Combining existing numerical models with data assimilation using weighted least-squares finite element methods

Authors: Prathish K. Rajaraman, T. A. Manteuffel, M. Belohlavek, and Jeffrey J. Heys

Combining Existing Numerical Models with Data Assimilation using Weighted Least-Squares Finite Element Methods

Prathish K. Rajaraman [a], T.A. Manteuffel [b], M. Belohlavek [c] and Jeffrey J. Heys [a,*]
[a] Chemical and Biological Engineering Department, Montana State University, Bozeman, MT 59717
[b] Department of Applied Mathematics, University of Colorado, Boulder, CO 80309
[c] Mayo Clinic Arizona, Scottsdale, AZ 85259

Summary

A new approach has been developed for combing and enhancing the results from an existing computational fluid dynamics model with experimental data using the weighted least squares finite element method (WLSFEM). Development of the approach was motivated by the existence of both limited experimental blood velocity in the left ventricle (LV) and inexact numerical models of the same flow. Limitations of the experimental data include measurement noise and having data only along a two-dimensional plane. Most numerical modeling approaches do not provide the flexibility to assimilate noisy experimental data. We previously developed an approach that could assimilate experimental data into the process of numerically solving the Navier-Stokes equations, but the approach was limited because it required the use of specific finite element methods for solving all model equations and did not support alternative numerical approximation methods. The new approach presented here allows virtually any numerical method to be used for approximately solving the Navier-Stokes equations, and then the WLSFEM is used to combine the experimental data with the numerical solution of the model equations in a final step. The approach dynamically adjusts the influence of the experimental data on the numerical solution so that more accurate data is more closely matched by the final solution and less accurate data is not closely matched. The new approach is demonstrated on different test problems and provides significantly reduced computational costs compared with many previous methods for data assimilation.

Keywords: Least-Square; Finite elements; Data assimilation; Navier-Stokes, echocardiography;

* Corresponding author: +1(406) 994-7902, jeffrey.heys@coe.montana.edu, Chemical and Biological Engineering Department, Montana State University, Box 173920, Bozeman, MT 59717

1. Introduction

In recent years, numerical models of blood flow in the left ventricle (LV) have been used as a tool by a number of researchers and scientists [1-5]. Echocardiographic particle imaging velocimetry (echo-PIV) methods have been developed in order to better understand the blood flow in the left ventricle with the goal of assessing the overall health of the heart [6-8]. Another example of an imaging method for blood flow in the left ventricle is using magnetic resonances imaging (MRI), but this approach is expensive for obtaining blood flow velocity information within the heart [5, 9, 10]. An alternate approach for imaging blood flow in the heart is using echo-PIV in combination with a microbubble contrast agent added to the blood. This approach is less expensive and less restrictive to patient mobility, but the images have a lower spatial resolution and are currently limited to a single spatial plane (or cross-section) if a high frame rate is desired [9-12].

One advantage of using *in vivo* measurements is to obtain patient specific information regarding blood flow in LV [3]. In contrast, *in silico* methods have the advantage of predicting physiological flow properties that cannot be measured via existing imaging methods and that potentially provide insights into the health of the heart [1, 6, 9]. In the past, researchers have used the data provided by *in vivo* methods as a benchmark and validation tool for *in silico* methods [3]. In order to complement the *in silico* approach, researchers have recently combined both *in vivo* and *in silico* approaches to better understand the physical process and create a patient specific numerical model that can potentially be used for diagnostic and prognostic information [3]. This combination of patient specific information from imaging techniques with a numerical model is one form of data assimilation, and this process requires the combining of noisy experimental data into a numerical model that is based on physical and constitutive laws (e.g., Navier-Stokes).

Data assimilation techniques have the potential to improve the quality of an existing numerical model through the incorporation of actual, physically accurate, patient specific data. For a numerical model of blood flow in the heart, the Navier-Stokes equations are typically the foundation for the numerical model. There are two approaches used extensively to assimilate data into a numerical model: variation methods and stochastic methods. Stochastic methods use Kalman filters that require an ensemble of approximate solutions [3, 9, 12-14]. This is a popular approach in metrology, but it increases computational cost dramatically relative to a single numerical simulation [11, 14]. The variational approach cast the problem as a control theory problem where

the computed numerical solution closely matches the data. Other approaches assimilate experimental data, which often lies at arbitrary spatial locations, by interpolating the data to the computational nodes and then requiring the numerical solution to exactly match the experimental data (i.e., the data is effectively an interior Dirichlet condition). Any error in the data pollutes the numerical solution, and, in some cases, the interpolation process can add even more error to the experimental data [15]. A new approach is presented in this paper that improves upon the previously developed weighted least-squares finite element method (WLSFEM) []. The new approach has several advantages compared to the previous WLSFEM approach, including: a significantly reduction (2-3 times) in computational cost, the ability to continue using existing numerical model algorithms, and potentially improved accuracy, especially with respect to the mass loss that has been previously observed with least-squares finite element methods [9, 12, 14-16].

The new approach is initially tested on Poiseuille flow through a straight cylinder, but the majority of the testing is on the assimilation of experimental data from echocardiographic particle imaging velocimetry (echo-PIV) measurements. The echo-PIV method works by injecting FDA approved microbubbles into the blood stream and using cardiac ultrasonography with a temporal resolution of $150\,\frac{frames}{s}$ (1 cardiac cycle) to determine the location of the bubbles as a function of time [6, 10-12]. Further processing with PIV software on the images from ultrasonography calculates the velocity of the microbubbles based on the displacement and image frame rate information [6, 10-12]. PIVlab, an open source software package, is used here to calculate the 2-dimensional velocity field in the left ventricle from the ultrasound images [17]. The technology of cardiac ultrasonography at sufficiently high frame rates for PIV analysis is limited to 2-dimesional scans. In Figure 1, the resulting 2-dimensional velocity generated by PIVlab is shown. Flow properties, such as viscous energy loss and pressure gradients, cannot be calculated from this limited (2-dimensional) velocity information. In order to determine flow properties of interest, a 3-dimensional velocity field is required, and the approach described here has the ability to assimilate noisy, 2-dimensional echo-PIV data into the numerical solution of a 3-dimensional simulation to produce a 3-dimensional velocity field that is influenced by the experimental data and can be used to compute flow properties that cannot be determined from the experimental data

alone. In this work, the data for assimilation exists on a single plane; however, including additional data on other planes can be accomplished using the framework described here.

[Figure 1 near here]

The new approach also provides flexibility in terms of integrating with almost any existing numerical algorithm for solving the Navier-Stokes equations, and the approach can potentially integrate with a range of numerical models that are based on a partial differential equation (PDE). Most existing numerical methods for the Navier-Stokes equations or other partial differential equation, e.g., the Galerkin finite element, finite volume or finite difference methods, do not allow simple, inexpensive assimilation of experimental data [14]. Here we demonstrate a flexible, new approach that can combine experimental data with an existing numerical solution to a PDE-based model into a final solution using the weighted least-squares finite element framework.

2. Methods

The new method for data assimilation is summarized in Figure 2. The approach is divided into two stages. The first stage begins with a numerical model of the physical problem (the numerical model is based on the Navier-Stokes equations here). The numerical model is solved without any direct assimilation of experimental data, and any numerical approximation method maybe be used to solve the model equations. The second stage is where the data assimilation step is performed, and this step requires the vorticity field, calculated from the numerical solution provided in stage 1, as an input. The WLSFEM approach is used in the second stage to combine experimental data with the numerical solution to obtain a new approximate solution that better reflects the physical problem of interest, provided the experimental data is sufficiently accurate. The individual stages are described below followed by a description of the various test problems employed here.

[Figure 2 near here]

2.1 Navier-Stokes

In order to model blood flow in the left ventricle, blood is assumed to be an incompressible, Newtonian fluid that can be described by the Navier-Stokes equations and can be written in dimensionless form as [18-20]:

$$\left(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \boldsymbol{f}\right) = -\nabla P + \frac{1}{Re}\nabla^2 \boldsymbol{u} \quad in\ \Omega,$$
$$\nabla \cdot \boldsymbol{u} = 0 \quad in\ \Omega, \tag{1}$$

where $\boldsymbol{u}$ is the dimensionless velocity, P is the dimensionless pressure and $\boldsymbol{f}$ is the source terms. Re is the Reynolds number, defined as $Re = \left(\frac{LU\rho}{\mu}\right)$, where L is the characteristic length, U is the characteristic velocity, ρ is the density of the fluid, and μ is the viscosity. Blood flow in the left ventricle is simulated here using $Re = 560$ based on previous measurements of density and viscosity for blood, and using ultrasound images and echo-PIV data to estimate the characteristic length and velocity. The first equation in system (1) is the conservation of momentum for a Newtonian fluid and the second equation describes conservation of mass for an incompressible fluid [18-23]. In order to solve the Navier-Stokes equations (1) for a fluid contained in Ω, the Galerkin finite element method was used to rewrite the partial differential equation into a variational form. The equations in are multiplied by test functions $(\boldsymbol{w}, q)$ and integrated (by parts for second-order terms) over the domain to give:

$$\int_\Omega \boldsymbol{w}\left(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} - \boldsymbol{f}\right) d\Omega + \frac{1}{RE}\int_\Omega \nabla \boldsymbol{w} \cdot \nabla \boldsymbol{u}\ d\Omega - \int_\Omega \nabla \boldsymbol{w} \cdot p\boldsymbol{I}\, d\Omega = 0$$
$$\int_\Omega (\boldsymbol{\nabla} \cdot \boldsymbol{u}) q\, d\Omega = 0\,. \tag{2}$$

Many previous studies have shown that the Galerkin finite element method can suffer from spurious oscillations for convection dominated problems (i.e.,$Re > 100$) unless the mesh is highly refined [19, 20, 23]. A standard approach for avoiding oscillations while retaining a computationally feasible mesh is it to introduce stabilization terms to the Navier-Stokes weak

form. There are many choices for the stabilization method, and the stabilization method used here is the continuous interior penalty finite element method (IPFEM) (for further information the reader can refer to [24, 25]). Based on the unique domain, initial conditions and boundary conditions for each problem, the IPFEM approach is used to obtain a velocity field for every time step. The approximate velocity solution from IPFEM (or any other numerical discretization approach) is then used to compute the vorticity field for a given mesh by taking the curl of the velocity field. This approximate vorticity, which is often discontinuous, is used as an input to the new data assimilation approach that is outlined in the section below.

