VIII International Conference on Computational Methods for Coupled Problems in Science and Engineering
COUPLED PROBLEMS 2019
E. Oñate, M. Papadrakakis and B. Schrefler (Eds)

COUPLING OF STRUCTURAL SOLVER AND VOLUME-CONSERVING SOLVER FOR FORM-FINDING OF MEMBRANE STRUCTURES SUBJECTED TO PONDING

Navaneeth K Narayanan[1,2,*], Roland Wüchner[2] and Joris Degroote[1,3]

[1] Department of Flow, Heat and Combustion Mechanics, Ghent University
Sint-Pietersnieuwstraat 41, 9000 Ghent, Belgium,
navaneeth.kodunthirappullynarayanan@ugent.be

[2] Chair of Structural Analysis, Technical University of Munich,
Arcisstr. 21, 80333 Munich, Germany

[3]Flanders Make, Belgium

Key words: Volume-conserving solver, ponding, partioned FSI, hydrostatic load

Abstract. The current study deals with coupling of a volume conserving solver and a structural solver to calculate the static deformation of flexible structures under the load of a given volume of water. The volume-conserving solver contains a horizontal plane representing the free surface of the fluid, which is moved in the non-linear iterations to conserve the volume. The Partitioned approach is chosen to have code modularity and reusability with many structural codes.

1 INTRODUCTION

Membrane structures have a wide variety of applications spanning across different engineering disciplines, from the construction of light weight structures in civil engineering to the deployment of parachutes during reentry in aerospace industry. Among the various types of light weight structures, tensioned membrane structures may be the most ubiquitous. These structures have a unique load carrying ability relative to their self-weight due to their large deflection behavior. This at the same time makes them vulnerable to ponding, which is the formation of indentation filled with liquid. There are many aspects of an analysis on membrane structures, such as large deformation analysis, form finding [1], wrinkling [2] and membrane wind interaction [3] but the analysis involving ponding water on a membrane structure is relatively rare. However, there are studies [4, 5] where the hydrostatic load is applied as a follower load on the structure and is solved in the non-linear iterations of the structural solver, but this implementation involves access to the structural solver which may not be possible in many structural codes. Therefore, the

approach of partitioned coupling has been chosen for the present work. An example of using this approach to study ponding on membrane structures is performed by Bown et al [6]. In their work, the authors couple their in house structural code inTENS with their shallow water solver in a partitioned manner to predict and analyze ponding on the membrane structure. However, this is computationally expensive as they couple two transient solvers to analyze what is in fact a steady state phenomenon. The volume-conserving solver proposed in this work with an incremental increase of volume should simulate the filling up of water in the local depression without incorporating the transient behavior.

One main motivation to simulate the ponding in membrane structures is to study the case where the ponding water is coupled with a wind excitation. A real-life example where this proved to be fatal is the Pukkelpop accident (Kiewit, Belgium 2011), where a strong wind interacting with ponding water led to huge swaying of a tent in the vertical direction and eventually resulting in the collapse of the structure. Computational methods to investigate this phenomenon will involve a fluid-structure interaction (FSI) simulation with a lightweight structure, ponding water and wind loads. The method discussed in this work will be used to determine the deformed shape of the structure under the influence of ponding, which will be the starting point of the FSI simulation discussed above. Moreover, based on the literature search simulation involving coupling of ponding water, a membrane structure and wind has not been studied before, therefore it is interesting from a research point of view. The proposed method involves coupling a structural solver and a volume-conserving solver in partitioned manner, in the same way how the effect of wind and ponding on a membrane structure will be studied. The volume-conserving solver consists of a plane representing the free surface of the water. The solver updates its position based on the deformation of the underlying structure in order to conserve a given volume of water, which in turn applies hydro-static loads on the structure. When the convergence is achieved in the partitioned iterations, the deformed shape of the structure is determined. The following sections discuss the proposed volume-conserving solver and how it is coupled to a structural solver in KRATOS [7], an open-source finite element framework. At the end, an example is presented where the static deformation of a flexible structure is calculated due to the accumulation of an incompressible fluid.

