

An Extended Partitioned Method for Conservative Solid-Fluid Coupling

MUZAFFER AKBAY, University of California, Riverside, USA

NICHOLAS NOBLES, University of California, Riverside, USA

VICTOR ZORDAN, Clemson University, USA

TAMAR SHINAR, University of California, Riverside, USA

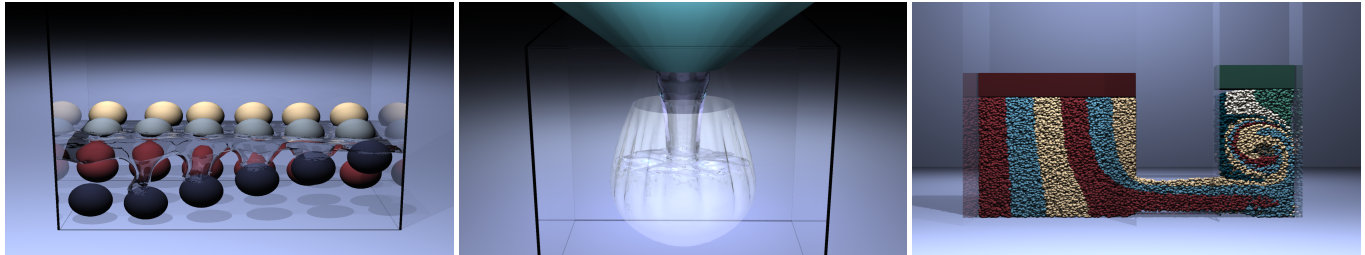


Fig. 1. Left: Rigid balls of varying densities are dropped into a pool of water, achieving correct buoyancy. Center: A thin, elastic balloon is stressed and filled by a fast stream of water. Right: Pressure applied to a small piston of a hydraulic press is transmitted undiminished to a large piston through a fully enclosed and incompressible medium.

We present a novel extended partitioned method for two-way solid-fluid coupling, where the fluid and solid solvers are treated as black boxes with limited exposed interfaces, facilitating modularity and code reusability. Our method achieves improved stability and extended range of applicability over standard partitioned approaches through three techniques. First, we couple the black-box solvers through a small, reduced-order monolithic system, which is constructed on the fly from input/output pairs generated by the solid and fluid solvers. Second, we use a conservative, impulse-based interaction term to couple the solid and fluid rather than typical pressure-based forces. We show that both of these techniques significantly improve stability and reduce the number of iterations needed for convergence. Finally, we propose a novel boundary pressure projection method that allows for the partitioned simulation of a fully enclosed fluid coupled to a dynamic solid, a scenario that has been problematic for partitioned methods. We demonstrate the benefits of our extended partitioned method by coupling Eulerian fluid solvers for smoke and water to Lagrangian solid solvers for volumetric and thin deformable and rigid objects in a variety of challenging scenarios. We further demonstrate our method by coupling a Lagrangian SPH fluid solver to a rigid body solver.

CCS Concepts: • **Computing methodologies** → **Physical simulation**;

Additional Key Words and Phrases: solid-fluid coupling, partitioned simulation, reduced-order model

Authors' addresses: Muzaffer Akbay, University of California, Riverside, 900 University Ave. Riverside, California, 92521, USA, akbaym@cs.ucr.edu; Nicholas Nobles, University of California, Riverside, 900 University Ave. Riverside, California, 92521, USA, nnobl002@ucr.edu; Victor Zordan, Clemson University, Clemson University, Clemson, South Carolina, 29634, USA, vbz@g.clemson.edu; Tamar Shinar, University of California, Riverside, 900 University Ave. Riverside, California, 92521, USA, shinar@cs.ucr.edu.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

© 2018 Copyright held by the owner/author(s). Publication rights licensed to ACM.

0730-0301/2018/8-ART86 \$15.00

<https://doi.org/10.1145/3197517.3201345>

ACM Reference Format:

Muzaffer Akbay, Nicholas Nobles, Victor Zordan, and Tamar Shinar. 2018. An Extended Partitioned Method for Conservative Solid-Fluid Coupling. *ACM Trans. Graph.* 37, 4, Article 86 (August 2018), 12 pages. <https://doi.org/10.1145/3197517.3201345>

1 INTRODUCTION

Physically-based simulation of coupled fluids and solids has been an area of active research in recent years, both in computer graphics and computational physics, and many solution approaches have been investigated [Hou et al. 2012]. The techniques generally fall into two categories, monolithic and partitioned approaches. In monolithic approaches, equations for the solid and fluid are combined into one system and solved simultaneously, implicitly coupling internal solid and fluid forces to the interaction forces between them. On the other hand, partitioned approaches employ separate solvers for the solid and fluid and interleave calls to each solver to achieve either a weak coupling, or if iterated to convergence, a strong coupling of the solvers.

Both the monolithic and partitioned approaches have advantages and disadvantages in the simulation of two-way solid-fluid coupling. Monolithic approaches have the advantage of better stability, as stiff coupling terms can be resolved simultaneously with other forces in the system, but they also incur significant cost in terms of development and do not fully leverage existing codes. On the other hand, partitioned approaches have the advantage of being able to reuse previously developed fluid and solid solvers in a black-box fashion as long as they expose appropriate interfaces. However, they suffer from poorer stability properties, and in challenging scenarios, they may require many iterations per time step or worse, fail to converge [Banks et al. 2014]. When strong coupling is desired, a large number of iterations may be required for convergence, negatively impacting performance. Despite their drawbacks, partitioned approaches remain an important class of methods due to the appeal of reusing existing software, and recent work on partitioned

approaches is aimed at improving their stability properties [Banks et al. 2014; Degroote et al. 2010; Haelterman et al. 2016]. We believe that with increasing variety of specialized solvers for fluids and solids, techniques for coupling disparate solvers without significant code rewriting are desirable.

In this work, we present a novel extended partitioned method (XPM) for two-way solid-fluid coupling of incompressible fluids to rigid and deformable solids and shells. While this approach does not outperform monolithic methods, which are generally more efficient, it does mitigate several drawbacks of partitioned schemes. Our approach builds on a framework first proposed in [Gerbeau and Vidrascu 2003] and [Vierendeels 2006; Vierendeels et al. 2007], which seeks strongly coupled solutions through a partitioned solve. Rather than simple interleaved calls to the fluid and solid solvers, potentially with underrelaxation, a monolithic reduced-order model of the system is used as an intermediate layer to generate improved boundary conditions for the individual fluid and solid solvers, thus accelerating convergence and improving stability. Whereas [Gerbeau and Vidrascu 2003] used a reduced model based on a simplified physical problem, [Vierendeels et al. 2007] constructs reduced models on the fly by computing local least-squares estimates of the solver Jacobians. In this paper, we follow the basic framework introduced in [Vierendeels 2006; Vierendeels et al. 2007] for deformable bodies, extending it to free surfaces, thin shells, and rigid bodies. Furthermore, we propose the use of a conservative, impulse-based interaction term, first developed in the context of the monolithic approach [Robinson-Mosher et al. 2008], instead of the pressure-based forces typically used in partitioned approaches. Finally, we propose a novel solution to the partitioned simulation of incompressible fluid regions fully enclosed by a dynamic solid, a scenario that has previously been problematic [Küttler et al. 2006]. We demonstrate that these novel contributions give stability and performance benefits over more standard partitioned approaches and extend the scope of scenarios that can be effectively solved in a partitioned fashion.

In summary, our partitioned method for two-way coupling of incompressible fluids to solids includes the following novel contributions:

- Extending the reduced-model interface approach of [Vierendeels et al. 2007] to free surfaces, thin shells, and rigid bodies, demonstrating the benefits of this general approach on a variety of solid-fluid coupling scenarios.
- Proposing the use of a conservative, impulse-based interaction term rather than the standard pressure-based forces typically used in partitioned approaches, resulting in improved stability and accelerated convergence.
- Proposing a novel boundary pressure projection method for solving incompressible fluid regions fully enclosed by dynamic solids, that is significantly simpler than previous approaches, extending the range of scenarios that can be simulated using a partitioned scheme.

2 RELATED WORK

Computer graphics researchers have developed a variety of approaches to coupling Eulerian fluids with deformable and rigid Lagrangian solids. Monolithic approaches include a nonsymmetric

linear system to capture the two-way interactions between fluids and deformable solids [Chentanez et al. 2006], a symmetric system capturing the interaction of smoke and rigid bodies [Klingner et al. 2006], a variational formulation based on kinetic energy minimization for coupling fluids to rigid bodies [Batty et al. 2007], a symmetric, momentum-conserving method for coupling fluids to volumetric and thin deformable and rigid bodies [Robinson-Mosher et al. 2008], a positive-definite formulation of that system [Robinson-Mosher et al. 2011], two-way coupling of fluids to reduced deformable bodies [Lu et al. 2016], and a cut-cell method which implicitly couples fluid pressure with solid elasticity and damping [Zarifi and Batty 2017]. Implicit coupling of solids and fluids was also achieved through a fully Eulerian treatment of both [Teng et al. 2016].

Partitioned approaches have also received considerable attention [Degroote 2013; Hou et al. 2012]. One common method is that the solid solver provides velocity boundary conditions to the fluid solver, and the fluid solver provides pressure-based forces to the solid solver (also referred to as a Dirichlet-Neumann decomposition). Separate calls to the fluid and solid solvers are interleaved, treating each as a black box, to achieve either a weak coupling, or if iterated to convergence, a strong coupling of the solvers. A partitioned approach to coupling fluid and rigid bodies was proposed in [Carlson et al. 2004], where additional forces are applied to the fluid velocity to enforce a rigid motion. In [Guendelman et al. 2005], thin deformable and rigid shells were weakly coupled to smoke and water in an interleaved fashion.