2.2 Data assimilation

The first stage approximately solved the Navier-Stokes equations and provided the vorticity without any impact from the echo-PIV data. The next step, described in this section, is to assimilate the experimental data and obtain a new numerical solution that is influenced by the patient specific data. The data is assimilated into a new model solution by solving an overdetermined system of first-order differential equations using a weighted least-square finite element method (WLSFEM) approach [9, 12, 15, 16, 26, 27]. This approach was inspired by previous WLSFEM approaches, which have been used successfully for data assimilation [9]. Unlike the previous approaches, the current approach only requires an approximate numerical solution from almost any discretization method for solving the Navier-Stokes equations. Given a numerical approximation for the vorticity, $\boldsymbol{\omega}$, obtained by taking the curl of the velocity solution, a first-order system of equations are solved to combine the experimental data and the numerical solution to determine the new, experimentally influenced velocity field. The WLSFEM equations are:

$$\begin{aligned} \nabla \times \boldsymbol{v} + \boldsymbol{\omega} &= 0 \;\; in\, \Omega, \\ \nabla \cdot \boldsymbol{v} &= 0 \;\; in\, \Omega. \end{aligned} \tag{3}$$

where $\boldsymbol{v}$ is the new, unknown velocity that is influenced by the experimental data. Typically, both the IPFEM step for Navier-Stokes and the WLSFEM assimilation step are solved on the same mesh and use the same boundary conditions for $\boldsymbol{u}$ and $\boldsymbol{v}$. It is also possible to solve Navier-Stokes for the vorticity on a different mesh from the mesh used for the WLSFEM

assimilation step, but having different meshes requires a projection of the vorticity to the WLSFEM mesh. It is also important to note that the WLSFEM does not place any requirements on the finite element space or smoothness for the given vorticity field. Previous studies have shown that when solving the Navier-Stokes equations with many common least-squares finite element methods, the standard LSFEM approach (or even a WLSFEM approach) can lead to an approximate solution with poor mass conservation depending on the method used to rewrite the momentum balance as a system of first-order equations and depending on the finite element approximation space that is used [21, 26, 28, 29]. A number of approaches have been developed to alleviate poor mass conservation, including the use of higher-order polynomial basis functions, but these can lead to higher computational costs for some problems [16, 18, 22, 26]. The approach demonstrated here largely avoids the mass conservation challenge because the Navier-Stokes equations are solved using alternative discretization methods to WLSFEM. The new approach is also computationally cheaper compared to previous WLSFEM approaches used for data assimilation problems [9, 12, 15].

The next step in the WLSFEM framework is to cast system (3) into an unconstrained optimization problem and this is achieved by defining a functional for system (3) by taking the L^2-norm of the equations on the 3D domain (Ω) and summing the equations:

$$F(\boldsymbol{v}; 0) = \|\nabla \times \boldsymbol{v} + \boldsymbol{\omega}\|_{0,\Omega}^2 + \|\nabla \cdot \boldsymbol{v}\|_{0,\Omega}^2 + \frac{1}{h}\|\boldsymbol{v} - \boldsymbol{g}\|_{0,\Gamma}^2 + \frac{w_{PIV}}{h}\|\boldsymbol{v} - v_{PIV}\|_{0,\Gamma_{PIV}}^2. \quad (4)$$

In the WLSFEM framework there are two options when imposing boundary conditions: strong or weak. Imposing weak boundary conditions is done by adding the boundary condition terms into the functional, while strong boundary conditions are enforced by restricting the finite element space [18, 22]. In functional (4), w_{PIV} is the weight used to assimilate the echo-PIV data. The goal is to set the weight for the echo-PIV data so that accurate data is more closely matched by the final numerical solution (i.e., a larger weight), and less accurate data is not matched as closely by the final solution (i.e., a smaller weight). In [9, 15], the reader can find more details on how the weight on the echo-PIV data, w_{PIV}, is optimally calculated, and in [9], a method is described for automatically estimating the weight for a given set of echo-PIV data based on estimates of the temporal accuracy of the data. The approach allows the 2-dimensional echo-PIV velocity, $\boldsymbol{v_{piv}}$, at any spatial location to be weakly matched by the final

solution, $\boldsymbol{v}$ [9, 15]. The functional term for the weak boundary conditions, given by $\boldsymbol{g}$, is scaled by $\frac{1}{h}$, the inverse mesh size, in order to approximate the $H^{\frac{1}{2}}$–norm with a weighted L_2-norm on the boundary. Functional (4) can also be used as a sharp measure of the error in the approximate solution to the first-order system of equations [30]. The WLSFEM data assimilation approach has been used by others [2, 3, 14, 31], and the approach can interpreted as finding a *maximum a posteriori* (MAP) estimator in a Bayesian framework.

2.3. Test Problem I

The new data assimilation approach is first tested on a case where the exact solution, $\boldsymbol{v}$, and the exact vorticity, $\boldsymbol{\omega}$, are known. For the flow in a straight cylinder, i.e., Poiseuille flow, artificial data was generated by adding Gaussian noise to the exact velocity and this artificial data was assimilated to obtain a final, numerical solution. The domain has a dimensionless radius of 0.5 and a dimensionless length of 5.0, and a tetrahedral finite element mesh was generated for the cylindrical geometry. Table 1 lists the boundary conditions used in the simulation and the type of enforcement. The vorticity, $\boldsymbol{\omega}$, input for system (3) was calculated based on the exact velocity solution (i.e., stage 1 in Figure 2 was replaced with the known, exact vorticity solution). In Table 1, the variable $\boldsymbol{f_{data}}$ is the artificially generated 2-dimensional velocity data at arbitrary spatial locations that is assimilated to obtain a final numerical solution. Since $\boldsymbol{\omega}$ is known everywhere in the domain, i.e., for this example the exact vorticity solution is used, the impact of the error in the artificial data on the final velocity, $\boldsymbol{v}$ can be quantified. It should be noted that if, in practice, the numerical model is perfect (exact), one would never assimilate experimental data, but this test problem was developed because it allows for the isolation of error from the experimental data on the final solution.

[Table 1 near here]

The artificial data used for this test case was generated on a plane ($x = 0$), using the exact velocity solution and adding random Gaussian noise. Since the standard deviation of the artificial noise is known for this test problem, the weight on the data assimilation term in the functional (4) was set to $w_{PIV} = 2.0$. The data was assimilated weakly for the two tangential components of the velocity vector along the ($x = 0$) plane.

2.4. Test Problem II

In Test Problem I, the goal was to evaluate the data assimilation scheme when an exact solution from the numerical model is available (i.e., the exact vorticity is known). The goal of the second test problem is to test the new data assimilation scheme when the vorticity field that is obtained from the numerical algorithm used to solve the Navier-Stokes equations also contains error. In this test problem, the same geometry and finite element mesh were used as in the Test Problem I, but in this test, stage 1 used the IPFEM algorithm to approximately solve the Navier-Stokes equations and obtain an approximate vorticity for input into stage 2.

The approximate velocity obtained using IPFEM was still highly accurate for this simple problem relative to the more complex problems that are normally solved using computational fluid dynamics, so quantifiable random noise was added, in some cases, to the approximate velocity solution from the IPFEM method (i.e., stage 1 in Figure 2) before moving to the data assimilation step (stage 2). The amount of noise added into the solution was proportional to τ. As a result, when $\tau = 0.0$ there is no noise added to the numerical solution and for $\tau > 0.0$, varying amounts of error are added to the stage 1 solution to the Navier-Stokes equations.

2.5. Left Ventricle Blood Flow

The new approach was also applied to the left ventricle data assimilation problem, and Table 2 lists the boundary conditions and the type of enforcement (i.e., strong or weak). In the blood flow simulation, the boundary conditions change with time depending on the stage of the cardiac cycle -- filling (late diastole) or ejection (systole) of the blood in the left ventricle. In Table 2, $g_1(t)$ and $\boldsymbol{v_{piv}(t)}$ are the specified velocity of the left ventricle wall and the echo-PIV data on the PIV-plane, and both were obtained from ultrasound measurements. The normal

velocity at the inlet during late diastole and outlet during systole are not specified. The boundary conditions in Table 2 were implemented in order to obtain a numerical solution with maximum influence of the echo-PIV data near the inlet and outlet. At the inlet and outlet the tangential velocity was set to zero and was enforced weakly. Note, the boundary conditions listed (Inlet, Outlet and Wall) in Table 2 were also used for the stage 1 IPFEM simulation.

