irregular meshes using diffusion and curvature flow. Proceedings of the 26th annual conference on Computer graphics and interactive techniques; 1999: Citeseer.

217.   Edelsbrunner H, Mücke EP. Three-dimensional alpha shapes. ACM Transactions on Graphics (TOG). 1994;13(1):43-72.

218.   Bozzi S, Morbiducci U, Gallo D, Ponzini R, Rizzo G, Bignardi C, et al. Uncertainty propagation of phase contrast-MRI derived inlet boundary conditions in computational hemodynamics models of thoracic aorta. Computer methods in biomechanics and biomedical engineering. 2017;20(10):1104-12.

219.   Mei R, Shyy W. On the finite difference-based lattice Boltzmann method in curvilinear coordinates. Journal of Computational Physics. 1998;143(2):426-48.

220.   Xian W, Takayuki A. Multi-GPU performance of incompressible flow computation by lattice Boltzmann method on GPU cluster. Parallel Computing. 2011;37(9):521-35.

221.   Boghosian BM, Yepez J, Coveney PV, Wager A. Entropic lattice Boltzmann methods. Proceedings of the Royal Society of London Series A: Mathematical, Physical and Engineering Sciences. 2001;457(2007):717-66.

222.   Chikatamarla S, Ansumali S, Karlin IV. Entropic lattice Boltzmann models for hydrodynamics in three dimensions. Physical review letters. 2006;97(1):010201.

223.   Yu H, Luo L-S, Girimaji SS. LES of turbulent square jet flow using an MRT lattice Boltzmann model. Computers & Fluids. 2006;35(8-9):957-65.

224.   Mezrhab A, Moussaoui MA, Jami M, Naji H, Bouzidi Mh. Double MRT thermal lattice Boltzmann method for simulating convective flows. Physics Letters A. 2010;374(34):3499-507.

225.   Zydney A, Oliver III J, Colton C. A constitutive equation for the viscosity of stored red cell suspensions: Effect of hematocrit, shear rate, and suspending phase. Journal of Rheology. 1991;35(8):1639-80.

226.   Shukla J, Parihar R, Rao B. Effects of stenosis on non-Newtonian flow of the blood in an artery. Bulletin of Mathematical Biology. 1980;42(3):283-94.

227.   Shibeshi SS, Collins WE. The rheology of blood flow in a branched arterial system. Applied rheology (Lappersdorf, Germany: Online). 2005;15(6):398.

228.   Abraham F, Behr M, Heinkenschloss M. Shape optimization in steady blood flow: a numerical study of non-Newtonian effects. Computer methods in biomechanics and biomedical engineering. 2005;8(2):127-37.

229.   Gijsen FJ, van de Vosse FN, Janssen J. The influence of the non-Newtonian properties of blood on the flow in large arteries: steady flow in a carotid bifurcation model. Journal of biomechanics. 1999;32(6):601-8.

230.   Wang D, Bernsdorf J. Lattice Boltzmann simulation of steady non-Newtonian blood flow in a 3D generic stenosis case. Computers & Mathematics with Applications. 2009;58(5):1030-4.

231.   Gabbanelli S, Drazer G, Koplik J. Lattice Boltzmann method for non-Newtonian (power-law) fluids. Physical review E. 2005;72(4):046312.

232.   Wang C-H, Ho J-R. A lattice Boltzmann approach for the non-Newtonian effect in the blood flow. Computers & Mathematics with Applications. 2011;62(1):75-86.

233.   Guo Z, Zheng C, Shi B. Discrete lattice effects on the forcing term in the lattice Boltzmann method. Physical Review E. 2002;65(4):046308.

Appendices

A1. 3D flow reconstruction with 1D US VFI

Chapters 2&3 demonstrate the feasibility and optimization of 3D DFI using 2D in-plane velocity components. However 1D Doppler imaging (C-mode) has been widely used for intraventricular flow quantification, and it is of great clinical interest to demonstrate the feasibility of 3D flow reconstruction using 1D Doppler along-beam velocity input. This may provide the clinicians a potential tool for fast 3D intraventricular flow visualization.

For this purpose first we assume for single beam Doppler using a matrix probe, we acquire the 3D distribution of 1D along-beam velocity component, i.e., velocity component of a 3D velocity $\vec{v}$ along the US beam direction denoted by $\vec{B}$. $\vec{B}$ points from the virtual Doppler focusing point to the flowing scatterer. Two corresponding changes have to be made to the optimized UADFI algorithm (see Chapter 3): 1. the corresponding projection matrix of UADFI has to be constructed using Eq. 17 which describes projection of a 3D velocity vector onto a 3D vector $\vec{B}$; 2. the condition where GMRES does not breakdown.