Partitioned approaches may suffer from stability issues under challenging scenarios such as high mass ratio of fluid to solid [Causin et al. 2005]. In the simplest interleaved approach, one might try to reduce the time step size to achieve better stability. However, it has been shown that the stability condition on the time step is not always achievable for certain problem parameters [Le Tallec and Mouro 2001; Vierendeels et al. 2011]. Several methods attempt to mitigate the stability problems of partitioned solvers, for example, by incorporating relaxation into the fixed-point iteration [Küttler and Wall 2008]. When Gauss-Seidel subiterations with underrelaxation are used, a very small underrelaxation factor may be needed for convergence [Vierendeels et al. 2011].

Another approach to improving stability of partitioned schemes is the use of a reduced monolithic model as an intermediary between the solvers [Degroote 2013; Degroote et al. 2010; Gerbeau and Vidrascu 2003; Vierendeels et al. 2007]. We build on the method introduced in [Vierendeels 2006; Vierendeels et al. 2007], where local, linear reduced-order models of the solid and fluid solvers are constructed on the fly using the input/output pairs collected from the solver calls. The reduced order models are then solved in a monolithic fashion to generate boundary conditions for each black-box solver, improving stability and reducing iterations counts. We note that this approach is closely related to quasi-Newton methods for partitioned solid-fluid coupling [Degroote 2013; Degroote et al. 2009, 2010; Haelterman et al. 2016], in which the black-box solver Jacobians are similarly approximated, and which have also been shown to improve performance of partitioned schemes.

Many methods have also been developed for the coupling of Lagrangian fluids to deformable and rigid solids. Müller et al. simulated fluids using the smoothed particle hydrodynamics (SPH) method,

interacting with deformable solids, modeling the no-penetration and no-slip boundary conditions as well as collisions via a momentum exchange [Müller et al. 2004]. Direct forcing to achieving boundary conditions was done in [Becker et al. 2009], and [Oh et al. 2009] proposed a conservative, impulse-based interaction between rigid bodies and SPH fluids. Particle-based granular flows were two-way coupled with solids in [Narain et al. 2010]. A momentum-conserving approach for coupling SPH to thin and volumetric rigid bodies was proposed in [Akinci et al. 2012], and coupling to elastic solids was presented in [Akinci et al. 2013]. Additionally, implicit two-way coupling of solids and fluids is naturally handled in frameworks that treat both in a unified fashion, such as with a fully Lagrangian approach [He et al. 2012; Keiser et al. 2005; Shao et al. 2015; Solenthaler et al. 2007], or a hybrid approach [Stomakhin et al. 2014].

A variety of reduced-order models have been developed for physics-based animation, for example [Treuille et al. 2006] for reduced fluid simulation, and [Barbič et al. 2009] for reduced deformable solid simulation. In contrast to these and related approaches, our approach uses the reduced models to stabilize and accelerate convergence of the iterated partitioned coupling, while the final fluid and solid state in each time step is determined by the full solvers.

3 SOLID-FLUID COUPLING

The fluid and solid are coupled at their interface Γ through the boundary conditions:

I: Kinematic (no-slip) condition $\mathbf{u} = \mathbf{V}$,

II: Dynamic condition $(-p\mathbf{I} + \mu(\nabla\mathbf{u} + \nabla\mathbf{u}^T))\mathbf{n} = \Sigma_S\mathbf{n}$.

where $\mathbf{u}$, p , μ , and $\mathbf{n}$ are the fluid velocity, pressure, dynamic viscosity, and outward unit normal, respectively, and $\mathbf{V}$ and Σ_S are the solid velocity and stress tensor, respectively. The dynamic condition, or balance of tractions at the interface, implies that there is a pair of action/reaction forces at the interface, equal in magnitude and opposite in direction, inspiring a conservative impulse-based approach. In this work, we neglect the viscous component of the fluid stress on the interface.

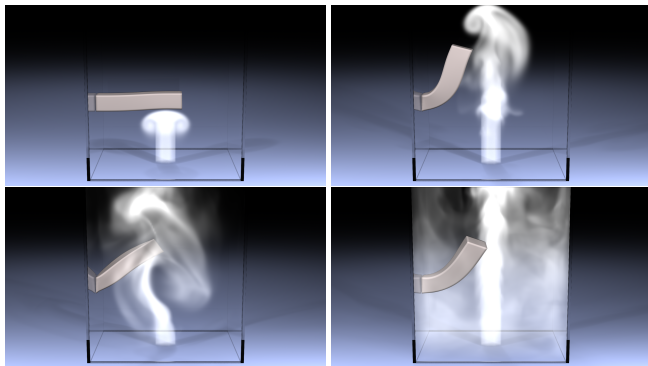


Fig. 2. A light piece of deformable foam interacts with a jet of smoke. (fluid grid size: $100 \times 150 \times 100$)

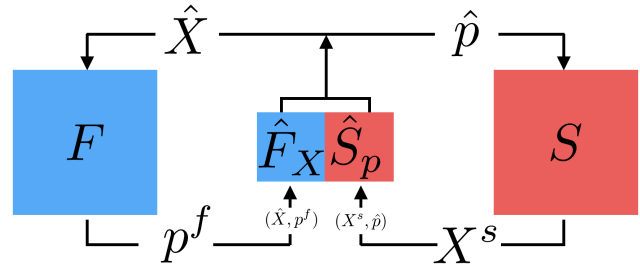


Fig. 3. Schematic representation of the partitioned coupling of black-box fluid (F) and solid (S) solvers through the reduced model interface. Reduced Jacobians $\hat{F}_X$ and $\hat{S}_p$ are computed from input/output pairs of F and S and tightly coupled in a low-rank monolithic system to generate improved boundary conditions to F and S.

4 REDUCED MODEL INTERFACE

Rather than simple Gauss-Seidel iterations between the fluid and solid solvers, we couple them through the Reduced Model Interface (RMI) (Figure 3). In this approach, the boundary conditions passed to the individual solvers are generated by an intermediate solver, a monolithic system coupling reduced models of the solid and fluid. Like other partitioned methods, this allows for the use of the fluid and solid solvers as black boxes. Input/output pairs collected from invocations of the fluid and solid solvers are used to approximate their Jacobians and build the monolithic reduced system. The RMI system is smaller than a full monolithic system, as it contains no more than the number of interface variables. Although dense, as opposed to the sparse monolithic discretizations such as [Robinson-Mosher et al. 2008], it is also low-rank, typically capturing a small number of modes, and this is exploited in the matrix computations to avoid forming dense matrices the size of the number of interface variables. Stability analysis of RMI coupling for unsteady flow in an elastic tube showed that unstable components appear during the first iterations and are implicitly coupled by the reduced model interface [Degroote et al. 2008].

4.1 Reduced monolithic system

We follow the approach of [Vierendeels 2006; Vierendeels et al. 2007] for coupling a fluid and volumetric deformable solid and extend it to rigid bodies, free surfaces, and thin shells. The black-box fluid solver is represented as

$$p = F(X, V(X)), \quad (1)$$

taking as input the solid node positions and effective velocities and returning the fluid pressures on the solid. The black-box solid solver is represented as

$$X = S(p), \quad (2)$$

taking as input fluid pressures and returning the solid simulation state.

The approach is illustrated in Figure 3. We seek to construct a reduced-order model of the fluid and solid solvers, the RMI, to act as intermediary, generating improved boundary condition inputs to the black-box solvers. Since the RMI is reduced-order, it can be efficiently solved in a monolithic fashion. Its cost is much less than that of the individual fluid and solid solvers, as shown in Section 7.

We next describe the RMI. During each time step, at iteration i , a call to the fluid solver (1) yields an input/output pair (X_i^f, p_i^f) , and a call to the solid solver (2) yields an output/input pair (X_i^s, p_i^s) . After k iterations, the linear, reduced-order models of the fluid and solid solvers are given by

$$\hat{p} = p_k^f + \hat{F}_X(\hat{X} - X_k^f), \quad (3)$$

$$\hat{X} = X_k^s + \hat{S}_p(\hat{p} - p_k^s), \quad (4)$$

respectively, where $\hat{F}_X$ is an approximation to the Jacobian of F with respect to X , and $\hat{S}_p$ is an approximation to the Jacobian of S with respect to p , and $(\hat{X}, \hat{p})$ are unknowns. Moving unknown terms to the left hand side, this becomes a monolithic set of equations

$$\begin{pmatrix} I & -\hat{F}_X \\ -\hat{S}_p & I \end{pmatrix} \begin{pmatrix} \hat{p} \\ \hat{X} \end{pmatrix} = \begin{pmatrix} p_k^f - \hat{F}_X X_k^f \\ X_k^s - \hat{S}_p p_k^s \end{pmatrix}. \quad (5)$$

As only one of $\hat{p}$ or $\hat{X}$ is needed when invoking the solid or fluid solver, they may be obtained individually by applying block Gaussian elimination to Equation (5) as

$$\hat{X} = (I - \hat{S}_p \hat{F}_X)^{-1} [X_k^s + \hat{S}_p (p_k^f - p_k^s - \hat{F}_X X_k^f)], \quad (6)$$

$$\hat{p} = (I - \hat{F}_X \hat{S}_p)^{-1} [p_k^f + \hat{F}_X (X_k^s - X_k^f - \hat{S}_p p_k^s)]. \quad (7)$$

4.2 Constructing the reduced Jacobians

Next, we describe the computation of the reduced model Jacobians $\hat{F}_X$ and $\hat{S}_p$. They are constructed from scratch at each iteration, using data from all calls to the fluid or solid solver in the current time step. The data pairs (X_i^f, p_i^f) and (X_i^s, p_i^s) , $i = 1, \dots, k$ are collected from the fluid and solid solvers, respectively, over k iterations. Note that the RMI layer translates between the disparate discretizations of the solvers. In our examples, interface variables are located at solid nodes, and fluid variables are mapped to/from the nodes through interpolation and/or conservative distribution operators.