[Table 2 near here]

The echo-PIV weight, w_{piv}, must be set so that the final solution from stage 2, $\boldsymbol{v}$, is able to match the more accurate velocity data more closely and less accurately velocity data should be matched loosely [9]. When solving for the flow in a cylinder, the standard deviation of the error in the assimilated data (σ) is known, but for the left ventricle case, the error in the assimilated experimental data is not available and the value varies with time and concentration of microbubbles [9, 15]. In [9], an alternative approach for calculating w_{piv} was developed that estimated the accuracy of the experimental data based on the temporal consistency of the data. Large changes to the velocity data over short time intervals result in w_{piv} being set to a smaller value due to the error in the data [9].

2.6. Implementation

The Navier-Stokes equations for stage 1 were solved using the open source package FENICS [32]. A quadratic basis was used for velocity and a linear basis was used for pressure, i.e., P_2P_1 tetrahedral elements [19, 20, 23]. A 2nd-order backward difference formula (BDF-2), which is unconditionally stable, was used for the temporal discretization with a dimensionless time step size of 0.019, the same time step that was available for the ultrasound frame rate that generated the echo-PIV data [19, 20]. In order to capture the blood flow in the left ventricle accurately, the walls of the domain are moved using displacements from the ultrasound images, and the mesh is moved using the FENICS mesh move functionality, which is based on a Poisson solution [32].

ParaFOS is an in-house C code that was developed to solve the second stage using the WLSFEM. The code can import the vorticity $\boldsymbol{\omega}$ from the stage 1 (IPFEM solution), import the echo-PIV data, and minimize the stage 2 WLSFEM functional. Tetrahedral meshes were used in both stage 1 and stage 2 and are generated using Cubit (13.2), developed by Sandia National Laboratory. The mesh quality was determined to be acceptable based on the limited range of element volumes in the final mesh. For example, the coarsest mesh is typically the most difficult to obtain acceptable quality, and the element volumes range from 9×10^{-5} to 7×10^{-4}. An example of mesh generated using Cubit with 52000 tetrahedral shown in Figure 3. Stage 1 was solved using incomplete LU factorization (ILU), while the linear matrix problem in the stage 2 was solved using an algebraic multigrid (AMG) preconditioner for a conjugate gradient (CG) iteration, using two V-cycles for every CG iteration [33, 34]. Simulations for both flow in a cylinder and left ventricle blood flow were performed with meshes varying in total number of elements. The left ventricle blood flow simulation used approximately 52,000 total elements.

[Figure 3 near here]

3. Results and discussion

The main goal of this paper is to examine the performance of the new data assimilation approach. The first test problem uses an exact vorticity field (i.e., an exact CFD model solution) in stage 1 and the data used for assimilation is obtained by adding artificial Gaussian noise to the exact solution. The second test problem uses a CFD algorithm to obtain an approximate vorticity field in stage 1 and then both exact and noisy data are used for assimilation in the second stage. The final test problem uses echo-PIV data obtained from the left ventricle of a pig [35]. Table 3 summarizes the three different test cases in terms of their different inputs.

[Table 3 near here]

3.1. Test Problem I

Figure 4(a), shows the cylinder geometry with the plane where assimilated data is available highlighted in the middle of the cylinder. The exact flow solution for the model problem is

fully developed and laminar, and the exact solution is $\boldsymbol{v} = \left(0,0,-1+(\frac{x^2+y^2}{R^2}\right)$, where $R = 0.5$ the radius of the tube is. Functional (4) can be used as a sharp measure of the error, and larger values for the functional indicate the approximate solution does not satisfy the first-order system of equations (3). Figure 4(b) shows the assimilated data without any noise added, and Figure 4(c) shows one example of the artificial data that was assimilated into the simulation (Test Problem I). The artificial data is generated using the exact solution and adding Gaussian noise with $(\sigma = 0.70711)$ and mean $(\mu = 0.0)$. The data being assimilated into the model is in the $x = 0$ plane and the assimilated data is located only in the mid-region of the plane, away from boundaries, as shown in Figure 4(a).

The results of the cylindrical flow test problem (Test Problem I), with and without data assimilation, are shown in Figure 5 using a mesh with approximately 90,000 tetrahedral elements. Figure 5 shows the simulation without any data assimilated into the numerical model (gray arrows), and the simulation with the data from Figure 4(c) assimilated (black arrows) are overlaid on top at every mesh nodes so that the influence of the assimilated data can be observed. The black arrows in the boxed region are slightly different from the gray arrows, showing that the assimilated data has a small impact on the numerical solution. It would be considered a failure if the data assimilation method showed no difference between the two solutions (with and without data assimilation) and it would be a failure if the data assimilation method showed a large difference between the two solutions because that would indicate that the errors in the assimilated data were being allowed to overly contaminate the final, numerical solution. At the boundaries of the domain, the simulations with and without noisy assimilated data show more consistent results because the assimilated data does not extend all the way to the wall. Global mass conservation can be computed by calculating the $L_2 - norm$ of the continuity equation $\|\nabla \cdot \boldsymbol{v}\|_{0,\Omega}^2$, for the simulations with and without assimilated data. It is interesting to note that improved mass conservation (i.e., less mass loss) is observed when data is assimilated.

[Insert Figure 4 near here]

[Insert Figure 5 near here]

3. 2. Test Problem II

The vorticity that is input into the WLSFEM for the second test case is shown in Figure 6 both without ((a), $\tau = 0.0$) and with ((b), $\tau = 0.025$) artificial noise being added to the numerical solution from the first stage. Recall that the goal of this test problem was to examine the impact of numerical model error from stage 1 on the final result. The initial result was obtained using the IPFEM with $Re = 1.0$ and variable amounts of random noise (i.e., known approximation error) were added to the velocity field. In order to obtain the vorticity field, the negative curl of the velocity is computed. The vorticity is then used as an input to stage 2 to perform the data assimilation step. The $L_2 - norm$ calculated is the difference between the final numerical solution, $\boldsymbol{v}$, and the exact solution for the flow in the cylinder can be calculated after stage 2.

Figure 7 shows the error in the final numerical solution,$\boldsymbol{v}$, after the second stage for different mesh resolutions and varying accuracy for the vorticity field from stage 1. The x-axis reflects the accuracy of the numerical solution from stage one and the y-axis reflects the overall accuracy of the final numerical solution after the second stage. The black points show the solution with the influence of assimilated data (see Figure 4(b)) and the gray points are not influenced by assimilated data. Comparing the gray points to the black points for a given mesh shows that the assimilation of high quality data with proper weighting always improves the accuracy of the final result for this test problem (i.e., the lower L_2-norm of the difference between the approximate and exact solution). This is exactly what one hopes will happen when experimental data (in this case artificial experimental data) is assimilated into an approximate numerical solution – a more accurate approximate solution is obtained. As the mesh is refined (e.g., the 38,000 element mesh (square symbols) and 60,000 (triangles) element mesh) the approximate numerical solution becomes more accurate and there is less benefit associated with the assimilation of the experimental data. Again, this is exactly what one would desire. An exact or nearly exact numerical model does not need the benefits that are associated with assimilating experimental data.

[Insert Figure 6 near here]

In reality, there is a broad accuracy range for different numerical models that could be used for the first stage, and some models will benefit more than others from the assimilation of experimental data. These results also shows the robustness of the new data assimilation approach to compensate for any inaccuracies in the numerical model (i.e., the stage 1 solution) through the addition of data to obtain a final numerical solution (i.e., the stage 2 solution) that better represents the physical system being modeled. It is important to emphasize that for this test problem (Test Problem 2) the artificial error is added to each node in the stage 1 solution so the finer meshes effectively lose accuracy as we move to the right in Figure 6. This was an intentional choice as the overall accuracy of all numerical methods used for stage 1 are well established.

[Insert Figure 7 near here]

3. 3. Results and discussion (Test Problem III)

The new approach was also tested for the blood flow in the left ventricle. The stage 1 numerical model (and many other published models of blood flow in the left ventricle) do not capture absolutely all the physical influences on blood flow in the left ventricle. Many simplifications and assumptions have been made in deriving the mathematical model. For example, the current model does not include the mitral valve. Omitting the mitral valve and other, hopefully minor, physical phenomenon, significantly reduces the computation time and programing complexity. Data assimilation provides a framework to partially recover the lost physical phenomenon due to model simplifications.

In Figure 8, the velocity field on the PIV plane is shown at each mesh node for the numerical solution influenced by the echo-PIV data (black) and without the influence of the data (gray). The impact of the echo-PIV data is highlighted on the PIV plane because this where the impact is greatest. Figure 8(a) highlights a temporal snapshot during the filing phase and Figure 8(b) is a sample time point during the ejection phase. Both plots shown in Figure 8 are a snapshot of a single time step of an entire cardiac cycle simulation, and the assimilated echo-PIV data improves the numerical solution even though the echo-PIV data is noisy (i.e., contains errors). The impact of the assimilated data on the numerical solution is relatively

small, which is anticipated since the numerical model agrees well with the experimental echo-PIV data at the time steps shown in Figure 8.

The boxed areas in Figure 8 were chosen to highlight interesting physics (e.g., vortices) observed during the late diastole (filling) and systole (ejection) phases. In Figure 8(a) the boxed area focused on the recirculation near the outlet and the simulations with and without echo-PIV are consistent, which shows that when the echo-PIV data is consistent with the numerical model, only small change are observed. In Figure 8(b) the boxed region is placed near the outlet during the systole phase, and the gray and black arrows show good agreement near the outlet. This agreement is due to the impact of the boundary conditions at the outlet that dominated the flow field relative the echo-PIV data. This observed solution is reasonable if the model is accurate enough and the data does not contain significant amounts of error.