Then we construct the combined projection matrix (Eq. 2.18) using $P_{\vec{B}}$ for Doppler imaging and multiply the combined projection matrix to both sides of Eq. 2.13. The obtained system is solved by GMRES, and the condition when GMRES does not breakdown is the maximum iteration step of $m_1 + 3m_2$ where m_1 is the number of 1D velocities acquired by US VFI and m_2 is the number of 3D velocities acquired by wall tracking. The feasibility is demonstrated by *in silico* 3D flow reconstruction using 1D along-beam velocities.

The potential application of this algorithm is to quantify ventricular blood flow which features strong vortices. However ventricle imaging using US is challenging because of low resolution (US resolution is depth dependent) and access limitation (US cannot penetrate human ribs). Due to the difficulty in ventricle US imaging, we do not have real US measurements. As a result, we use 3D steady-state Lid Driven Cavity (LDC, 4mm-4mm-

4mm) flow to test the algorithm, where a main vortex forms in the centre of the square cavity and small vortex may form in the corners for reconstruction. The right side of the cube slides constantly at 0.15 m/s, serving as the moving plane. Density and kinematic viscosity of fluid are 1060 kg/m^3 and 3.2547×10^{-6} m^2/s, Re=222.2. We uniformly extract the 3D velocities at the fictitious imaging points from CFD, and then project them to a vector pointing from the virtual Doppler focusing point to the image point. For Doppler imaging using a matrix US probe, 3D distribution of 1D velocity components can be acquired as shown in Fig. A1-1. Fig. A1-1 shows the input, i.e., 1D along-beam velocity components and 3D velocities at the wall, where the US probe is illustrated by the red point. The input is without artificial noise because we aim at studying the influence of other factors on the reconstruction accuracy. we set ζ=20% (due to relatively large input error) and use GMRES to solve the UADFI system.

The reconstruction time with a total number of 2057 input velocity vectors is 0.75 s, while the reconstruction accuracy is ~91% which shows the high accuracy of the method. Although higher accuracy can be obtained with a larger number of input velocity vectors, we only use 2057 velocity vectors for faster reconstruction. Fig. A1-2 compares the reconstructed 3D flow and the CFD reference.
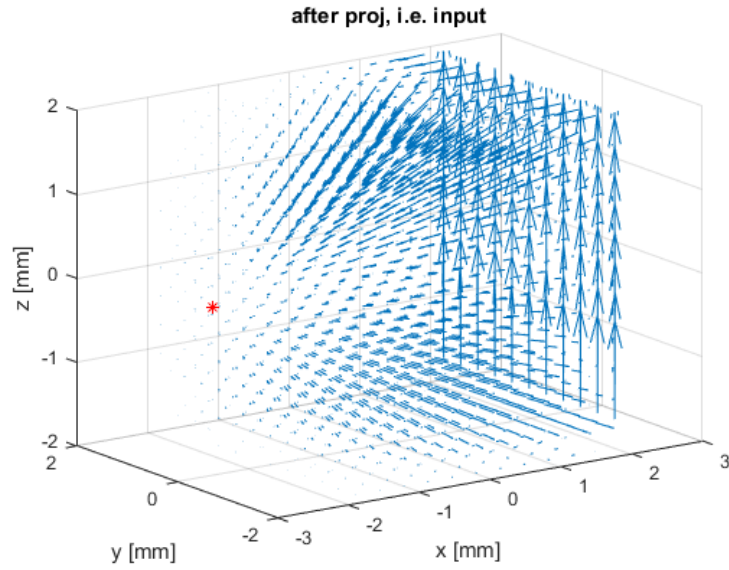
Figure A1- 1 Sparse 1D velocity components from Doppler using a 2D matrix probe, and 3D velocities at the boundary
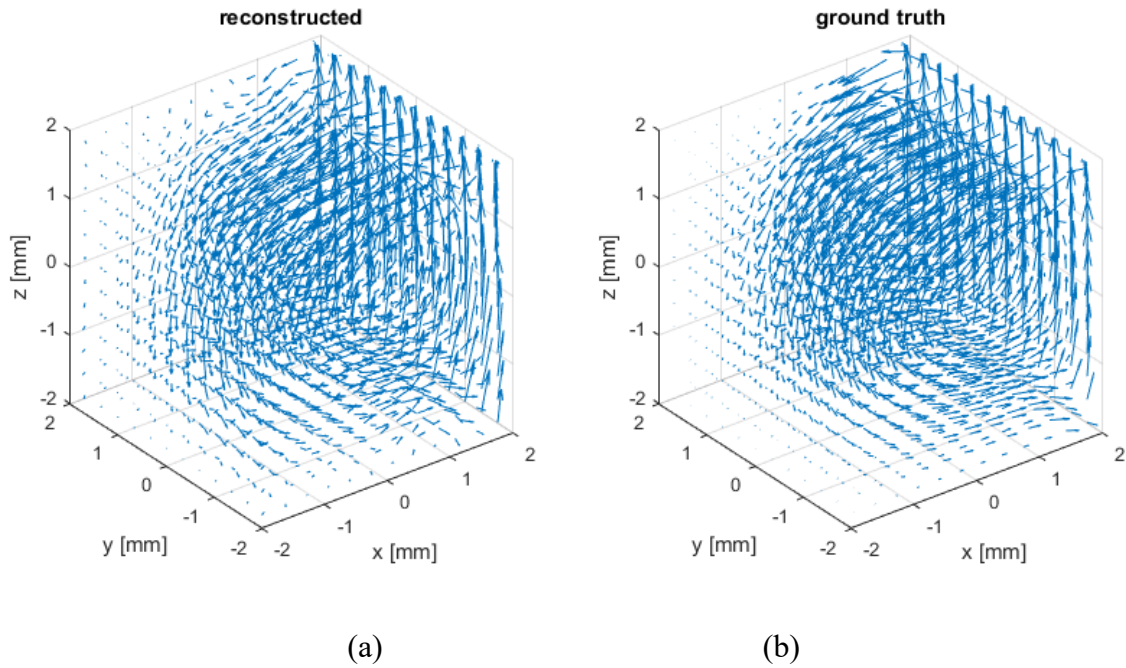