From the k fluid solver data pairs, we construct a position difference matrix $V = (\Delta x_1^f \dots \Delta x_{k-1}^f)$, where $\Delta x_j^f = X_k^f - X_j^f$, and a pressure difference matrix $P = (\Delta p_1^f \dots \Delta p_{k-1}^f)$ where $\Delta p_j^f = p_k^f - p_j^f$. A displacement ΔX is approximated by the vectors in V by solving the least squares problem $\min_\alpha \|\Delta X - V\alpha\|_2$. Taking the QR factorization of V ,

$$V = QR, \quad R = \begin{bmatrix} R_1 \\ 0 \end{bmatrix}, \quad Q = [Q_1 \quad Q_2],$$

it follows that

$$\alpha = R_1^{-1} Q_1^T \Delta X,$$

and setting the pressure differences consistently with the displacements as $\Delta p = P\alpha$, we get

$$\Delta p = PR_1^{-1} Q_1^T \Delta X.$$

The Jacobian $\hat{F}_X$ is finally determined as

$$\hat{F}_X = \frac{\partial p}{\partial x} = \frac{\partial p}{\partial \alpha} \frac{\partial \alpha}{\partial x} = PR_1^{-1} Q_1^T.$$

In the same manner, the approximate Jacobian of the solid solver $\hat{S}_p$ can be computed. Input/output pairs gathered from the solid

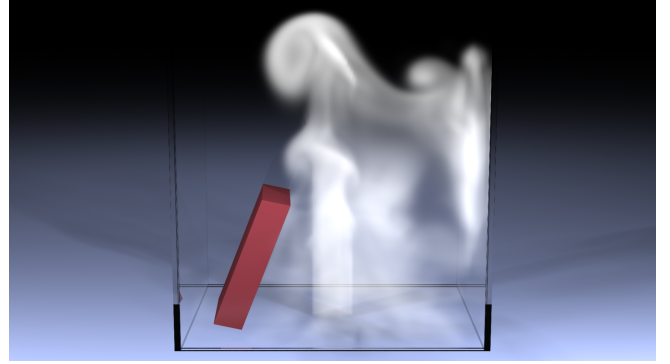


Fig. 4. A light rigid plank interacts with a jet of smoke. (fluid grid size: $100 \times 150 \times 100$)

solver are used to compute difference matrices $\bar{V}$ and $\bar{P}$. Using a QR factorization $\bar{P} = \bar{Q}\bar{R}$, with $\bar{R}_1$ and $\bar{Q}_1$ analogous to R_1 and Q_1 , respectively, and the same procedure as for $\hat{F}_X$ above, we get

$$\hat{S}_p = \bar{V} \bar{R}_1^{-1} \bar{Q}_1^T.$$

The least squares systems described above can become arbitrarily ill-conditioned as more basis vectors are incorporated. To stably compute the near rank-deficient least squares problem, we use the Eigen library's [Guennebaud et al. 2010] QR decomposition with full pivoting.

4.3 Applying the RMI in the partitioned coupling

Equations (6) and (7) are solved in a matrix-free manner using GMRES, and the expressions of the form $R^{-1}Q^T x$ are evaluated through backward substitution rather than formed explicitly. Therefore, although for m interface nodes, $\hat{F}_X$ is $m \times 3m$ and $\hat{S}_p$ is $3m \times m$, the algorithm can exploit their low rank (no more than $k-1$) in the computations to achieve efficiency, and the cost of solving these systems is a small fraction of the overall simulation time (0.2%–4.2% in our examples).

Equation (6) is solved before each call to the fluid solver, while Equation (7) is solved before each call to the solid solver, in order to provide as input the RMI prediction of the converged state to the individual solvers. Effective velocities are computed as $V_{k+1}^f = (X_{k+1}^f - X_0)/\Delta t$ every time before calling the fluid solver. The overall procedure is summarized in Algorithm 1.

4.4 Rigid Bodies

To handle rigid bodies, the RMI samples them with embedded points, whose positions and velocities are determined from the rigid body state. The embedded points are spaced at a distance $1-1.5dx$ apart to be commensurate with the grid. Fluid interaction forces at the embedded points are mapped to a force and torque and passed to the rigid body solver. The process described above for solid nodes is then used without modification with the embedded points. Alternatively, one could formulate the reduced model directly on the rigid body degrees of freedom.

ALGORITHM 1: Single substep of the extended partitioned method, integrating from time t^n to time t^{n+1}

extrapolate solid positions to get X_1^f
 $p_1^s = p_1^f = F(X_1^f, (X_1^f - X^n)/\Delta t)$
 $X_2^f = X_1^s = S(p_1^s)$
 $p_2^s = p_2^f = F(X_2^f, (X_2^f - X^n)/\Delta t)$
 $X_2^s = S(p_2^s)$
 $R = X_2^s - X_1^s$
Construct $\hat{F}_X$ with 1 mode
 $k = 1$
while $\|R\| > \text{tolerance}$ **do**
 Construct $\hat{S}_p$ with k modes
 Solve Eq. (6) to get X_{k+1}^f
 $p_{k+1}^f = F(X_{k+1}^f, (X_{k+1}^f - X^n)/\Delta t)$
 Construct $\hat{F}_X$ with $k+1$ modes
 Solve Eq. (7) to get p_{k+1}^s
 $X_{k+1}^s = S(p_{k+1}^s)$
 $R = X_{k+1}^s - X_k^s$
 $k = k + 1$
end while

4.5 Thin Shells

Similar to [Guendelman et al. 2005], we compute the pressure difference in the direction of the face normal, $\Delta p_f \mathbf{n}_f$, on each fluid grid face f that is marked as a solid boundary. For a solid node i , the pressure difference from face f is weighted by $\mathcal{K}_{if} = \mathcal{K}(\mathbf{x}_i - \mathbf{x}_f)$, a distance-based kernel function. The final pressure-based force on node i is $f_i = \sum_f w_{if} \Delta p_f \mathbf{n}_f \cdot \mathbf{n}_i A_i$ where $w_{if} = \mathcal{K}_{if} / \sum_{f'} \mathcal{K}_{if'}$ is normalized by the sum of weights of all faces influencing node i .

These nodal force components are used in place of the values p_i in Section 4.2. For volumetric solids, we use the same formulation with pressures inside the solid set to zero.

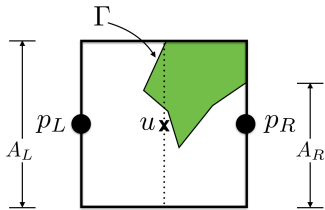


Fig. 5. Schematic of solid in a mixed dual cell of the fluid simulation grid.

5 CONSERVATIVE IMPULSE-BASED COUPLING

The conservative monolithic coupling scheme of [Robinson-Mosher et al. 2008] identified the impulse, I_{DC} , implicitly applied in a mixed dual cell of the fluid simulation MAC grid in order to satisfy the no-slip boundary condition I. We review that here.

Consider a dual cell about a sample of an x-component of the velocity field, u , as shown in Figure 5. Assume that forces other than pressure and interaction with the solid at the interface Γ have been integrated to get an intermediate velocity u^* , as in a typical MAC grid-based Eulerian fluid simulation. Given pressure samples

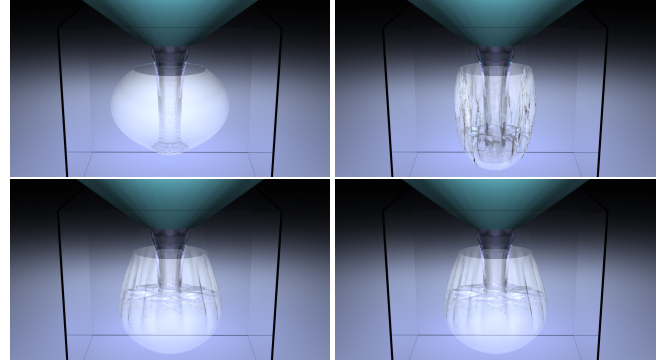


Fig. 6. Center: A thin, transparent elastic balloon is stressed and filled by a fast stream of water. (fluid grid size: $100 \times 125 \times 100$, 4520 solid nodes)

p_L and p_R along the left and right faces of the cell with areas A_L and A_R , respectively, the change in x-momentum due to pressure and interaction with the solid can be written as

$$M_f u^{n+1} - M_f u^* = p_L A_L - p_R A_R + I_{DC}, \quad (8)$$

where $I_{DC} \approx \int_{\Gamma} p n_x dS$, and n_x is the x-component of the outward unit normal to the solid. As in [Robinson-Mosher et al. 2008], rather than explicitly computing $-I_{DC}$ as an approximate boundary integral, and applying that to the solid, we infer its value from Equation (8). Since I_{DC} is the impulse applied on the fluid by the solid, the impulse applied on the solid by the fluid must be $-I_{DC}$ in order to satisfy the dynamic boundary condition II. We propose to apply this impulse in the partitioned approach instead of explicitly approximating $\int_{\Gamma} p n_x dS$.