2 CONTINUOUS PROBLEM

Let us assume a flexible structure which can be modeled as shell or membrane elements with a suitable material model. The surface of the structure is denoted by $\partial\Omega_s$ which contains a certain volume of incompressible fluid. The free surface of the fluid is horizontal (perpendicular to gravity), denoted by $\partial\Omega_f$. The part of $\partial\Omega_s$ which is below $\partial\Omega_f$ is the fluid-structure interface called wetted surface, denoted by $\partial\Omega_{fs}$,which experiences hydrostatic pressure from the fluid above. In other words, a point S on $\partial\Omega_{fs}$ experiences traction, $\mathbf{t}$ in the form of pressure proportional to the height of the free surface above the point. Mathematically, the traction at a point S with a position vector $\mathbf{x}$ on the structure

can be stated as:

$$\begin{aligned} \mathbf{t} &= \gamma_f[(\mathbf{x}-\mathbf{x}_f)\cdot\mathbf{n}_f]\mathbf{n} && \forall \mathbf{x} \in \partial\Omega_{fs} \\ \mathbf{t} &= \mathbf{0} && \forall \mathbf{x} \in \partial\Omega_s \setminus \Omega_{fs} \end{aligned} \tag{1}$$

where, $\mathbf{x}_f \in \partial\Omega_f$, γ_f is the specific weight of the fluid, $\mathbf{n}$ is the outward unit normal vector at the point S and $\mathbf{x}_f$ is the position vector of any point at the free surface with an unit normal $\mathbf{n}_f$. The expression $(\mathbf{x}-\mathbf{x}_f)\cdot\mathbf{n}_f$ gives the the vertical height of the point S from the free surface. The symbols and terminologies introduced above are clearly shown in figure 1.

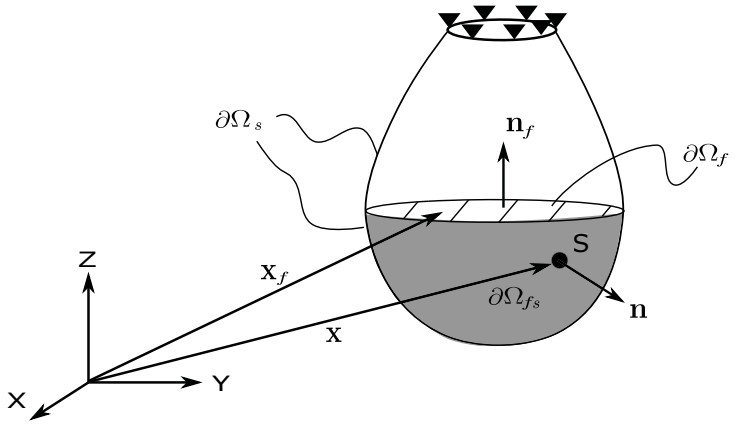


Figure 1: Structure loaded by the hydrostatic pressure from the fluid.

3 VOLUME-CONSERVING SOLVER

The fluid enclosed between Ω_f and Ω_{fs} is incompressible and therefore during the partitioned FSI iterations, the volume of the fluid should always be conserved. This is achieved by an algorithm called volume-conserving solver. It consists of two building blocks: a volume calculation algorithm and the non-linear iterations responsible for updating the free surface to conserve the volume. The equation for calculating the fluid volume using the points on wetted surface can be obtained by using equation 2 and the Gauss divergence theorem. Following some substitutions, we arrive at the expression of volume (V) of the fluid domain (Ω_f) bounded by the wetted surface ($\partial\Omega_{fs}$) and the free surface ($\partial\Omega_f$), given in equation 3.