(a)   (b)

Figure A1- 2 (a) reconstructed 3D flow; (b) CFD reference

A2. Non-Newtonian LBM

Chapter 4 only studies Newtonian flow for simplicity, while blood is a shear shinning fluid and exhibits remarkable non-Newtonian behaviours. Blood viscosity is determined by various factors, including the viscosity of plasma, shear rate, mechanical properties, distribution of blood cells especially the Red Blood Cells (RBC), and haematocrit level. Among such factors,

non-Newtonian effects mostly originate from the mechanical properties of the blood cells, as they are elastic and may aggregate forming 3D structures in low shear rate. (225) shows that the size of blood cell aggregation is a function of blood cell concentration and shear rate, and blood cells aggregate to bigger clusters when shear rate tends to 0.

The non-Newtonian effects of blood make the circulation simulation more difficult. Although in most cases blood is assumed Newtonian under normal conditions for simplicity, the non-Newtonian effects must be considered especially in cases of irregular lumen geometries such as stenosed arteries, bifurcation and trifurcation, or in disease conditions. Examples include that non-Newtonian effects result in a higher Wall Shear Stress (WSS) than the Newtonian models (59, 60) in some diseases conditions, and (226) shows that due to the non-Newtonian effects the flow resistance and stress reduce due to the increase of the size of the stenosis.

Appendix A2 introduces the GPU based non-Newtonian LBM. Various constitutive equations have been proposed to describe the non-Newtonian viscosity of blood (59), including the (truncated) Power-Law model, Casson Model, (modified) Powell-Eyring Model, and the Carreau model. In this appendix we use the Carreau model where the dynamic viscosity is

$$\boldsymbol{\mu}(\dot{\mathbf{r}}) = \boldsymbol{\mu}_\infty + (\boldsymbol{\mu_0} - \boldsymbol{\mu}_\infty)[\mathbf{1} + (\boldsymbol{\lambda}\dot{\mathbf{r}})^2]^{(\mathbf{n-1})/2} \qquad\qquad (\text{A}.1)$$