This has several potential benefits. First, this formulation is conservative, as the fluid/solid interaction is treated through a momentum exchange. Second, in [Robinson-Mosher et al. 2008], this formulation led to an “added mass” term in the solid momentum equation, which contributed to the improved stability properties of that semi-implicit scheme. The added mass effect, whereby the effect of the fluid on the solid can be modeled in part as an increase in solid mass, has been shown to have a destabilizing effect on fully explicit schemes [Causin et al. 2005]. While the added mass term is treated explicitly in the case of our partitioned solve, we also observe improved stability with the impulse-based formulation, which departs from the standard Dirichlet-Neumann scheme. Finally, as in [Robinson-Mosher et al. 2008], the impulse-based formulation applies directly to thin shells as well as volumetric solids, with $M_f = V\rho$ and $A_R = A_L = A$, where V and A are the volume and face area, respectively, of the dual cell. As in [Robinson-Mosher et al. 2008], we take $A_R = A_L = A$ in our implementation.

In order to apply the conservative formulation in a partitioned fashion, the fluid solver must supply as output the impulse $-I_{DC}$ rather than an explicitly computed approximation of the boundary forces. This is a minor modification for fluid solvers that compute pressures through a pressure projection step, such as fractional step fluid solvers [Chorin 1967], or some recent SPH solvers [Ihmsen et al. 2014; Raveendran et al. 2011]. For other solvers, one would have to estimate the constraint force applied to enforce the solid boundary condition. We do not investigate this further in the present work.

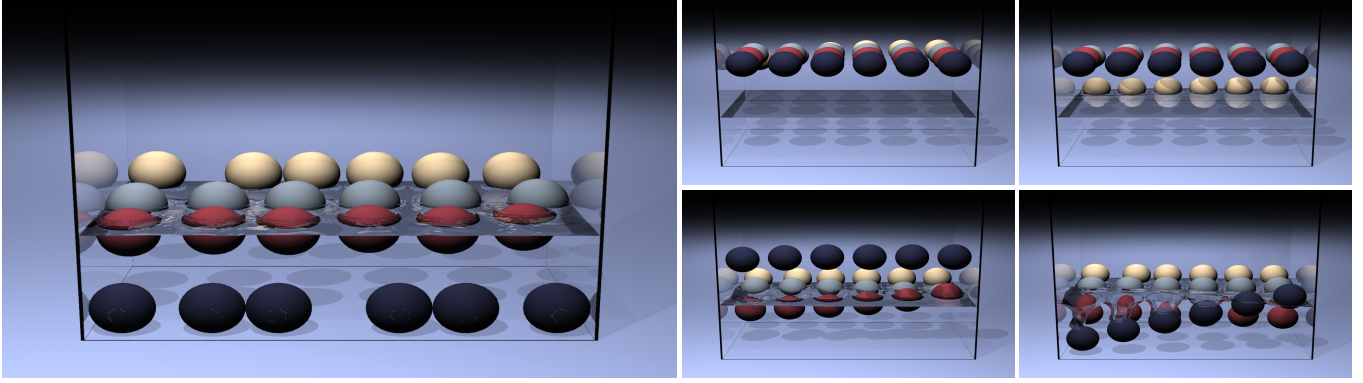


Fig. 7. Rigid balls of varying densities dropped into a pool of water exhibit correct buoyancy. Their densities relative to the water are (back row to front row) 0.1, 0.5, 0.9, and 10. (fluid grid size: $225 \times 300 \times 150$)

When using the conservative formulation as part of the RMI, we replace the pressure values p_i in Section 4.2 with the impulses I_{DC} . Furthermore, rather than interpolating to solid nodes, the impulses are distributed in a conservative fashion to solid nodes as in [Robinson-Mosher et al. 2008]. We demonstrate in our examples that use of the conservative formulation leads to improved stability and lower iteration counts as compared with the non-conservative formulation, for both the partitioned scheme with underrelaxation, and the partitioned scheme with the RMI (see Table 2).

6 BOUNDARY PRESSURE PROJECTION

Several authors have noted a limitation in handling fully enclosed fluid regions in partitioned or interleaved approaches for coupling incompressible flows to structures [Batty et al. 2007; Degroote 2013]. Since the fluid cannot compress or expand, there should be no net flow through the boundary of any fluid region. In a typical partitioned solve, however, the solid solver is not aware of this constraint, and will supply the fluid solver with boundary velocities that violate it in the case of a fully enclosed fluid region. Internal to the fluid solver, this constraint manifests as a pressure Poisson equation with only Neumann boundary conditions, and hence with a nontrivial nullspace, and constraint-violating velocities from the solid lead to an incompatible linear system for pressure. These “Neumann regions” occur in practice, sometimes spontaneously, and must be detected and handled for robustness.

In [Guendelman et al. 2005], a partial solution was implemented by projecting the solid velocities to be compatible. A related issue is that the fluid pressure inside the enclosed region is determined only up to an unknown constant p_0 . Since the fluid is accelerated by the gradient of pressure, p_0 is not needed for the fluid simulation. However, at the enclosed region boundary, the solid should be affected by p_0 , and if it is not properly accounted for, the forces on the solid will be incorrect.

We propose a novel boundary pressure projection (BPP) method, that to our knowledge has not been proposed elsewhere. In particular, we show that p_0 is the Lagrange multiplier associated with the boundary velocity constraint, and show how it can be determined and incorporated into a partitioned solid/fluid coupling approach. Like [Guendelman et al. 2005], our solution involves a projection of the solid velocities to be compatible for the pressure solve. Our

derivation is more general and reduces to the form they used if solids are voxelized to the grid and other boundary source terms are neglected. Additionally, unlike that work, we also solve for the unknown constant component of pressure p_0 and apply this pressure to the solid. This causes the solid to evolve toward constraint-satisfying velocities.

Figure 8 demonstrates the partitioned approach with and without determining and applying p_0 to the solid. The simulation contains a Neumann fluid region, a region fully enclosed by faces with velocities determined either from sources or the dynamic solid. Note that without p_0 , the balloon fails to inflate, whereas with p_0 determined through our BPP, the balloon inflates similarly to our ground truth monolithic simulation.

Let Γ be the boundary of a closed region in the fluid domain, for example, the surface of the balloon and air source as pictured in Figure 8. The divergence free constraint on the fluid implies that

$$\int_{\Gamma} \mathbf{u} \cdot \mathbf{n} dS = 0.$$

This can be discretized on the MAC grid as

$$\sum_f A_f u_f n_f = 0,$$

where A_f is the face area, u_f is the velocity component on that face, and n_f is the closed region normal component on that face (± 1). We write the velocities, area-weighted normal components, and dual cell masses associated with the boundary faces as $\mathbf{U} = (u_1, u_2, \dots, u_n)^T$, $\mathbf{C} = (A_1 n_1, A_2 n_2, \dots, A_n n_n)^T$, and $\mathbf{M} = \text{diag}(M_1, M_2, \dots, M_n)$. The

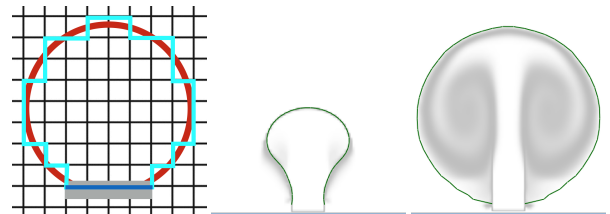


Fig. 8. Inflating a balloon. Left: a schematic of the fluid simulation, with source faces (blue), where air is injected, and solid faces (cyan). Center: a partitioned solve with compatibility projection but without p_0 fails to inflate. Right: the same simulation using our BPP method inflates properly.

constraint equation is

$$C^T U = 0, \quad (9)$$

and the Jacobian of the constraint is given by C^T . We seek an unknown pressure p_0 that when applied to the fluid in the enclosed region causes the fluid to satisfy the constraint (9):

$$M\dot{U} = MU^* + \Delta t C p_0. \quad (10)$$

Note that the pressure p_0 is the unknown Lagrange multiplier for the constraint force $C p_0$. See [Witkin 1997] for a general description of constrained particle dynamics and Lagrange multipliers. Multiplying both sides by M^{-1} , applying the constraint (9), and solving for p_0 , we get

$$p_0 = -\frac{1}{\Delta t} \left(C^T M^{-1} C \right)^{-1} C^T U^*. \quad (11)$$

Plugging this result back into Equation (10) and rearranging gives the projection

$$\hat{U} = \mathbb{P} U^* \quad (12)$$

$$\mathbb{P} = I - M^{-1} C (C^T M^{-1} C)^{-1} C^T \quad (13)$$

If $M_f = \rho V$ in all dual cells and $A_f = A$ is a constant for all faces, then the projection operator in Equation (13) simplifies to $(I - \frac{1}{n} N N^T)$, where n is the number of boundary faces, and N is the vector of normal components, i.e., $C = A N$. This is the projection used in [Guendelman et al. 2005] to ensure compatible velocities for the pressure solve, though they did not calculate p_0 to apply to the solid.