[Insert Figure 8 near here]

Figure 9 is a snap shot of the numerical solution during the final stages of the late diastole phase where there is minimal flow into the domain. Specifically, the figure shows contours of the magnitude of the dimensionless velocity in the left ventricle. The numerical solution with assimilated echo-PIV data is shown in Figure 9(a), and numerical solution without data can be seen in Figure 9(b). The stage 1 numerical model tends to be less accurate during these lower flow rate time steps due to the underlining assumptions made for the numerical model, which include neglecting the effects of the mitral valve. The contour lines in Figure 9(a) show higher velocities in the lower right region of the left ventricle compared to the contours shown in Figure 9(b). These higher velocities are due to a vortex (i.e., a secondary recirculation of the fluid and higher velocity) generated by the mitral valve. This vortex and locally higher velocity are either not observed or are much weaker when there is no echo-PIV data incorporated into the numerical model because the model does not include the mitral valve. Without the mitral valve, only a very weak vortex is observed in the numerical model at this stage of the cardiac cycle, even with a highly refined mesh. The data assimilation method presented here allows subject-specific echo-PIV data to influence the final solution to the numerical model. (Note: earlier results from Test Problem II showed that when the model is sufficiently accurate, it is not beneficial to assimilate experimental data).

[Insert Figure 9 near here]

In order to quantify the effects of the assimilated echo-PIV data, the value of the functional (4) was calculated at every time step. The functional is a measure of the error in the numerical solution relative to the exact solution to the second stage equations (i.e., system (3)). It is important to note that the functional is not a measure of the error between the true, physical solution and the approximate solution, only a measure of the error for the second stage solution. Hence, a large functional value indicates the incompatibility between the equations defined in system (3) and the echo-PIV data and a smaller value indicated "good" agreement between the model and the data.

In Figure 10, the gray lines are the functional values without data assimilation while the black lines are the functional values with data assimilation calculated at every time step. The WLSFEM approach minimizes the value of the functional over the finite element approximation space, so the functional value must increase when additional data is assimilated and influences the approximate solution. The functional is a measure of the error in the approximate numerical solution to Navier-Stokes equations, and it is important to note that it is not a measure of the error relative to the true solution to the physical problem. Any errors associated with the use of the Navier-Stokes equations and an idealized geometry, for example, are not measured. The functional value increases during the filling phase (late diastole) and decreases during the ejection phase (systole) due to both the change in the mesh size, h, (the expanding domain increases the size of the elements and makes the numerical model less accurate), and the changes in the accuracy of the echo-PIV data (higher velocities can make the tracking of microbubbles more difficult for the PIV software). Comparing both parts of Figure 10, the functional shows a mild decrease when the mesh is refinement from approximately 40000 elements (Figure 10(a)) to 52000 elements (Figure 10(b)), consistent with finite element approximation theory. During the late filling (diastole) phase, the functional increases significantly when the echo-PIV data is assimilated due to the noisy data generated by tracking all the microbubbles.

[Insert Figure 10 near here]

4. Conclusion

We have presented a new numerical approach for assimilating experimental data into an existing numerical model solution through a two-stage process: (1) solving the original numerical model, and (2) assimilating experimental data using the WLSFEM approach. The new approach was used to assess the impact of error in the assimilated experimental data (Test Problem 1) and to assess the impact of error in the stage 1 numerical model (Test Problem 2). Finally, the new approach for data assimilation was demonstrated through the assimilation of echo-PIV data into a numerical model of blood flow in the left ventricle (Test Problem 3). The advantages of the new approach include: the ability to integrate with existing CFD codes (or many other numerical modeling codes based on PDE models), and the computational cost can be significantly lower compared to using least-squares finite element methods for all aspects of the numerical model.

5. Acknowledgment

This work was supported by NSF grants CBET 1249950 and CBET 1403053.

6. References

1. Belohlavek, M., *Vortex formation time: an emerging echocardiographic index of left ventricular filling efficiency?* European Heart Journal-Cardiovascular Imaging, 2012. **13**(5): p. 367-369.
2. D'Elia, M., *Assimilation of velocity data into fluid dynamic simulations, an application to computational hemodynamics*. 2011, Emory University.
3. D'Elia, M., L. Mirabella, T. Passerini, M. Perego, M. Piccinelli, C. Vergara, and A. Veneziani, *Applications of variational data assimilation in computational hemodynamics*, in *Modeling of Physiological Flows*. 2012, Springer. p. 363-394.
4. Sengupta, P.P., V.K. Krishnamoorthy, J. Korinek, J. Narula, M.A. Vannan, S.J. Lester, J.A. Tajik, J.B. Seward, B.K. Khandheria, and M. Belohlavek, *Left ventricular form and function revisited: applied translational science to cardiovascular ultrasound imaging.* Journal of the American Society of Echocardiography, 2007. **20**(5): p. 539-551.
5. Westerdale, J., M. Belohlavek, E.M. McMahon, P. Jiamsripong, J.J. Heys, and M. Milano, *Flow velocity vector fields by ultrasound particle imaging velocimetry in vitro comparison with optical flow velocimetry.* Journal of Ultrasound in Medicine, 2011. **30**(2): p. 187-195.
6. Agati, L., S. Cimino, G. Tonti, F. Cicogna, V. Petronilli, L. De Luca, C. Iacoboni, and G. Pedrizzetti, *Quantitative analysis of intraventricular blood flow dynamics by echocardiographic particle image velocimetry in patients with acute myocardial infarction at different stages of left ventricular dysfunction.* European Heart Journal-Cardiovascular Imaging, 2014: p. jeu106.

7. Bertagna, L., M. D'Elia, M. Perego, and A. Veneziani, *Data Assimilation in Cardiovascular Fluid–Structure Interaction Problems: An Introduction*, in *Fluid-Structure Interaction and Biomedical Applications*. 2014, Springer. p. 395-481.
8. Gao, H., P. Claus, M.-S. Amzulescu, I. Stankovic, J. D'hooge, and J.-U. Voigt, *How to optimize intracardiac blood flow tracking by echocardiographic particle image velocimetry? Exploring the influence of data acquisition using computer-generated data sets.* European Heart Journal-Cardiovascular Imaging, 2011: p. jer285.
9. Rajaraman, P.K., T. Manteuffel, M. Belohlavek, E. McMahon, and J.J. Heys, *Echocardiographic particle imaging velocimetry data assimilation with least square finite element methods.* Computers & Mathematics with Applications, 2014. **68**(11): p. 1569-1580.
10. Sengupta, P.P., G. Pedrizzetti, P.J. Kilner, A. Kheradvar, T. Ebbers, G. Tonti, A.G. Fraser, and J. Narula, *Emerging trends in CV flow visualization.* JACC: Cardiovascular Imaging, 2012. **5**(3): p. 305-316.
11. Borazjani, I., J. Westerdale, E.M. McMahon, P.K. Rajaraman, J.J. Heys, and M. Belohlavek, *Left ventricular flow analysis: recent advances in numerical methods and applications in cardiac ultrasound.* Computational and mathematical methods in medicine, 2013. **2013**.
12. Wei, F., J. Westerdale, E.M. McMahon, M. Belohlavek, and J.J. Heys, *Weighted least-squares finite element method for cardiac blood flow simulation with echocardiographic data.* Computational and mathematical methods in medicine, 2012. **2012**.
13. D'Elia, M., M. Perego, and A. Veneziani, *A variational data assimilation procedure for the incompressible Navier-Stokes equations in hemodynamics.* Journal of Scientific Computing, 2012. **52**(2): p. 340-359.
14. Dwight, R.P. *Bayesian inference for data assimilation using least-squares finite element methods*. in *IOP Conference Series: Materials Science and Engineering*. 2010. IOP Publishing.
15. Heys, J.J., T.A. Manteuffel, S.F. McCormick, M. Milano, J. Westerdale, and M. Belohlavek, *Weighted least-squares finite elements based on particle imaging velocimetry data.* Journal of Computational Physics, 2010. **229**(1): p. 107-118.
16. Heys, J.J., E. Lee, T.A. Manteuffel, S.F. McCormick, and J. Ruge, *Enhanced mass conservation in least-squares methods for Navier-Stokes equations.* SIAM Journal on Scientific Computing, 2009. **31**(3): p. 2303-2321.
17. Thielicke, W. and E.J. Stamhuis, *PIVlab–Towards user-friendly, affordable and accurate digital particle image velocimetry in MATLAB.* Journal of Open Research Software, 2014. **2**(1): p. e30.
18. Bochev, P.B. and M.D. Gunzburger, *Least-squares finite element methods*. Vol. 166. 2009: Springer Science & Business Media.
19. Donea, J. and A. Huerta, *Finite element methods for flow problems*. 2003: John Wiley & Sons.
20. Gresho, P., R. Sani, and M. Engelman, *Incompressible flow and the finite element method: advection-diffusion and isothermal laminar flow. 1998.* John Wiley& Sons.
21. Bochev, P.B., *Analysis of Least-Squares Finite Element Methods for the Navier--Stokes Equations.* SIAM Journal on Numerical Analysis, 1997. **34**(5): p. 1817-1844.
22. Jiang, B.N., *A least-squares finite element method for incompressible Navier-Stokes problems.* International Journal for Numerical Methods in Fluids, 1992. **14**(7): p. 843-859.
23. Rajaraman, P. and J.J. Heys, *Simulation of nanoparticle transport in airways using Petrov–Galerkin finite element methods.* International journal for numerical methods in biomedical engineering, 2014. **30**(1): p. 103-116.
24. Burman, E. and M.A. Fernández, *Continuous interior penalty finite element method for the time-dependent Navier–Stokes equations: space discretization and convergence.* Numerische Mathematik, 2007. **107**(1): p. 39-77.
25. Burman, E. and P. Hansbo, *Edge stabilization for the generalized Stokes problem: a continuous interior penalty method.* Computer methods in applied mechanics and engineering, 2006. **195**(19): p. 2393-2410.