The parameters are $\mu_0 = 0.056\,\text{Pa}\cdot\text{s}$, $\mu_\infty = 0.00345\,\text{Pa}\cdot\text{s}$, $\lambda = 3.313\text{s}$, $n = 0.3568$ for blood (227-229). At low shear rate ($\dot{r} \to 0$) we have $\mu(\dot{r}) \to \mu_0$ and at high shear rate ($\dot{r} \to +\infty$) we have $\mu(\dot{r}) \to \mu_\infty$. $\dot{r} = \sqrt{2D_{\text{II}}}$ is the shear rate where D_{II} is the second invariant of the strain rate tensor defined as

$$D_{II} = \sum_{\alpha,\beta=1}^{3} \sum_{\alpha,\beta=1}^{3} S_{\alpha\beta} S_{\alpha\beta} \qquad (A.2)$$

and $S_{\alpha\beta}$ is the strain rate tensor, a 3-by-3 symmetric matrix for 3D flow. Two methods can be used to calculate $S_{\alpha\beta}$ in LBM, and in the first method $S_{\alpha\beta}$ is defined by a finite difference scheme using its definition

$$S = \begin{bmatrix} \frac{\partial u}{\partial x} & \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) & \frac{1}{2}\left(\frac{\partial u}{\partial z} + \frac{\partial w}{\partial x}\right) \\ \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) & \frac{\partial v}{\partial y} & \frac{1}{2}\left(\frac{\partial v}{\partial z} + \frac{\partial w}{\partial y}\right) \\ \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) & \frac{1}{2}\left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right) & \frac{\partial w}{\partial z} \end{bmatrix} \qquad (A.3)$$

where u, v and w are the three velocities in three directions. However this method is memory expensive as it requires additional device memory to store the velocity gradients after collision in the previous time-step. To avoid this, we use the second method where $S_{\alpha\beta}$ is defined as

$$S_{\alpha\beta} = -\frac{1}{2\tau c_s^2}\left(\sum_i f_i^{(1)} c_{i\alpha} c_{i\beta}\right) \qquad (A.4)$$

where $f_i^{(1)}$ is the nonequilibrium part of the pdf before collision. After obtaining $S_{\alpha\beta}$ the shear dependent viscosity can be calculated as a function of $S_{\alpha\beta}$ by two schemes, as introduced below.

A) Scheme 1: locally adjustment of the relaxation time τ

To include the shear dependent viscosity into LBM model the relaxation parameter τ is locally adjusted using $\tau = 3\nu + 0.5$, where $\nu = \frac{\mu(\dot{r})}{\rho} * \frac{Ct}{CH^2}$ is the nondimensionalized kinematic viscosity (230). $\frac{Ct}{CH^2}$ is the nondimensionalization parameter for kinematic viscosity (not introduced in this thesis). Combining it with Eq. 4.14 we have

$$\tau = \tau_\infty + (\tau_0 - \tau_\infty)[1 + (\lambda\dot{r})^2]^{(n-1)/2} \qquad (A.5)$$

where τ_∞ and τ_0 are the relaxation parameter τ corresponding to 0 and infinite kinematic viscosity. Selection of τ is a trade-off between accuracy and numerical stability, where LBM

suffers from large numerical instability when $\tau \rightarrow 0.5$ and large error when $\tau \rightarrow 1$. (231) showed that setting thresholds on the value of τ can balance the stability and accuracy, but at the risk of unphysical viscosity. The advantage of using the Carreau model is, the shear shinning flows ν is between $\frac{\mu_0}{\rho}$ (when $\dot{r} = 0$) and $\frac{\mu_\infty}{\rho}$ (when $\dot{r} = \infty$) and no unphysical viscosity will be induced. The nondimensionalization parameter for viscosity (i.e., $\frac{Ct}{CH^2}$) has to be carefully chosen so that

$$0.505 \leq \tau_\infty < \tau_0 \leq 0.6 \qquad (A.6)$$

is fulfilled to balance accuracy, stability and computational overheads. Using Scheme 1 for GPU based LBM, two 1D arrays are necessary to store the relaxation rate τ before and after collision respectively and are updated during each time-step.

B) Scheme 2: artificial external force

Apart from locally adjusting the relaxation time, (232) proposed another approach for non-Newtonian LBM simulation by simplifying the non-Newtonian effects as an artificial local shear dependent body force exerting on Newtonian fluids, with a fixed value of τ.

To consider the effects of external force in LBM, first we introduce the LBM equation with external force using the method proposed by (233). The LBM D3Q19 model with external force is given by

$$f_i(\vec{x} + \vec{c}_i dt, t + dt) - f_i(\vec{x}, t) = -\frac{1}{\tau}\left(f_i - f_i^0\right) + \Delta t F_i, i = 0, 1, \dots, 18 \qquad (A.7)$$