Note that some of the closed region boundary faces may be sources or kinematic objects whose velocities are not determined by the dynamic solid. In that case, those should be isolated from the other faces in (9), and the constraint force and projection should not be applied to those faces in (10) and (13). By partitioning the velocity vector as $U = (U_s^T | U_k^T)^T$, and the constraint equation as $C = (C_s^T | C_k^T)^T$, where the subscripts s, k represent solid and source/kinematic components, respectively, Equations (9)-(13) become

$$\begin{aligned} C_s^T U_s &= -C_k^T U_k \\ M_s \dot{U}_s &= M_s U_s^* + \Delta t C_s p_0 \\ p_0 &= -\frac{1}{\Delta t} \left(C_s^T M_s^{-1} C_s \right)^{-1} (C_s^T U_s^* + C_k^T U_k) \\ \hat{U} &= \mathbb{P} U^* - M_s^{-1} C_s (C_s^T M_s^{-1} C_s)^{-1} C_k^T U_k \\ \mathbb{P} &= I - M_s^{-1} C_s (C_s^T M_s^{-1} C_s)^{-1} C_s^T \end{aligned}$$

The BPP method also applies directly to multiple coupled Neumann regions, as may occur, for example, with cloth folding over itself. In this case, m constraints lead to m rows in C^T , $C^T M^{-1} C$ is an $m \times m$ matrix rather than a scalar, and p_0 is an m -vector containing the unknown constant pressures for all of the enclosed regions. The procedure for finding the vector p_0 is then identical to that described above.

[Küttler et al. 2006] also addressed the problem of partitioned coupling of structures to incompressible fluids. They proposed several possible solutions: to enforce the velocity constraint directly on the solid inside the solid solver, while the fluid solver determines a

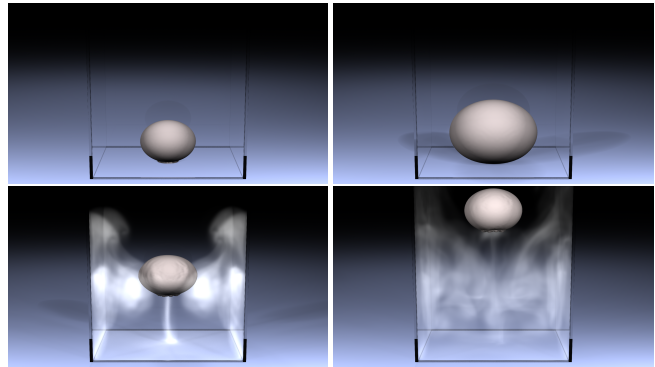


Fig. 9. A thin, elastic balloon is filled with a fast jet of smoke, and then released. The balloon inflates properly due to the use of our BPP method. Computation of this challenging scenario is stabilized by our extended partitioned method. (fluid grid size: $100 \times 150 \times 100$, 4520 solid nodes)

displacement (coupling the fluid and solid in a Neumann-Dirichlet fashion for that one mode), or alternatively to relax the incompressibility constraint. Our solution instead projects the solid velocities based on the state of the fluid, and supplies the fluid with the compatible velocities, and the solid with the pressure p_0 needed to project the velocities, maintaining the Dirichlet-Neumann structure of the partitioning. Hence, in our approach, as for the other fluid pressures, the determination of p_0 is not implicitly coupled to other solid forces, and does not require a constraint to be imposed on the solid. When strong coupling is achieved, the solid velocities on the enclosed fluid region boundary converge to satisfy the incompressibility constraint.

To incorporate the BPP, each call $F(X, (X - X^n)/\Delta t)$ in Algorithm 1 is replaced by a call to `Solve_Fluid(X)`, shown in Algorithm 2. We demonstrate the BPP method by simulating a balloon being filled with smoke (Figure 9). The incompressible fluid inside the balloon is fully enclosed by the balloon surface and smoke source, and using the BPP it inflates properly. We further demonstrate the BPP method by simulating a hydraulic press, where a fully enclosed incompressible medium is used to transmit pressures between two pistons (Figure 10). In both cases, determining p_0 and applying it to the solid is essential to capturing the dynamics of the system.

ALGORITHM 2: Wrapper for the fluid solver which applies the boundary pressure projection

```

p = function Solve_Fluid(X)
    Compute effective velocities  $\tilde{V} = (X - X^n)/\Delta t$ 
     $(V, p_0) = \text{BPP}(\tilde{V})$ 
     $p = F(X, V) + p_0$ 
end function

```

7 RESULTS

Here, we demonstrate individually the performance of the RMI, the conservative impulses, the BPP, and the combination of these techniques, the XPM. In general, monolithic approaches are expected to be more efficient in computing a strongly coupled solution. The benefit of the partitioned approach lies not in fast computation time,

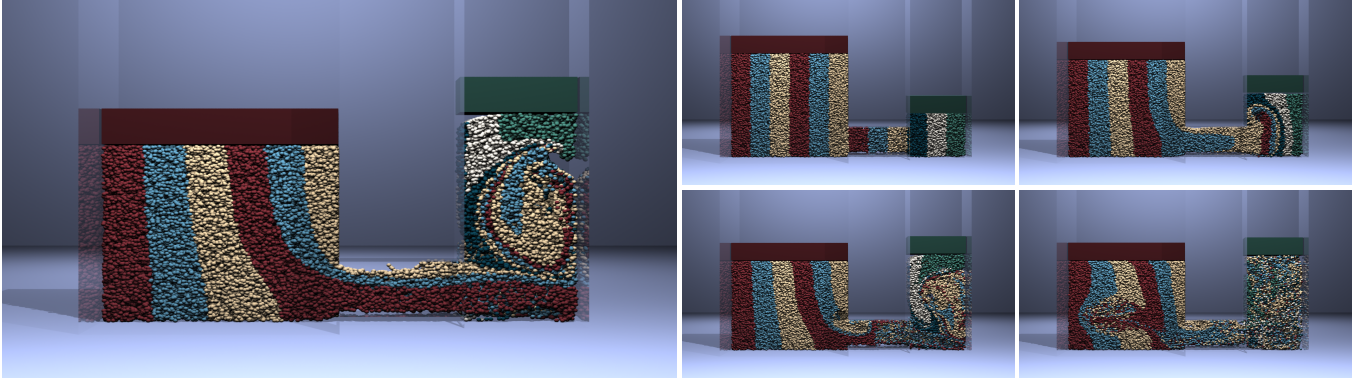


Fig. 10. Pressure applied to a small piston of a hydraulic press is transmitted undiminished to a large piston through a fully enclosed, incompressible medium, using our BPP method to determine the constant component of fluid pressure p_0 . (fluid grid size: $160 \times 160 \times 80$)

but in the ability to reuse existing software with little modification. Therefore, we do not expect to outperform the monolithic solver, but aim instead to improve performance of the partitioned approach.

The cost of the RMI layer is relatively small compared with the cost of the fluid and solid solvers. Moreover, since we are using existing black-box fluid and solid solvers, our primary measure of performance gains due to our approach is in the reduction of iteration counts.

We computed our examples with MPI parallelism on a cluster of 16 nodes, each comprised of 2 AMD Opteron 6272 CPUs totaling 32 cores, 64GB ECC memory, and 40Gbps InfiniBand networks. In all of our simulations, we used the smaller of the time step sizes computed by the individual fluid and solid solvers, and did not impose additional stability conditions on the time step due to the coupling. We limited the number of reduced model subiterations to 30, although the typical number of subiterations was much less. We used a convergence tolerance of 5% of the fluid grid cell size.

7.1 Improved stability and convergence with RMI, conservative impulses, and XPM

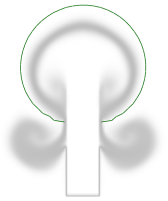


Fig. 11. 2D cloth interacting with smoke.

In this section, we study the benefits of using the RMI, the conservative impulses, and both (the XPM), on a two-dimensional example of cloth interacting with a jet of smoke with source velocity $U = 0.5$, shown in Figure 11. We used the PhysBAM cloth simulator and PhysBAM smoke simulator. Though PhysBAM supports monolithic coupling [Robinson-Mosher et al. 2008], we treated the solvers as black boxes and coupled them in a partitioned fashion.

We compare the performance of our method with that of a partitioned solver with underrelaxation, shown in Algorithm 3, for different values of the underrelaxation parameter, ω . For smaller values of ω , we expect better stability at the expense of a greater number of iterations. Table 1 shows the average number of iterations for the solves, with a dash indicating the simulation became unstable before completion. M is the mass ratio of the balloon to the air, and problem difficulty increases with decreasing M . We found that for $\omega \geq 0.5$, the simple partitioned approach

Table 1. Average number of iterations per substep in the simulation of Figure 11, with fewest iterations for each M in blue bold. The solvers compared were the underrelaxed partitioned (P) with various underrelaxation parameters, ω , underrelaxed partitioned with conservative impulses (P-C), reduced model interface (RMI), and reduced model interface with conservative impulses (XPM). Our XPM required the fewest iterations and was able to handle the most challenging mass ratios where the underrelaxed partitioned approach became unstable. Use of conservative impulses instead of pressure-based forces improved stability overall for both the underrelaxed partitioned method (P vs. P-C) and for the RMI (RMI vs. XPM).