$$\nabla\cdot[(\mathbf{x}-\mathbf{x}_f)\cdot\mathbf{n}_f\mathbf{n}_f] = 1 \qquad \forall \mathbf{x} \in \Omega_f \tag{2}$$

$$\int_{\Omega_f} \nabla\cdot[(\mathbf{x}-\mathbf{x}_f)\cdot\mathbf{n}_f\mathbf{n}_f]\; dV = \int_{\partial\Omega_{fs}} [(\mathbf{x}-\mathbf{x}_f)\cdot\mathbf{n}_f\mathbf{n}_f]\cdot\mathbf{n}\; dS$$

$$\implies V = \int_{\Omega_f} dV = \int_{\partial\Omega_{fs}} [(\mathbf{x}-\mathbf{x}_f)\cdot\mathbf{n}_f\mathbf{n}_f]\cdot\mathbf{n}\; dS \tag{3}$$

The non-linear iterations to conserve the volume consist of a modification to the Newton's method called leap-frogging Newton, with residual equal to the difference between the current volume, V(calculated using equation 3) and the target volume, V_t. The method is discussed in detail in [9], which consists of a Newton step and followed by a pseudo secant step. The main advantage of this method is that it only involves one derivative evaluation, like Newton's method and yet it can attain cubic convergence. Both methods were tested for volume conservation with a complicated geometry and it was found that the leap-frogging Newton was much more robust and had faster convergence rate than Newton's method. Hence, it was chosen over the other.

To understand the method, consider a function $f(x) \in C^2[a,b]$ with a root x^* in the interval $[a,b]$. Let x_n be the solution at n^{th} iteration. The Newton's step to find the intermediate solution $\tilde{x}_n$ is given as,

$$\tilde{x}_n = x_n - \frac{f(x_n)}{f'(x_n)} \tag{4}$$

In the next step, a secant is constructed from the points $(x_n, f(x_n))$ and $(\tilde{x}_n, f(\tilde{x}_n))$, given by equation 5. The point where the secant line intersects the x-axis is taken as the solution at iteration $n+1$. By substituting equation 4 in equation 5 and performing some algebraic manipulation, one can obtain an expression to find x_{n+1}, given in equation 6. The two steps discussed above are clearly shown in figure 2.

$$y - f(x_n) = \frac{f(x_n) - f(\tilde{x}_n)}{(x_n - \tilde{x}_n)}(x - x_n) \tag{5}$$

$$x_{n+1} = x_n - \frac{(f(x_n))^2}{f'(x_n)\,(f(x_n) - f(\tilde{x}_n))} \tag{6}$$

For the volume-conserving solver, volume $V - V_t$ is the non-linear function f in equation 6. Considering a case where $\mathbf{n}_f$ is along the z-direction, the equation for calculating the volume given in equation 3 simplifies to equation 7, where z and z_f are the z-coordinates of a point in Ω_{fs} and Ω_f, respectively. Clearly, this is a function of z, z_f and $\mathbf{n}$. However, it can be noted that during the volume-conserving process the wetted surface doesn't change, hence the volume depends only on z_f. The derivative of the function, f' can be obtained by deriving the equation 7 by z_f. As given in equation 8, the derivative of the difference between the target and current volume is equal to the area of the free surface, A_f.

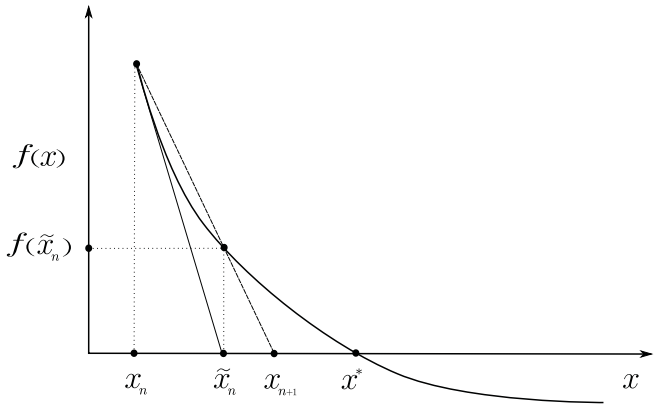


Figure 2: Leap-frogging Newton method to find the root of a non-linear function.

$$f(z_f) = V - V_t = \int_{\partial\Omega_{fs}} (z - z_f)\mathbf{e}_z \cdot \mathbf{n}\, dS - V_t \tag{7}$$

$$f'(z_f) = -\int_{\partial\Omega_{fs}} \mathbf{e}_z \cdot \mathbf{n}\, dS = A_f \tag{8}$$

Using equations 6, 7 and 8, we can write an algorithm for the volume conserving solver as given in algorithm 1, where the free surface is a plane with normal ($\mathbf{n}_f = \mathbf{e}_z$). A radius, r is given as input to give a spatial limit to the volume calculation and hydrostatic load application on the structure.