26. Heys, J.J., E. Lee, T.A. Manteuffel, and S.F. McCormick, *On mass-conserving least-squares methods.* SIAM Journal on Scientific Computing, 2006. **28**(5): p. 1675-1693.
27. Heys, J.J., E. Lee, T.A. Manteuffel, and S.F. McCormick, *An alternative least-squares formulation of the Navier–Stokes equations with improved mass conservation.* Journal of Computational Physics, 2007. **226**(1): p. 994-1006.
28. Adler, J. and P. Vassilevski, *Improving Conservation for First-Order System Least-Squares Finite-Element Methods*, in *Numerical Solution of Partial Differential Equations: Theory, Algorithms, and Their Applications*. 2013, Springer. p. 1-19.
29. Heys, J., C. DeGroff, T. Manteuffel, and S. McCormick, *First-order system least-squares (FOSLS) for modeling blood flow.* Medical engineering & physics, 2006. **28**(6): p. 495-503.
30. Manteuffel, T., S. McCormick, J. Nolting, J. Ruge, and G. Sanders, *Further results on error estimators for local refinement with first-order system least squares (FOSLS).* Numerical Linear Algebra with Applications, 2010. **17**(2-3): p. 387-413.
31. D'Elia, M. and A. Veneziani, *Uncertainty quantification for data assimilation in a steady incompressible Navier-Stokes problem.* ESAIM: Mathematical Modelling and Numerical Analysis, 2013. **47**(04): p. 1037-1057.
32. Logg, A., K.-A. Mardal, and G. Wells, *Automated solution of differential equations by the finite element method: The FEniCS book*. Vol. 84. 2012: Springer Science & Business Media.
33. Falgout, R. and U. Yang, *hypre: A library of high performance preconditioners.* Computational Science—ICCS 2002, 2002: p. 632-641.
34. Greenbaum, A., *Iterative methods for solving linear systems*. Vol. 17. 1997: Siam.
35. Niekrasz, M., Y. Ye, L. Rolf, N. Zuhdi, and D. Cooper. *The pig as organ donor for man*. in *Transplantation proceedings*. 1992.

List of Tables

Boundary	Condition	Enforcement
Inlet	$\boldsymbol{n} \times \boldsymbol{v} = 0$	Strong
	$\boldsymbol{n} \cdot \boldsymbol{v} = -1.0 + \left(\frac{x^2 + y^2}{0.25}\right)$	Strong
Outlet	$\boldsymbol{n} \times \boldsymbol{v} = 0$	Strong
	$\boldsymbol{n} \cdot \boldsymbol{v} = -1.0 + \left(\frac{x^2 + y^2}{0.25}\right)$	Strong
Wall	$\boldsymbol{v} = 0$	Strong
Data-Plane	$\boldsymbol{n} \times \boldsymbol{v} = \boldsymbol{f}_{\boldsymbol{data}}$	Weak

Table 1. Boundary conditions used for flow in cylinder problem.

Boundary	Condition	Enforcement	Time Step
Inlet	$\boldsymbol{n} \times \boldsymbol{v} = 0$	Weak	≤ 59 (diastole)
	$\boldsymbol{v} = 0$	Strong	≥ 60
Outlet	$\boldsymbol{v} = 0$	Strong	≤ 74
	$\boldsymbol{n} \times \boldsymbol{v} = 0$	Weak	≥ 75 (systole)
	$\boldsymbol{v} = 0$	Strong	$= 134$
Wall	$\boldsymbol{v} = g_1(t)$	Strong	$all\ Time\ Steps$
PIV-Plane	$\boldsymbol{n} \times \boldsymbol{v} = \boldsymbol{v_{piv}}(\boldsymbol{t})$	Weak	$all\ Time\ Steps$

Table 2. Boundary conditions used for data assimilation left ventricle simulation

Test Problem	$\boldsymbol{\omega}$	Data
1	CFD	Exact + Gaussian Noise
2	Exact + Gaussian Noise	Exact
3	CFD	Echo-PIV experiment

Table 3. Summary of test problems, including the inputs for the data assimilation algorithm (stage 2). The vorticity $\boldsymbol{\omega}$ is the output from stage 1 and may be obtained by solving Navier-Stokes using FEM or from the 'Exact' solution to the Navier-Stokes equations when it is available.

List of Figures

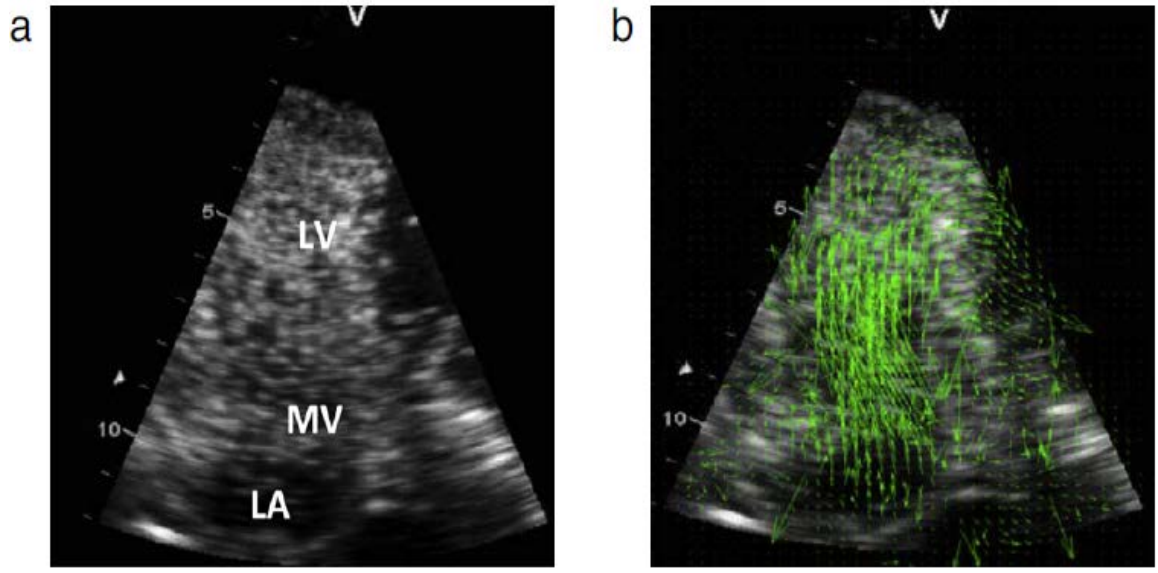


Figure 1. Ultrasound image containing microbubbles in the left ventricle (LV) along with the left atrium (LA) and mitral valve (MV) (a). The corresponding velocity data generated using PIVlab shown in (b) [9]. The velocity data generated by PIVlab is obtained at $time = 0.019$, which is the first frame in the echo-PIV scan.