M	P $\omega \geq .5$	P $\omega = .2$	P $\omega = .1$	P $\omega = .05$	P $\omega = .01$	P-C $\omega = .05$	RMI $\omega = 1$	XPM $\omega = 1$
.1	-	-	-	-	28.2	5.5	5.5	3.8
.2	-	-	-	5.7	27.3	4.5	4.0	3.2
.5	-	-	2.6	4.8	23.8	4.6	3.0	2.6
.8	-	3.4	2.6	5.3	10.2	5.0	2.7	2.1
1	-	2.2	2.8	5.6	9.1	4.7	2.6	2.0
2	-	2.0	2.9	5.1	5.2	3.2	2.0	2.0

failed to converge from the outset for all values of M . As M was decreased, the partitioned approach required increasingly stringent underrelaxation for convergence, which led to a greater number of subiterations.

ALGORITHM 3: Partitioned solver with underrelaxation

```

k = 1
while ||R|| > tolerance do
   $X^{k+1} = S(p^k)$ 
   $X^{k+1} = \omega X^{k+1} + (1 - \omega)X^k$ 
   $p^{k+1} = F(X^{k+1}, (X^{k+1} - X^n)/\Delta t)$ 
   $R = X^{k+1} - X^k$ 
  k = k + 1
end while

```

When partitioning with underrelaxation is used, performance can be improved by using Aitken relaxation, which dynamically determines the underrelaxation parameter [Küttler and Wall 2008]. Though we do not compare with that approach here, the RMI has been demonstrated to outperform Aitken relaxation as well [Vierendeels et al. 2007].

7.1.1 Effect of RMI. Table 1 compares the partitioned approach (P columns) with the RMI (RMI column). Specifically, we compare the RMI with the partitioned method with the largest value of ω that converged. The RMI outperformed the underrelaxed partitioned approach in 3 out of 6 cases, tied it in 1 case, and performed slightly worse in 2 cases. For $M = .1$, the RMI converged, whereas the underrelaxed partitioned method with $\omega \geq 0.05$ did not. Therefore, the RMI appears to improve the stability of the partitioned approach over underrelaxation alone.

7.1.2 Effect of conservative impulses. The conservative impulse-based formulation is orthogonal to the RMI and can be used with the underrelaxed partitioned scheme. To illustrate the effect of using conservative impulses instead of the explicit pressure-based forces, Table 1 compares the partitioned approach with pressured based forces (column P, $\omega = .05$) and with the conservative impulses (column P-C, $\omega = .05$). In every case, using the conservative impulses reduced the number of iterations required. Furthermore, for $M = .1$, $\omega \geq .05$, the partitioned approach with pressure-based forces was unstable, whereas the partitioned approach with impulse-based forces was stable. For this example, RMI and XPM also differ only in the use of the conservative impulses. Comparison of Table 1 columns RMI and XPM shows that the use of conservative impulses reduced average iteration counts by up to 30%.

Tables 1 and 2 show only average performance for a problem that is initially easy. The significant benefits of the XPM are better illustrated in Figure 12, which shows the number of iterations of the underrelaxed partitioned and XPM solutions over 540 substeps of two-dimensional simulations of a balloon filling with smoke, with $M = 0.8$, $\omega = 0.2$. Initially, both methods perform comparably, but as the balloon stretches and the simulation becomes more challenging, the performance of the underrelaxed partitioned approach drops, while the XPM performance is largely unchanged, offering 3-4x better performance in the most challenging part of the simulation.

Table 2. Percentage improvement when using conservative impulse terms, with $-$ indicating both simulations were unstable, and ∞ indicating the unstable simulation was stabilized by using the conservative scheme. For the underrelaxed partitioned scheme, using the conservative impulses generally improved iteration counts and allowed for more challenging mass ratios, though smaller ratios were still unstable. Using the XPM, all simulations were stable and iteration counts were improved up to 30% over the RMI.

M	P vs. P-C				RMI vs. XPM (RMI-C)
	$\omega = .05$	$\omega = .1$	$\omega = .2$	$\omega = .5, .8$	
.1	∞	-	-	-	29.9
.2	21.3	∞	-	-	21.4
.5	4.0	-13.3	∞	-	12.4
.8	6.2	8.2	15.4	-	22.0
1	16.3	25.1	-18.1	-	21.6
2	38.4	30.0	0.0	∞	0.3

Table 2 gives further details on the performance improvement using the conservative impulses, comparing the partitioned scheme with and without conservative impulses for a variety of underrelaxation parameters, as well as the RMI and XPM schemes.

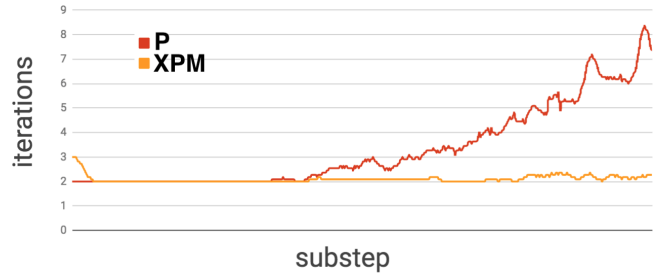
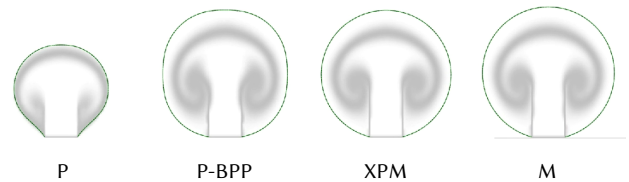


Fig. 12. Number of iterations of the underrelaxed partitioned (P) and reduced model interface with conservative impulses (XPM) over 540 substeps of two-dimensional simulations of a balloon filling with smoke. $M = 0.8$, $\omega = 0.2$. The values are averaged over a window of 10 substeps to reduce noise.

7.2 Boundary pressure projection method

The boundary pressure projection method is orthogonal to the RMI or use of conservative impulses and can be used with any partitioned scheme. Figure 8 shows a two-dimensional simulation of a deformable balloon being inflated with air. Inside the balloon, the air is surrounded by either solid faces, or by the source boundary conditions, and hence constitutes a “Neumann region”. Without the BPP, the balloon fails to properly inflate.



Further comparison is shown in the figure above, depicting results of the partitioned scheme without BPP (P), partitioned scheme with BPP (P-BPP), XPM, and monolithic scheme (M). Using the BPP, both the P-BPP and XPM approaches successfully inflate the balloon. XPM is better able than P-BPP to replicate the balloon shape produced with the “ground truth” monolithic method.

Figure 9 shows a three-dimensional simulation of a balloon filling with smoke, using XPM. As in the 2D case, the balloon does not properly inflate without the BPP. Figure 10 shows a three-dimensional simulation of a hydraulic press, where pressure applied to the small piston is transmitted undiminished to the large piston through a fully enclosed and incompressible medium. Without the BPP, both rigid plates fall downward and the simulation eventually fails. Passive tracer particles are included as a post-process to better visualize the flow. We note that they are advected with interpolated velocities and forward Euler time integration, leading to visual density artifacts that do not reflect density errors in the flow solver.

7.3 Variety of solids coupled to an Eulerian smoke and water

We demonstrate our method in three dimensions on a variety of challenging scenarios. We use PhysBAM for both the solid and fluid solvers. We again use a tolerance of $0.05dx$ for the XPM. Figure 2 demonstrates the coupling of a deformable object and smoke through the RMI. Figure 4 demonstrates our extension of the RMI to handle rigid bodies by simulating a rigid plank coupled to smoke. In Figure 6, a deformable balloon is filled with water using our XPM.

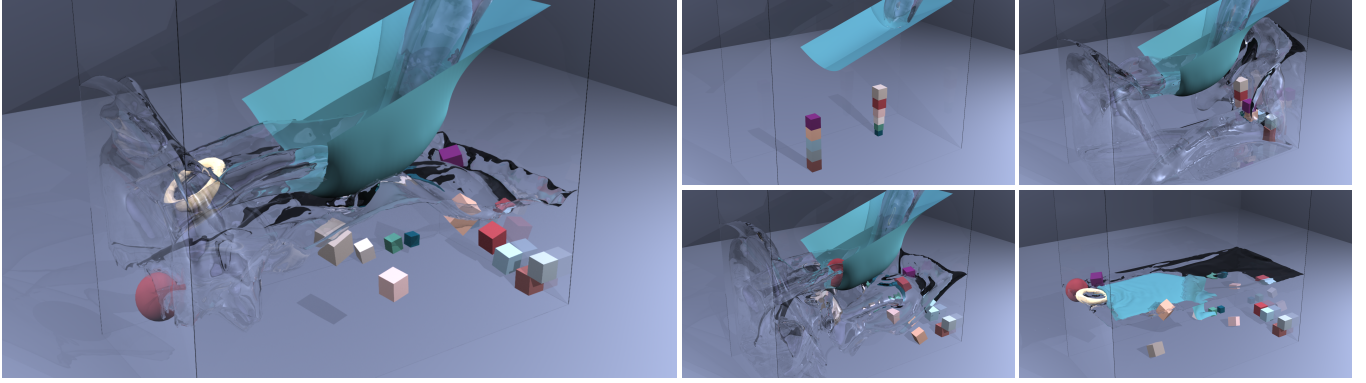


Fig. 14. Complex example: Simulation of cloth and rigid bodies of varying densities coupled to Eulerian water. (fluid grid size: $120 \times 120 \times 60$)

In Figure 7, the XPM is used to couple many rigid balls to water, capturing the buoyancy forces. Figure 14 depicts a complex example with interacting water, cloth, and rigid bodies of varying densities.