Algorithm 1 Leap-frogging Newton's method for volume-conservation.

$n = 0$
while $\left\| \frac{V - V_t}{V_t} \right\| > \varepsilon$ **do**
  if $n < n_{max}$ **then**
    Calculate $f(x_n) = V - V_t$ using equation 7, where $x_n = z_f$ at n^{th} iteration.
    Calculate $f'(x_n) = A_f$ using equation 8.
    Move the plane to an intermediate position along $\mathbf{e}_z$ using equation 4.
    Calculate $f(\tilde{x}_n) = \tilde{V} - V_t$ using equation 7.
    Calculate x_{n+1} using equation 6. Move the plane to $z_f = x_{n+1}$
    $n = n + 1$
  end if
end while

4 IMPLICIT PARTITIONED COUPLING

The deformed shape of the structure under the load of given volume of fluid, V_t is obtained once the structural equilibrium equations and boundary conditions given in equation 1 are satisfied with fluid volume, $V = V_t$. The implicit partitioned approach to find the solution consists of iterations involving sequential calls to the structural solver to calculate the structural deformation and volume-conserving solver to conserve the volume. In each iteration, the structural solver gets the traction at the wetted surface based on the plane's position (equation 1), which in turn updates the plane's position based on the structural deformation so that $V = V_t$ (algorithm 1). Convergence accelerators such as Aitken and IQN-ILS [8] are used to achieve faster convergence. In order to explain the algorithm, we need to define some terms mathematically. Let $\mathbf{d}$ and $\mathbf{t}$ be the displacement and traction on the structure, respectively. With this definition the volume-conserving solver ($\mathcal{F}$) and structural solver ($\mathcal{S}$) can be written as:

$$\mathbf{t} = \mathcal{F}(\mathbf{d}) \tag{9}$$

$$\mathbf{d} = \mathcal{S}(\mathbf{t}) \tag{10}$$

The problem of finding the equilibrium shape of the structure under the hydrostatic load of a fixed volume of fluid can be written as a fixed point problem,

$$\mathbf{d} = \mathcal{S} \circ \mathcal{F}(\mathbf{d}) \tag{11}$$

Algorithm 2 Partioned FSI iterations to calculate structural deformation under hydrostatic load.

$k = 0$
$\tilde{\mathbf{d}}^k = \mathcal{S} \circ \mathcal{F}(\mathbf{d}^k)$
$\mathbf{r}^0 = \tilde{\mathbf{d}}^0 - \mathbf{d}^0$
while $\left\|\mathbf{r}^k\right\| > \varepsilon$ **do**
    if $k < k_{max}$ **then**
        $\mathbf{d}^{k+1} = \mathbf{d}^k + \delta\mathbf{d}^k$
        $\tilde{\mathbf{d}}^k = \mathcal{S} \circ \mathcal{F}(\mathbf{d}^k)$
        $\mathbf{r}^{k+1} = \tilde{\mathbf{d}}^k - \mathbf{d}^k$
        $\delta\mathbf{d}^k =$ `Convergence accelerator.ComputeUpdate`$(\mathbf{r}^{k+1}, \mathbf{d}^k)$
        $k = k + 1$
    end if
end while

In the present work, the solution to the problem given in equation 11 is obtained by implicit partitioned coupling, which involves additional coupling iterations between the structural solver $\mathcal{S}$ and volume-conserving solver $\mathcal{F}$. If k represents the iteration number

for the coupling iterations, then the residual of equation 11 at the k^{th} iteration is given by equation 12, where $\mathbf{d}^k$ is the displacement at the k^{th} iteration and $\tilde{\mathbf{d}}^k = \mathcal{S} \circ \mathcal{F}(\mathbf{d}^k)$.

$$\mathbf{r}^k = \tilde{\mathbf{d}}^k - \mathbf{d}^k \tag{12}$$

With all these definitions, we can write the implicit partitioned coupling algorithm for the problem as given in algorithm 2.