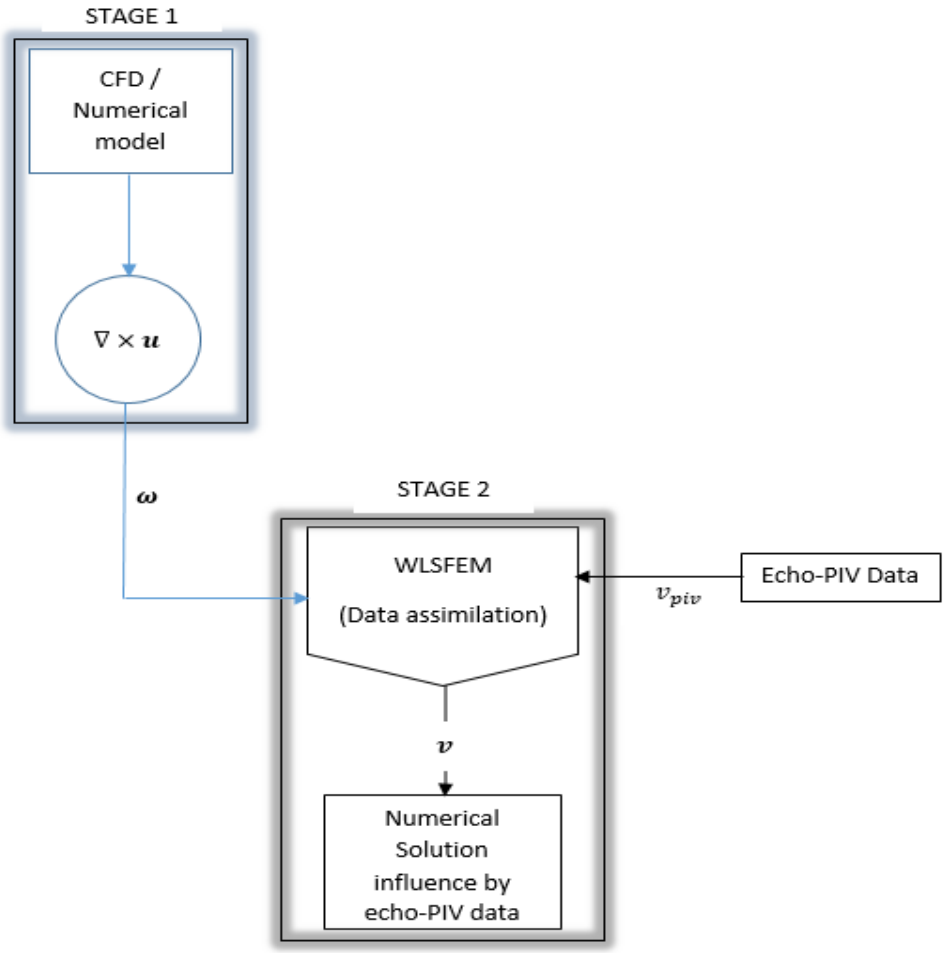


Figure 2. Workflow for new data assimilation approach showing stage 1: solution of the numerical model, and stage 2: assimilation of experiment data to obtain a new numerical solution that is influence and improved by the assimilated data. Variables $\boldsymbol{u}$ and $\boldsymbol{v}$ are the velocity without the influence of assimilated data and with influence from data along with boundary conditions listed in Table 1 and Table 2. $\boldsymbol{\omega}$ is the vorticity field without the influence of data as input to stage 2, and v_{piv} is the experimental data.

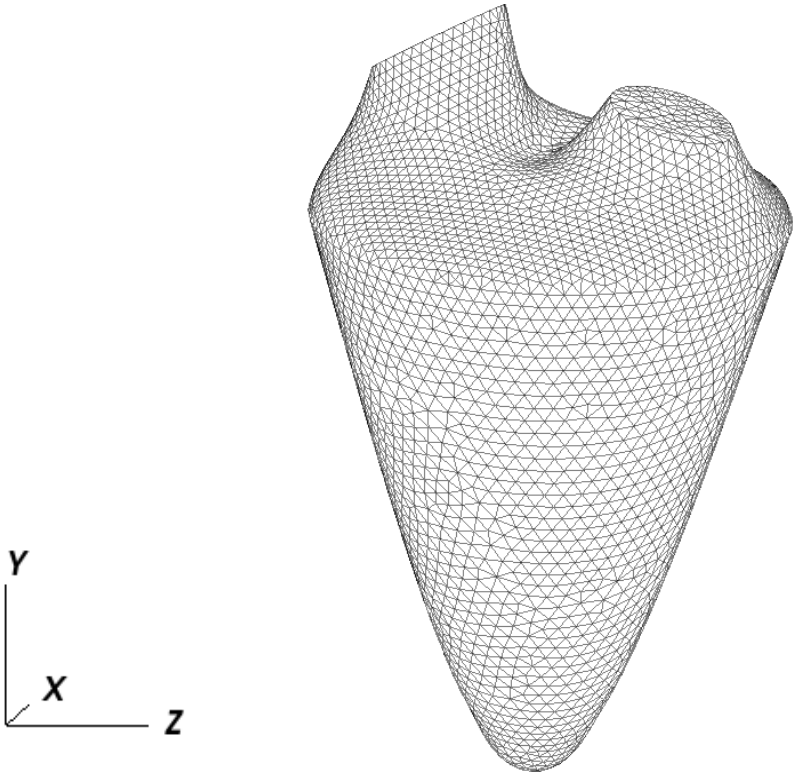
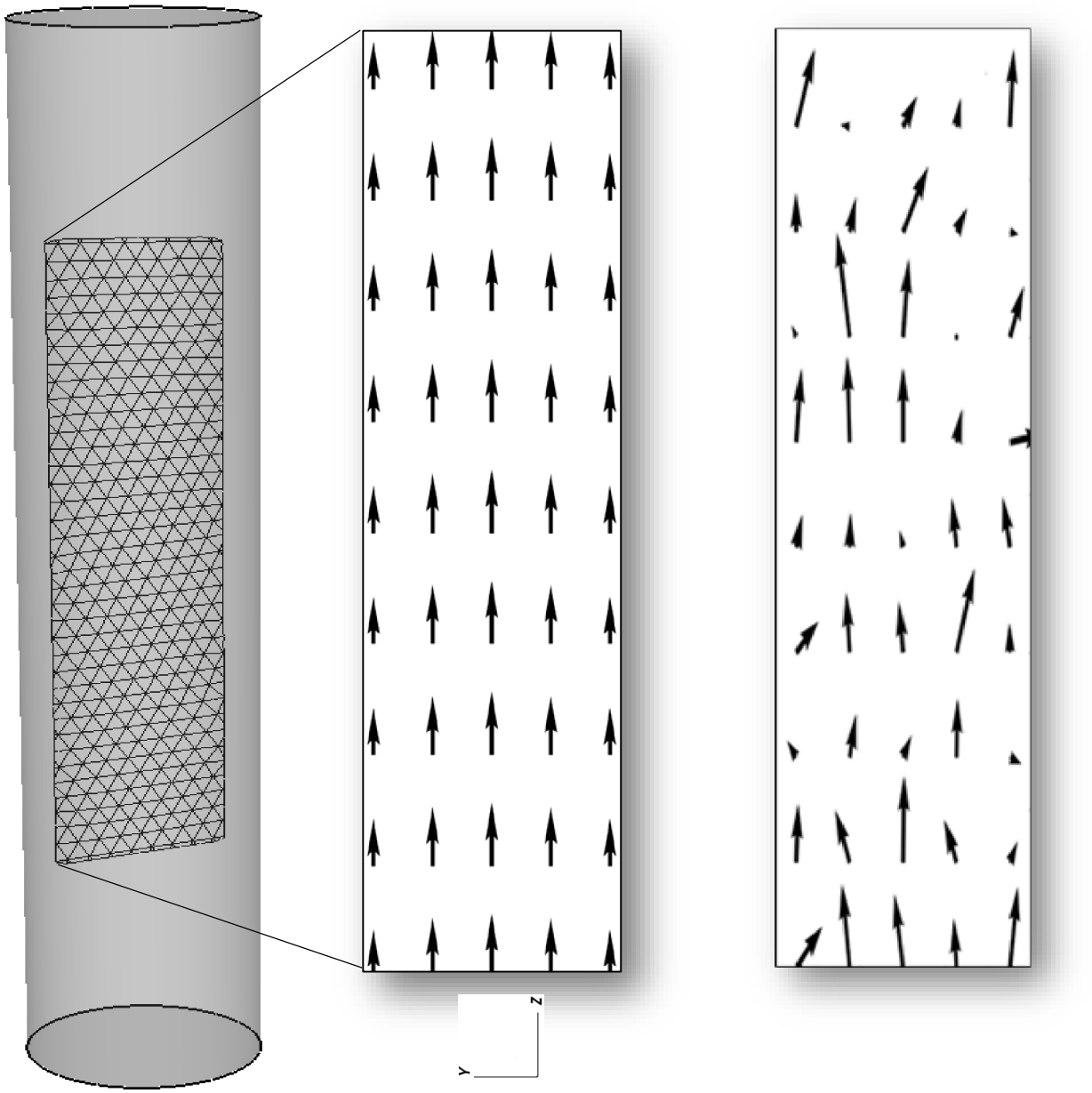


Figure 3. A tetrahedral mesh with approximately 52000 elements of the left ventricle generated using Cubit.



(a) (b) (c)

Figure 4. Cylindrical geometry with an aspect ratio of 1:1:5 with the velocity data plane from 25%-75% of the cylinder. The assimilated velocity data is shown on the right of the figure.