Table 3. The number of frames, average simulation time per substep, and average number of substeps per frame are shown for several examples.

Example	# frames	Avg. time per substep (s)	Avg. number substeps per frame
smoke balloon (Fig. 9)	117	30.3	39.2
water balloon (Fig. 6)	143	96.3	30.0
water piston (Fig. 10)	143	140.6	3.2
smoke rigid (Fig. 4)	80	96.6	10.5
smoke deformable (Fig. 2)	500	14.6	12.1
water rigid (Fig. 7)	495	30.7	11.8
water complex (Fig. 14)	447	48.9	16.0

Table 4. Percentage of time spent in the fluid solver, solid solver, and XPM interface, and average number of modes used by the RMI (average number of iterations minus one). The XPM interface computations require a relatively small amount of time, taking 0.2–4.2% of the total simulation time in our examples.

Example	Fluid solver	Solid solver	XPM interface	Avg # modes
smoke balloon (Fig. 9)	95.5	0.3	4.2	2.1
water balloon (Fig. 6)	98.5	0.4	1.1	5.9
water piston (Fig. 10)	98.2	0.4	1.4	3.8
smoke rigid (Fig. 4)	99.7	0.1	0.2	2.5
smoke deformable (Fig. 2)	92.6	6.3	1.0	3.2
water rigid (Fig. 7)	99.1	0.4	0.5	1.0
water complex (Fig. 14)	94.5	3.8	1.7	5.1

Table 3 summarizes the simulations times for each example. Overall, our 3D simulations took roughly between a few minutes per frame to up to a couple of hours per frame on up to 64 cores. A small amount of time is spent in the XPM interface layer as compared with the individual solid and fluid solvers (see Table 4). We also note that in our experimentation, without conservative impulses, some of these simulations did reach the maximum 30 iterations, whereas with the conservative impulses, the average number of iterations was 2–7.

7.4 Rigid bodies coupled to SPH fluids

We have implemented the reduced model interface for coupling an SPH fluid solver and a rigid body solver. For the SPH solver, we used the implementation of Divergence-Free SPH [Bender and Koschier 2017] in the open source SPLisHSPlasH library [Bender 2017], and for the rigid body solver, we use the external position-based dynamics [Deul et al. 2016] rigid body solver included with the SPLisHSPlasH library. Internal to the SPLisHSPlasH library, solid-fluid coupling is implemented using the method of Akinci et al. [Akinci et al. 2012], which is a conservative, weak coupling scheme. In this method, particles are sampled inside the solid surface and used as ghost SPH particles in updating the dynamics of the fluid. Pressure forces are applied conservatively to both components in opposite directions to preserve momentum. We compare the performance of an underrelaxed partitioned strong coupling scheme, the XPM, and the native weak coupling of SPLisHSPlasH. For the XPM method, we use the positions and pressure forces of the solid sample particles to build the reduced system. In order to apply the XPM, we had to add rollback functionality to both the rigid body and SPH solvers. The method did not require any other changes.

7.4.1 XPM improves stability over weak coupling. Figure 15 (top row) shows snapshots from a simulation of a rigid ball interacting with fluid, where the density ratio of the ball to the fluid is 0.1. The XPM results are iterated until a convergence tolerance of 10% times the particle radii is achieved. In this case, both the weak coupling native to SPLisHSPlasH and the XPM solution are stable. Next, we decreased the density ratio to 0.05, making the problem more challenging. In this case, the weakly coupled SPLisHSPlasH became unstable. The XPM was able to compute the simulation stably. Figure 15 (bottom row) shows snapshots from the simulation with density ratio 0.05. In these examples, 5684 fluid particles were used. Figure 16 shows an XPM simulation of an SPH fluid coupled to a rigid sphere ($M=0.05$), cylinder ($M=2$), and torus ($M=0.7$), with 24389 SPH fluid particles.

7.4.2 XPM improves performance over underrelaxed strong coupling. We also tested the underrelaxed partitioned strong coupling approach in Algorithm 3 on the SPH/rigid coupling. We chose $\omega = 0.05$ for the underrelaxation parameter, since larger values of ω had failed for density ratios less than 0.1 (see Table 1). As

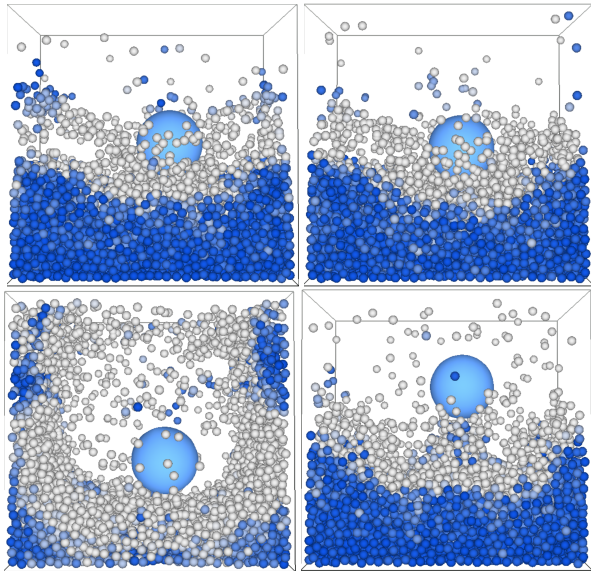


Fig. 15. Top row: Simulations of a rigid ball coupled to an SPH fluid, with ball/fluid density ratio of 0.1. All images are at frame 26. Left: Weakly coupled solution computed with SPLisHSPlasH is stable. Right: Solution computed with XPM, with a tolerance of 10%, is also stable. Bottom row: The density ratio has been decreased to a more challenging value of 0.05. Left: Weakly coupled solution computed with SPLisHSPlasH is now unstable. Right: Solution computed with XPM, with tolerance of 10%, is still stable.

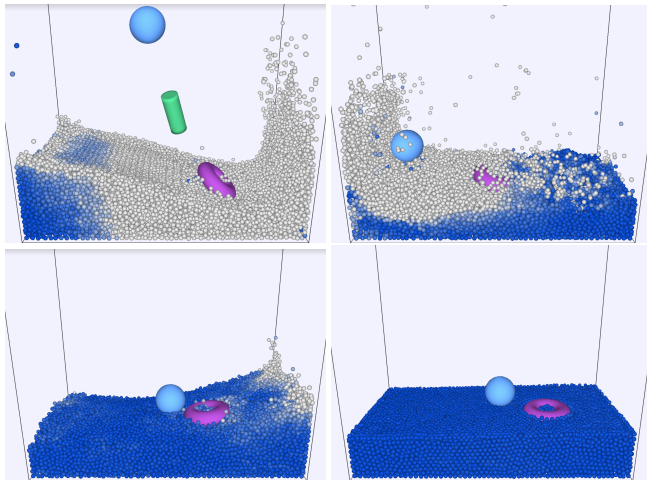


Fig. 16. XPM simulation of an SPH fluid coupled to a rigid sphere ($M=0.05$), cylinder ($M=2$), and torus ($M=0.7$), with 24389 SPH fluid particles. Top to bottom, left to right: frame 17, 61, 105, and 360.

shown in Figure 17, strong coupling was able to simulate the example with density ratio of 0.05, but took more iterations than the XPM on average to do so. In the most challenging parts of the simulation, the underrelaxed partitioned coupling approach required 2-3x more iterations than the XPM. Therefore, the XPM improves performance when compared with the underrelaxed, partitioned strong coupling.

7.4.3 Strong versus weak XPM. So far in the paper, we have not explored the tolerance parameter used by the XPM. Here, we illustrate that the XPM can be used with a larger tolerance to compute a

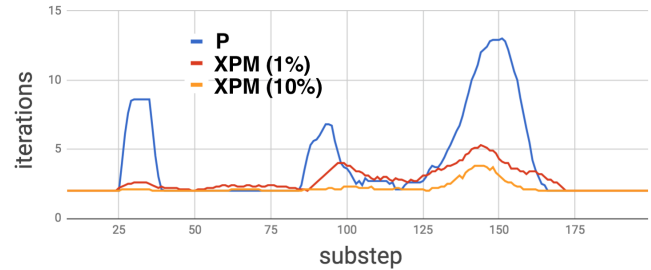


Fig. 17. Number of iterations per substep for simulation of an SPH fluid coupled to a rigid ball with density ratio 0.05, using the underrelaxed partitioned scheme (P) and the XPM with tolerance 1% and 10%. The XPM requires fewer iterations on average than the underrelaxed partitioned scheme. Increasing the tolerance further reduces iteration counts while stability is maintained. The values are averaged over a window of 10 substeps to reduce noise.

weak coupling, which will be faster but less accurate. We computed the rigid ball/fluid simulation with the XPM with both a tolerance of 1% for strong coupling and a tolerance of 10% for weak coupling. Both computed stably, and the 10% tolerance required fewer iterations as expected. The larger tolerance may be a good choice when speed is preferred over accuracy and stability is not compromised. Figure 17 compares the iteration counts for the simulations of the underrelaxed, strongly coupled solution, XPM with tolerance 1%, and XPM with tolerance 10%.