5 DISCRETIZED PROBLEM

The structural equilibrium equations with the boundary conditions are generally solved using the finite element method. This involves first discretizing the structure into elements that converts the continuous problem into a discrete set of equations. When the equations are non-linear, an iterative algorithm based on Newton-Raphson is used where the linearized equation given in equation 13 is solved in every iteration to get the solution update until convergence.

$$\mathbf{K}_T \Delta \mathbf{d} = \mathbf{r}_s \tag{13}$$

where, $\mathbf{K}_T$ is the tangent stiffness matrix, $\mathbf{r}_s$ is the residual vector of the discretized structural equations and $\Delta \mathbf{d}$ is the update in the displacements.

In the current work, the structural-solver responsible for performing non-linear iterations is treated as a black-box. It takes the traction in the form of nodal pressures as input and gives the displacements as output. Consequently, this process introduces an error in the elements that are cut by the free surface plane. As shown in figure 3, the nodes of the cut elements that are below the free surface plane receive pressure values proportional to their vertical distance, as given in equation 1. By contrast, the nodes that are above this plane receive zero pressure, which means equation 1 is not satisfied exactly in the cut elements. However, this error should decrease with a finer discretization.

The volume-conserving solver presented in this paper basically involves computation of two quantities, $V - V_t$ and A_f. As evident from the equation 7 and 8, this will involve integration of a scalar quantity in the domain $\partial\Omega_{fs}$, which is performed by numerical integration based on Gauss quadrature in the discretized geometry. This is especially challenging at the cut elements as the elements have to be divided around the free surface plane to resolve the discontinuity. The number and type of divisions will depend on the type of elements. Therefore, it was decided to keep the implementation simple and consider only linear triangular elements for all the simulations. Figure 4 shows the different possibilites and how the elements will be divided to perform the integration. The integration is only performed on the wetted surface and therefore, in the numerical integration only the Gauss points in the shaded region are considered. It can be noted in equation 7 that the volume is calculated by integrating the projection of the distance vector along the surface normal of the wetted surface i.e. $(z - z_f)\mathbf{e}_z \cdot \mathbf{n}$, also shown in figure 4. Therefore, the volume is calculated by first calculating the distance vector, $(z - z_f)\mathbf{e}_z$ for the Gauss

points based on their vertical distance from the plane. Then the weighted sum of their projection with the unit normal vector gives the volume of the enclosed region.

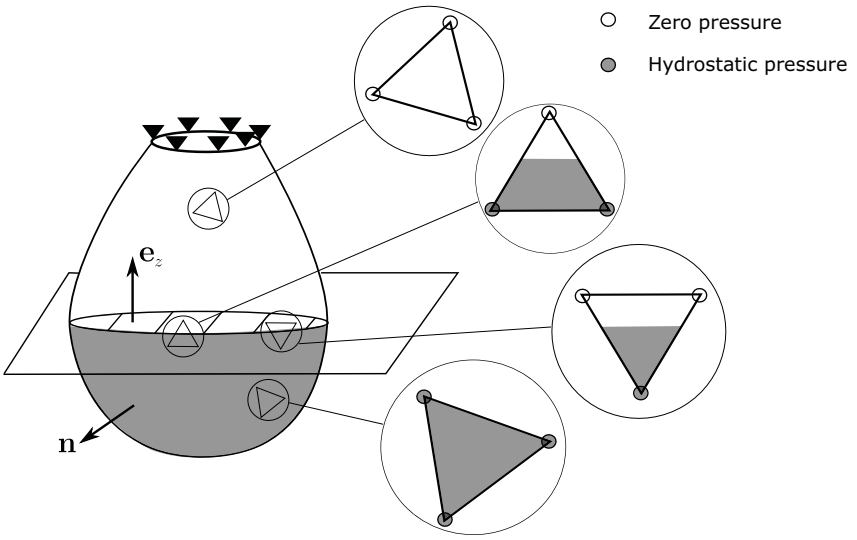


Figure 3: Hydrostatic pressure applied on the discretized structure.