With the existence of body force, velocity is updated by

$$\rho\vec{u} = \sum_{i=0}^{18} \vec{c}_i f_i + \frac{1}{2}\vec{F} \qquad (A.8)$$

and F_i is given by (233):

$$F_i = \left(1 - \frac{1}{2\tau}\right) w_i \left[\frac{\vec{c}_i - \vec{u}}{c_s^2} + \frac{\vec{c}_i \cdot \vec{u}}{c_s^4} \vec{c}_i\right] \cdot \vec{F} \qquad (A.9)$$

where $\vec{F}$ is the body force. The β coordinate component of the artificial body force can be written as $\vec{F}_\beta = 2(\mu_0 - \mu_\infty)\{[1 + (\lambda\dot{r})^2]^{(n-1)/2} - 1\}\partial_\alpha S_{\alpha\beta}$, and the dynamic viscosity of the Newtonian fluids is μ_0. Combining Eqs. A.7 and A.9 the discrete form of the artificial non-Newtonian force in LBM language is (232):

$$F_i^{NN} = \left(1 - \frac{\Delta t}{2\tau}\right) w_i \left[\frac{\vec{c}_i - \vec{u}}{c_s^2} + \frac{\vec{c}_i \cdot \vec{u}}{c_s^4} \vec{c}_i\right] \cdot \vec{F} \qquad (A.10)$$

However two problems exist for this scheme: 1. LBM convergences slowly with exerted external body force, and the additional requirement of registers to store F_i^{NN} restricts the number of available threads due to limited GPU shared memory, which jeopardizes the LBM GPU performance. 2. Calculation of $\partial_\alpha S_{\alpha\beta}$ requires two arrays to store the value of the 3-by-3 strain rate tensor $S_{\alpha\beta}$ before and after collision and streaming. Given the symmetry of $S_{\alpha\beta}$, a total number of 6 * NLATTICE single precision floating point variables are needed where NLATTICE is the number of computational grids. This poses a challenge to the GPU version LBM as GPU is usually limited by the global memory capacity.

Considering the above-mentioned reasons, the method to locally adjust τ, as introduced in Scheme 1, is more suitable for CUDA implementation, and it only requires two additional variables to store the shear dependent τ before and after collision. This requires a storage of 2*NLATTICE single precision floating points. Schemes 1&2 have comparable accuracy, however the average computation time of each time-step using Scheme 2 is ~1.6 times of Scheme 1 and is thus not adopted due to the efficiency.

C) non-Newtonian fluid simulation

Three steady Carreau flow simulations in a straight circular vessel with diameter 0.01 m are conducted with different rheological parameters, i.e., different values of the exponential index

n: Simulation 1: n = 0.3568 (the blood parameter used in this study), Simulation 2: n = 0.61, Simulation 3: n = 1 (Newtonian flow). The boundary condition applied is uniform velocity of 0.01 m/s at inlet and atmospheric pressure at the outlet. We simulated the Laminar flow using non-Newtonian Scheme I, and compare the results with reference obtained by STAR-CCM+. For fair comparison, the Carreau model parameters of the STAR-CCM+ simulation are the same with the LBM model used in the study.

The results of LBM are shown below, and the average error of each simulation is ~2% compared with STAR-CCM+, which indicates that the model used in this study has a high accuracy. The error may be caused by: 1. The GPU LBM uses single floating arithmetic for efficiency, while STAR-CCM+ uses double floating arithmetic; 2. The LBM computational grids are uniform, and the geometry of the straight vessel is approximated by staircase; 3. CFD results on a nonuniform grid using STAR-CCM+ have to be interpolated to the LBM uniform grid for accuracy evaluation.
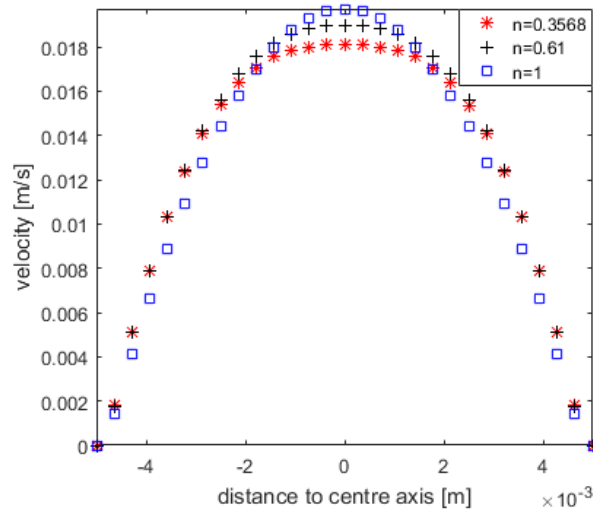

Figure A2- 1 Velocity profile with Carreau model using different values of the exponential index n