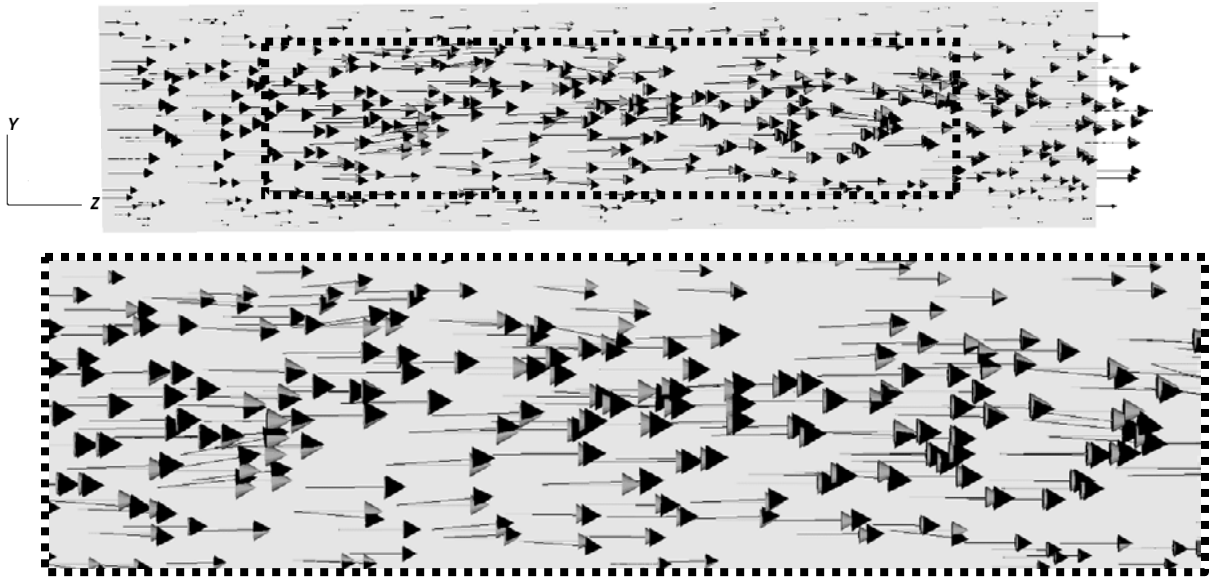
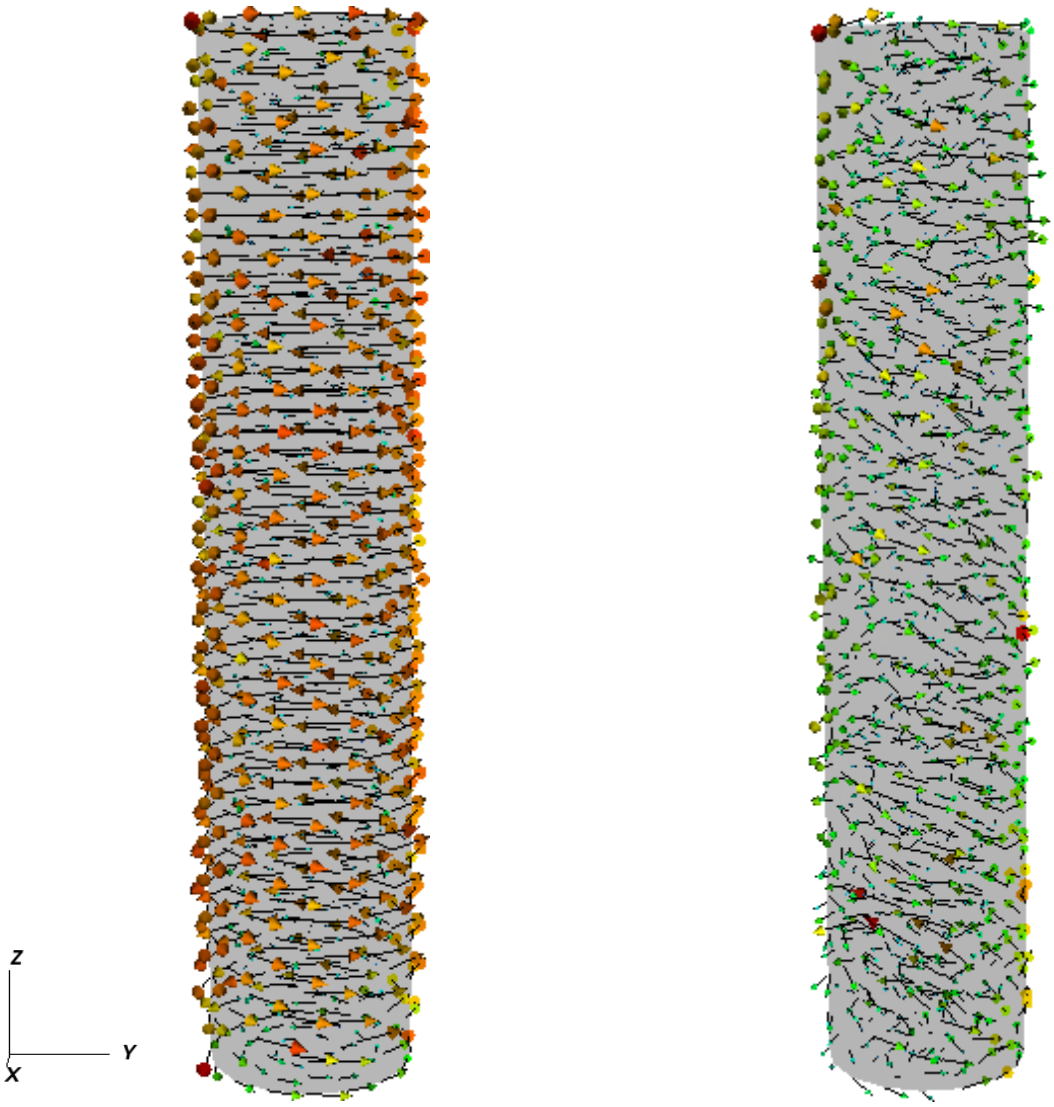


Figure 5. Comparison of velocity with data assimilation (black) and without data assimilation (gray) for mesh with approximately 90000 elements.



(a) (b)

Figure 6. Vorticity field generated from stage 1 with $Re = 1.0$: (a) $\tau = 0.0$, and (b) $\tau = 0.025$.

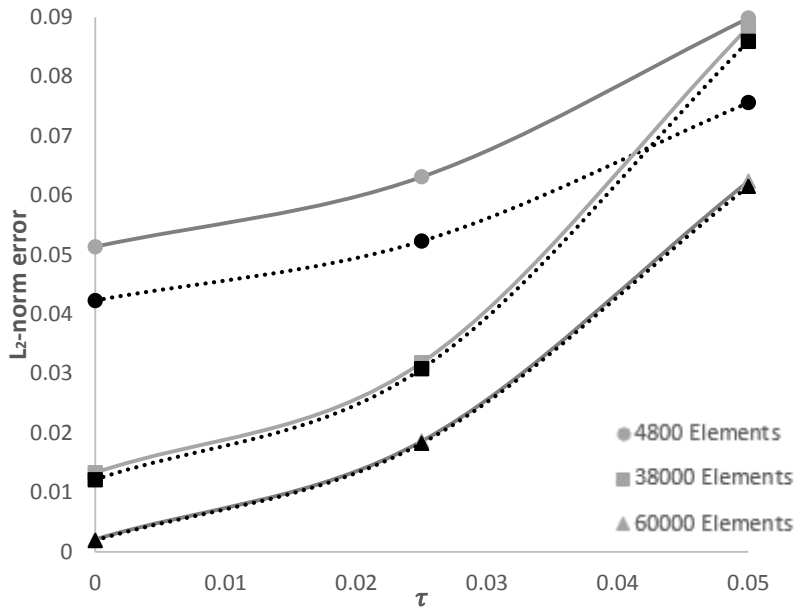
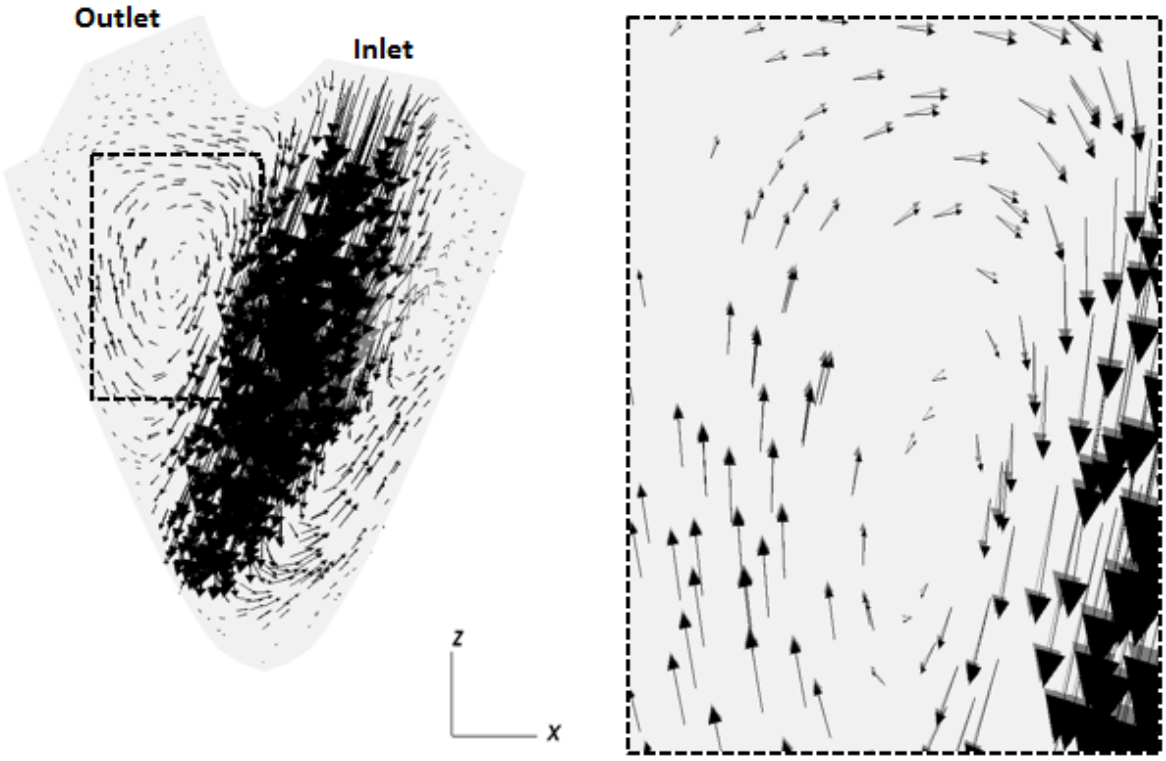
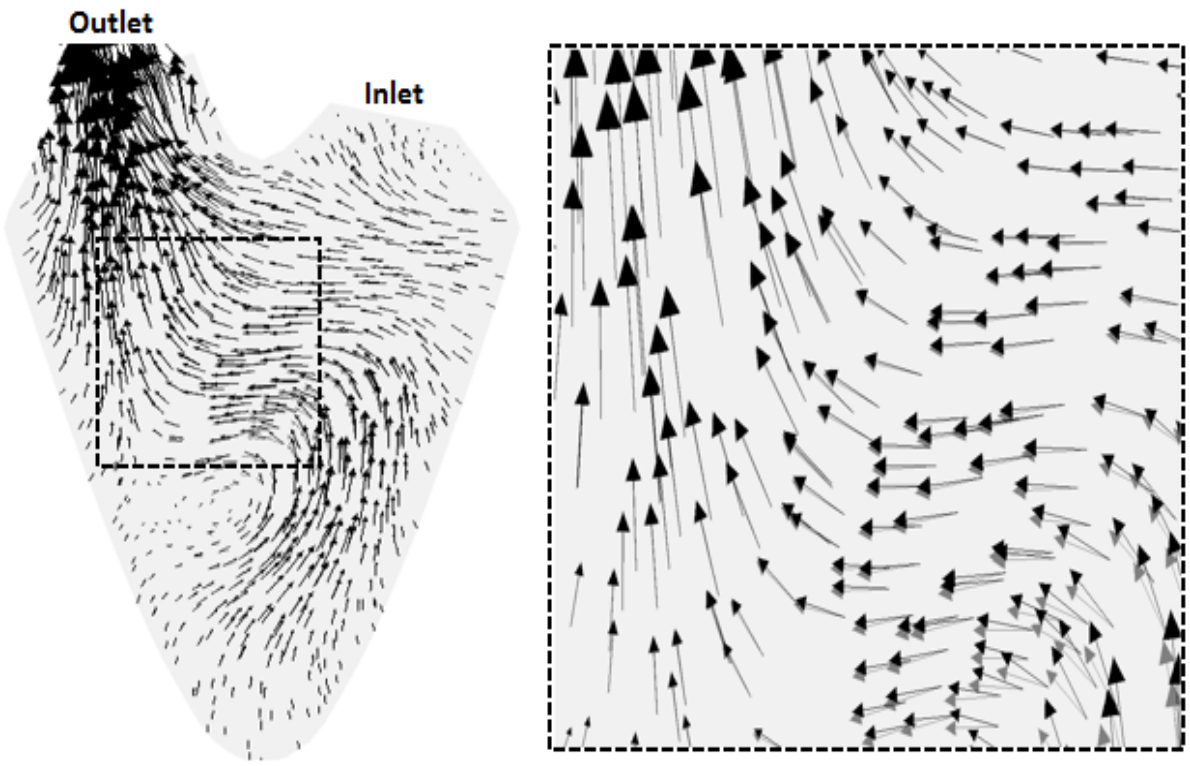