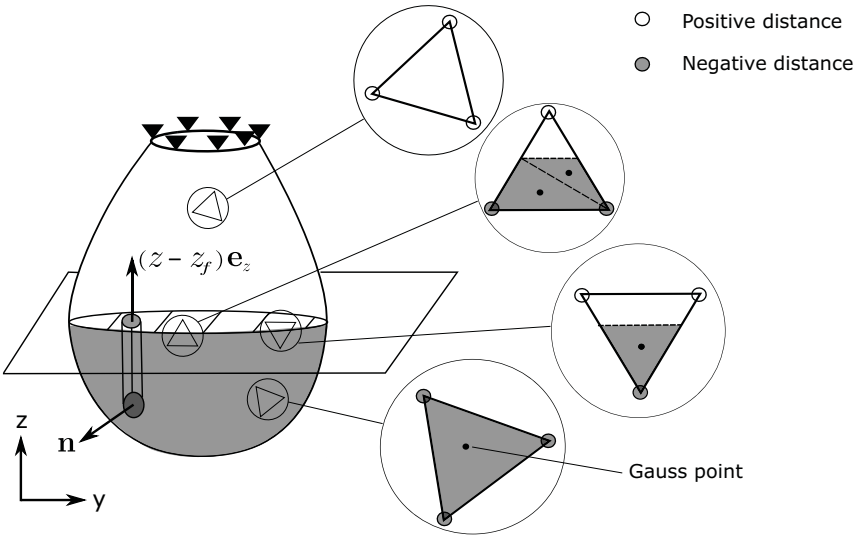


Figure 4: Volume calculation on the discretized structure using Gauss integration.

6 NUMERICAL EXAMPLES

6.1 Volume enclosed between a plane and hemisphere

Before performing numerical simulation of a structural problem, it is important to validate the volume-conserving solver developed in this work. To this end, let us consider a hollow hemisphere of radius $R = 1.0\ m$. The volume enclosed by a horizontal plane at vertical distance of $0.5\ m\ (= z_{ref})$ below the center of hemisphere would be $V = \frac{5\pi}{24} = 0.6545\ m^3$. The volume-conserving solver was run with a target volume, $V_t = 0.6545 m^3$ and $r = 2.0$ and center coinciding with the center of the hemisphere. Three different meshes were considered with element size $0.1\ m$, $0.05\ m$ and $0.025\ m$ and the results (z_{comp}) for the three cases are summarized in table 1. It clearly shows, that the volume-conserving solver was able to calculate the plane's position accurately, limited by the discretization error in the surface representation.

Table 1: Results of volume-conserving solver with $V_t = 0.6545\ m^3$.

Mesh size (m)	$z_{comp}\ (m)$
0.1	-0.4984
0.05	-0.4996
0.025	-0.4999

6.2 Hydrostatic load on a hemisphere

In this numerical example, we consider a flexible hemisphere of radius $R = 1.0\ m$, with thickness $t = 0.05\ m$, fixed at the top edge. The hemisphere is discretized using linear triangular shell elements with the material properties Young's modulus $E = 10^4\ Pa$ and Poisson's ratio $\nu = 0.3$. A fluid of specific weight $\gamma = 200\ N\ m^{-3}$ is gradually filled inside the structure until the fluid volume reaches $V = 1.5\ m^3$. The simulations are performed using implicit partitioned coupling between the structural solver in KRATOS and the implemented volume-conserving solver. Three different meshes with element size $0.1\ m$, $0.05\ m$ and $0.025\ m$ were considered to study the effect of discretization on the result. Inside the coupling iterations, Aitken relaxation is used as a convergence accelerator for all the simulations. As a quantity for comparison, the vertical displacement of point A (u_z) shown in figure 5 is determined for all the meshes. Figure 6 shows the variation of the vertical displacement of point A with the volume of fluid. This data is also presented in table 2 for $V = 1.5\ m^3$, where we can observe that the difference in the vertical displacements between the meshes decreases as the mesh becomes finer. The table also contains the total vertical reaction force at the fixed top edge. If the loading condition is correct, then this value should be equal to the total weight of the fluid ($W = \gamma V$). From table 2, we can observe that it is indeed the case with small errors. One interesting result

is that this value approaches the weight of the fluid as the mesh is more refined. This behaviour indicates that the deviation in the reaction force is due the error in the volume calculation and hydrostatic load application on nodes (in section 5) which are directly related to the discretization of the structure. As an example of results, a deformed geometry for mesh with element size 0.05 m and $V_t = 1.5\ m^3$ is shown in figure 7a and 7b, where the distance and displacement field are plotted, respectively. The distance value in the color legend is scaled to positive and negative values very close to zero, to show the wetted region of the structure.

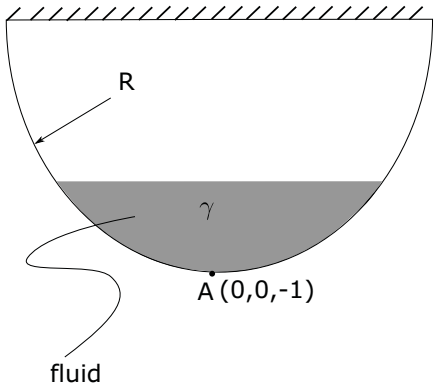


Figure 5: Problem set up for the hydrostatic load on a hemisphere.

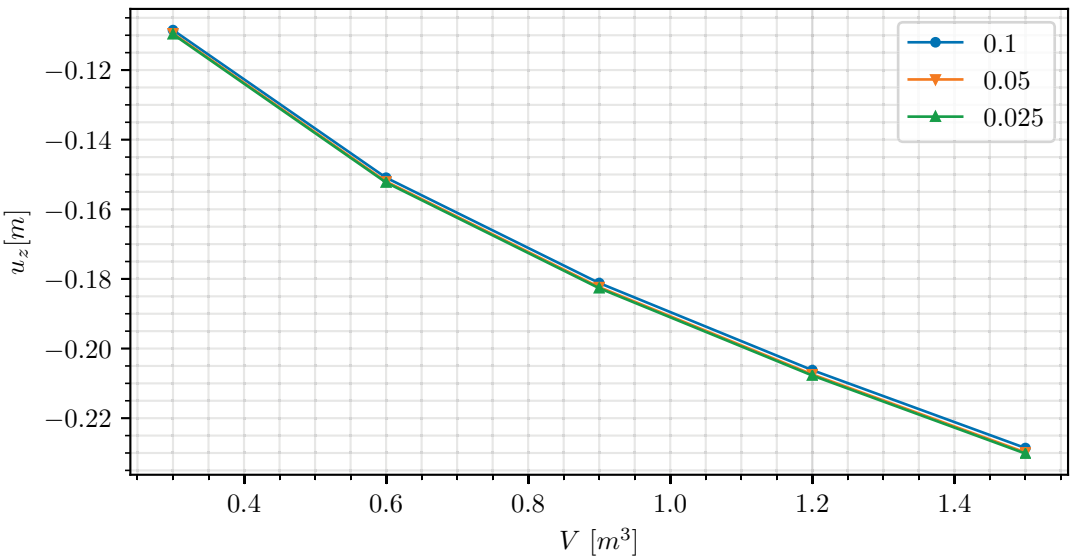
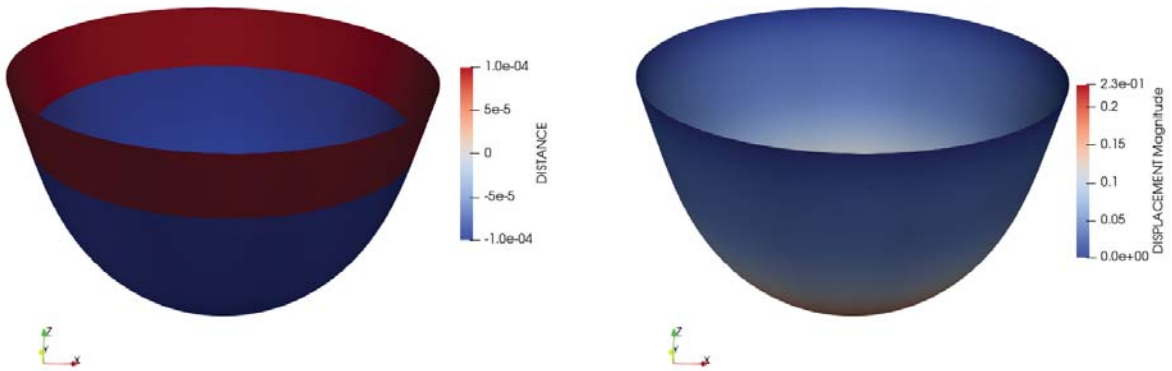