Figure 7. The error in the final numerical solution (after the second stage) as a function of the error in the initial numerical solution from stage 1. The gray points have no assimilated data and the black points include assimilated data. Circle points are an approximately 4800 element mesh, square points are a 38000 element mesh, and triangles are a 60000 element mesh.

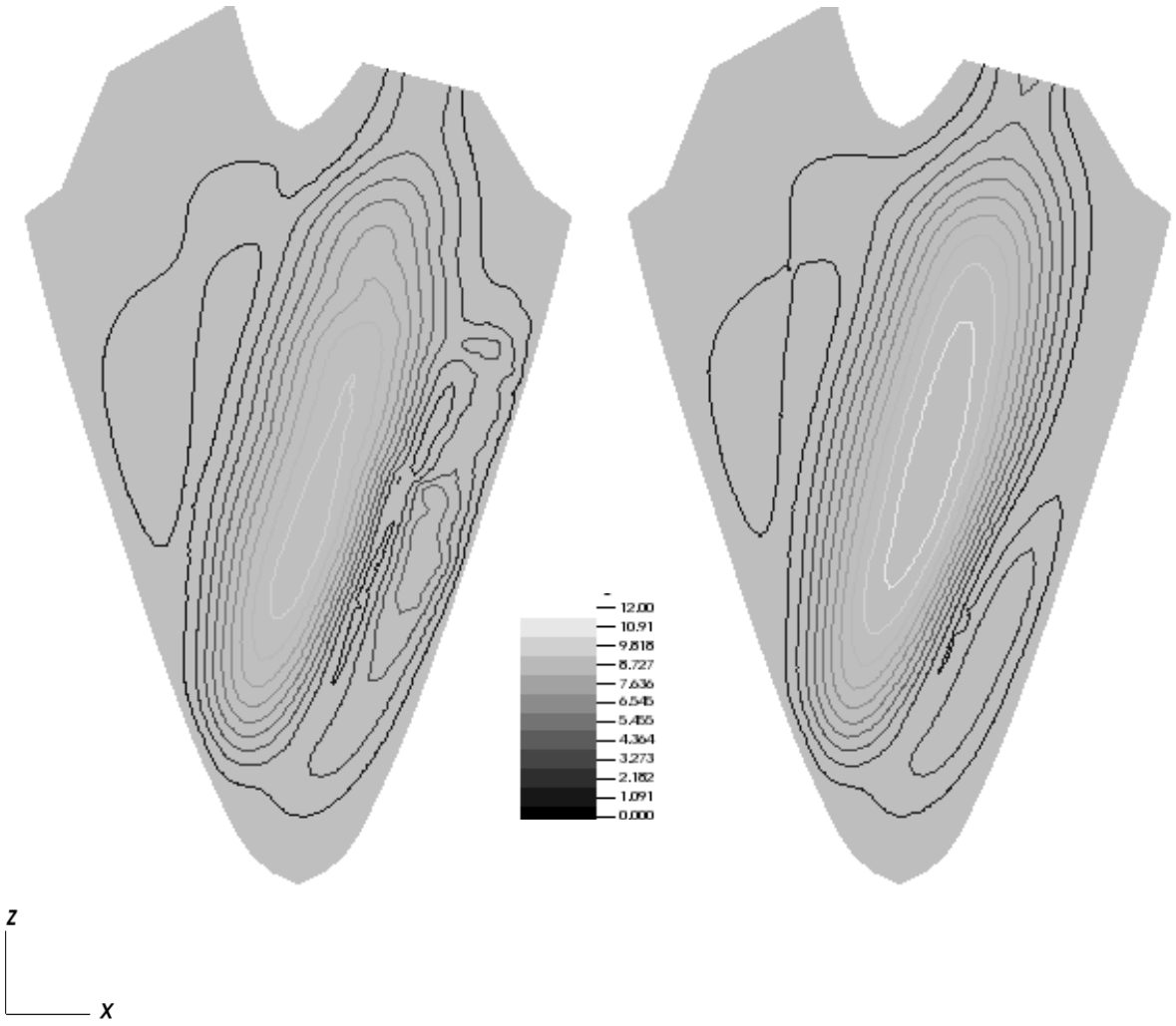


(a) Filling Phase



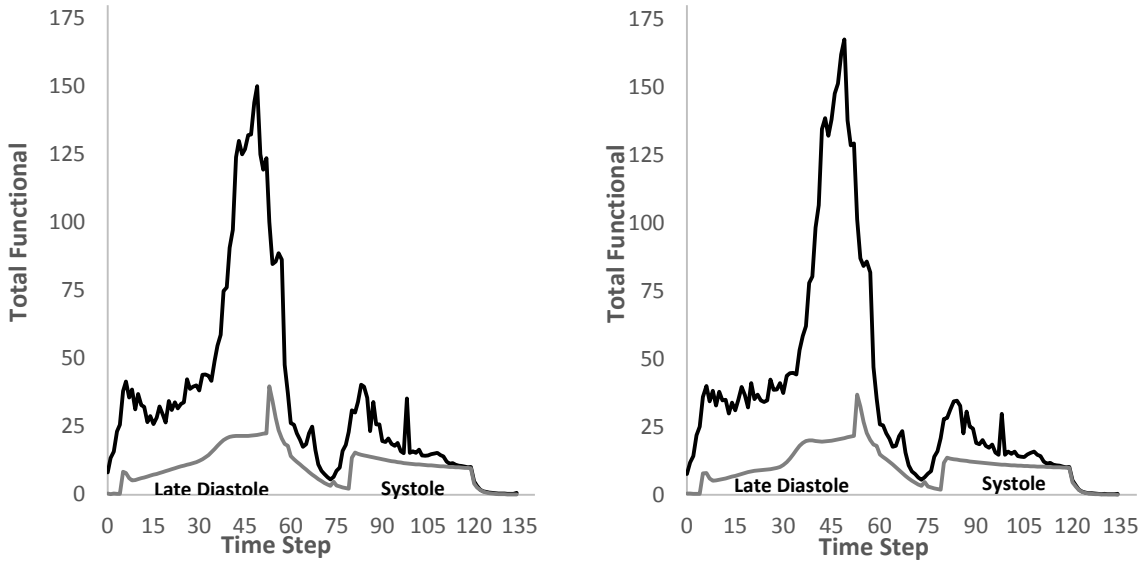
(b) Ejection Phase

Figure 8. Velocity field with assimilated echo-PIV data (black) and without data assimilation (gray) during: (a) late diastole, and (b) systole. The boxed region is shown on the right.



(a) (b)

Figure 9. Velocity contour field with echo-PIV data (a) and without echo-PIV data (b) during late diastole.



(a) (b)

Figure 10. Functional (4) calculated at each time step with echo-PIV data (black) and without data (gray). Mesh with approximately 40000 (a) elements and 52000 (b) elements.