Figure 6: Vertical displacement vs volume of the fluid for three different meshes

Table 2: Comparison of (u_z) and total reaction force (F_z) for different meshes with $V = 1.5\ m^3$

Mesh size (m)	u_z (m)	F_z (N)	W (N)
0.1	-0.228121	300.163	300
0.05	-0.229833	300.050	300
0.025	-0.23017	300.014	300



(a) Element size = 0.1 m (b) Element size = 0.025 m

Figure 7: Distance and displacement results for mesh with element size 0.05 m and $V_t = 1.5\ m^3$.

7 CONCLUSIONS AND OUTLOOK

The main objective of this work was to develop an algorithm for calculating the static deformation of a structure under the hydrostatic load from a fixed volume of fluid. The partition approach was chosen to have code modularity and reusablity with many structural codes. Consequently, a volume conserving-solver was developed which was coupled to a structural solver to achieve the goal. This procedure will be used in the future to find the shape of the membrane due to ponding. Further, the obtained shape will serve as a starting point for performing FSI simulations with membrane, ponding water and wind loads.

8 ACKNOWLEDGEMENT

The authors gratefully acknowledge the funding of the Research Foundation – Flanders (FWO) for this work (project number G086517N)

REFERENCES

[1] K-U. Bletzinger, R. Wüchner, F. Daoud, and N. Camprubi. Computational methods for form finding and optimization of shells and membranes. *Computer Methods in Applied Mechanics and Engineering* (2005) **194(30)**:3438-3452.

[2] A. Jarasjarungkiat, R. Wüchner, K-U. Bletzinger. A wrinkling model based on material modification for isotropic and orthotropic membranes. *Computer Methods in Applied Mechanics and Engineering* (2008) **197(6)**:773-788.

[3] G. De Nayer, A. Apostolatos, J.N. Wood, R. Wüchner and M. Breuer. Numerical studies on the instantaneous fluid–structure interaction of an air-inflated flexible membrane in turbulent flow. *Journal of Fluids and Structures* (2018) **82**:577-609.

[4] T. Rumpel, and K. Schweizerhof. Hydrostatic fluid loading in non-linear finite element analysis. *International Journal for Numerical Methods in Engineering* (2004) **59(6)**:849-870.

[5] C. Hoareau and J.-F Deü. Non-linear finite element analysis of an elastic structure loaded by hydrostatic follower forces. *Procedia Engineering* (2017) **199**:1302-1307.

[6] A. Bown, T. Makin and D. Wakefield. Beyond Static Analysis: Investigation of Membrane Structure Performance Using Time Stepping, Transient and Progressive Analyses. *Procedia Engineering* (2016) **155**:1877-7058.

[7] P. Dadvand, R. Rossi and E. Oñate. An object-oriented environment for developing finite element codes for multi-disciplinary applications. *Archives of computational methods in engineering* (2010) **17(3)**:253–297.

[8] J. Degroote. Partitioned Simulation of Fluid-Structure Interaction. *Archives of Computational Methods in Engineering* (2013) **20(3)**:185-238.

[9] J.B. Kasturiarachi. Leap-frogging Newton's method. *International Journal of Mathematical Education in Science and Technology* (2002) **33(4)**:521-527