8 CONCLUSION

We have presented a novel extended partitioned method for two-way coupling of incompressible fluids and solids. We have demonstrated that the XPM allows one to solve with a partitioned solver, reusing existing black box solid/fluid solvers, scenarios that were previously solved with specialized monolithic solvers. The XPM could be used to couple other fluid and solid solvers in a black-box fashion, and the reduced model interface could further apply to coupling various materials efficiently. We also note that the reduced model is dynamically updated throughout the simulation and applied locally in a small time window, rather than using a fixed, precomputed basis over a long simulation time. The RMI method is closely related to quasi-Newton methods for partitioned solid-fluid coupling [Degroote 2013; Degroote et al. 2009, 2010]. Recently, it was shown that saving solver state over multiple time steps can lead to improved estimates of the solver Jacobians [Haelterman et al. 2016]. It may also be interesting to consider the use of other types of reduced models in constructing the RMI.

When using the XPM, we ran most of our examples to a convergence tolerance of 5% of the fluid grid cell size. Since convergence is not necessarily required for graphics applications, it may be practically better to run the XPM with a small, fixed number of iterations, or enforce a larger convergence tolerance, as demonstrated in the SPH example. This maintains stability benefits of XPM, but the reduces the cost at the expense of accuracy. While the XPM significantly improves stability over the partitioned approach with underrelaxation, it is not unconditionally stable.

In future work, we would like to further investigate the BPP on multiple coupled regions, as outlined above. We are also interested in applied our method to scenarios with surface tension.

ACKNOWLEDGMENTS

The authors are grateful for the feedback of the anonymous reviewers. The authors thank Avi Robinson-Mosher and Adam W. Bargteil for helpful discussions and feedback and Mark Lieu for help in rendering. This work was supported in part by National Science Foundation award IIS-1314813.

REFERENCES

- Nadir Akinci, Jens Cornelis, Gizem Akinci, and Matthias Teschner. 2013. Coupling elastic solids with smoothed particle hydrodynamics fluids. *Computer Animation and Virtual Worlds* 24, 3-4 (2013), 195–203.
- Nadir Akinci, Markus Ihmsen, Gizem Akinci, Barbara Solenthaler, and Matthias Teschner. 2012. Versatile Rigid-fluid Coupling for Incompressible SPH. *ACM Trans. Graph.* 31, 4, Article 62 (July 2012), 8 pages. <https://doi.org/10.1145/2185520.2185558>
- Jeffrey W Banks, William D Henshaw, and Donald W Schwendeman. 2014. An analysis of a new stable partitioned algorithm for FSI problems. Part I: Incompressible flow and elastic solids. *J. Comput. Phys.* 269 (2014), 108–137.
- Jernej Barbič, Marco da Silva, and Jovan Popović. 2009. Deformable object animation using reduced optimal control. *ACM Transactions on Graphics (TOG)* 28, 3 (2009), 53.
- Christopher Batty, Florence Bertails, and Robert Bridson. 2007. A Fast Variational Framework for Accurate Solid-fluid Coupling. *ACM Trans. Graph.* 26, 3, Article 100 (July 2007). <https://doi.org/10.1145/1276377.1276502>
- M. Becker, H. Tessenendorf, and M. Teschner. 2009. Direct Forcing for Lagrangian Rigid-Fluid Coupling. *IEEE Transactions on Visualization and Computer Graphics* 15, 3 (May 2009), 493–503. <https://doi.org/10.1109/TVCG.2008.107>
- Jan Bender. 2017. SPHSHPlasH Library. <https://github.com/InteractiveComputerGraphics>
- Jan Bender and Dan Koschier. 2017. Divergence-Free SPH for Incompressible and Viscous Fluids. *IEEE Transactions on Visualization and Computer Graphics* 23, 3 (March 2017), 1193–1206. <https://doi.org/10.1109/TVCG.2016.2578335>
- Mark Carlson, Peter J. Mucha, and Greg Turk. 2004. Rigid Fluid: Animating the Interplay Between Rigid Bodies and Fluid. *ACM Trans. Graph.* 23, 3 (Aug. 2004), 377–384. <https://doi.org/10.1145/1015706.1015733>
- Paola Causin, Jean-Frédéric Gerbeau, and Fabio Nobile. 2005. Added-mass effect in the design of partitioned algorithms for fluid–structure problems. *Computer methods in applied mechanics and engineering* 194, 42 (2005), 4506–4527.
- Nuttapong Chentanez, Tolga G. Goktekin, Bryan E. Feldman, and James F. O'Brien. 2006. Simultaneous Coupling of Fluids and Deformable Bodies. In *Proceedings of the 2006 ACM SIGGRAPH/Eurographics Symposium on Computer Animation (SCA '06)*. Eurographics Association, Aire-la-Ville, Switzerland, Switzerland, 83–89. <http://dl.acm.org/citation.cfm?id=1218064.1218075>
- Alexandre Joel Chorin. 1967. A numerical method for solving incompressible viscous flow problems. *Journal of computational physics* 2, 1 (1967), 12–26.
- Joris Degroote. 2013. Partitioned Simulation of Fluid-Structure Interaction. *Archives of Computational Methods in Engineering* 20, 3 (2013), 185–238. <https://doi.org/10.1007/s11831-013-9085-5>
- Joris Degroote, Klaus-Jürgen Bathe, and Jan Vierendeels. 2009. Performance of a new partitioned procedure versus a monolithic procedure in fluid–structure interaction. *Computers & Structures* 87, 11 (2009), 793–801.
- Joris Degroote, Peter Bruggeman, Robby Haelterman, and Jan Vierendeels. 2008. Stability of a coupling technique for partitioned solvers in FSI applications. *Computers & Structures* 86, 23 (2008), 2224–2234.
- Joris Degroote, Robby Haelterman, Sebastiaan Annerel, Peter Bruggeman, and Jan Vierendeels. 2010. Performance of partitioned procedures in fluid–structure interaction. *Computers & structures* 88, 7 (2010), 446–457.
- Crispin Deul, Patrick Charrier, and Jan Bender. 2016. Position-based Rigid-body Dynamics. *Comput. Animat. Virtual Worlds* 27, 2 (March 2016), 103–112. <https://doi.org/10.1002/cav.1614>
- Jean-Frédéric Gerbeau and Marina Vidrascu. 2003. A quasi-Newton algorithm based on a reduced model for fluid-structure interaction problems in blood flows. *ESAIM: Mathematical Modelling and Numerical Analysis* 37, 4 (2003), 631–647.
- Eran Guendelman, Andrew Selle, Frank Losasso, and Ronald Fedkiw. 2005. Coupling Water and Smoke to Thin Deformable and Rigid Shells. *ACM Trans. Graph.* 24, 3 (July 2005), 973–981. <https://doi.org/10.1145/1073204.1073299>
- Gaël Guennebaud, Benoît Jacob, et al. 2010. Eigen v3. <http://eigen.tuxfamily.org>.
- R Haelterman, Alfred EJ Bogaers, K Scheufile, B Uekermann, and M Mehl. 2016. Improving the performance of the partitioned QN-ILS procedure for fluid–structure interaction problems: Filtering. *Computers & Structures* 171 (2016), 9–17.
- Xiaowei He, Ning Liu, Guoping Wang, Fengjun Zhang, Sheng Li, Songdong Shao, and Hongan Wang. 2012. Staggered meshless solid-fluid coupling. *ACM Transactions on Graphics (TOG)* 31, 6 (2012), 149.
- Gene Hou, Jin Wang, and Anita Layton. 2012. Numerical Methods for Fluid-Structure Interaction – A Review. *Communications in Computational Physics* 12, 2 (2012), 337–377. <https://doi.org/10.4208/cicp.291210.290411s>
- Markus Ihmsen, Jens Cornelis, Barbara Solenthaler, Christopher Horvath, and Matthias Teschner. 2014. Implicit incompressible SPH. *IEEE Transactions on Visualization and Computer Graphics* 20, 3 (2014), 426–435.
- Richard Keiser, Bart Adams, Dominique Gasser, Paolo Bazzi, Philip Dutré, and Markus Gross. 2005. A unified lagrangian approach to solid-fluid animation. In *Point-Based Graphics, 2005. Eurographics/IEEE VGTC Symposium Proceedings*. IEEE, 125–148.
- Bryan M. Klingner, Bryan E. Feldman, Nuttapong Chentanez, and James F. O'Brien. 2006. Fluid Animation with Dynamic Meshes. *ACM Trans. Graph.* 25, 3 (July 2006), 820–825. <https://doi.org/10.1145/1141911.1141961>
- Ulrich Küttler, Christiane Förster, and Wolfgang A Wall. 2006. A solution for the incompressibility dilemma in partitioned fluid–structure interaction with pure Dirichlet fluid domains. *Computational Mechanics* 38, 4 (2006), 417–429.
- Ulrich Küttler and Wolfgang A. Wall. 2008. Fixed-point fluid–structure interaction solvers with dynamic relaxation. *Computational Mechanics* 43, 1 (2008), 61–72. <https://doi.org/10.1007/s00466-008-0255-5>
- Patrick Le Tallec and Jean Mouro. 2001. Fluid structure interaction with large structural displacements. *Computer Methods in Applied Mechanics and Engineering* 190, 24 (2001), 3039–3067.
- Wenlong Lu, Ning Jin, and Ronald Fedkiw. 2016. Two-way coupling of fluids to reduced deformable bodies. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. Eurographics Association, 67–76.
- Matthias Müller, Simon Schirm, Matthias Teschner, Bruno Heidelberger, and Markus Gross. 2004. Interaction of Fluids with Deformable Solids: Research Articles. *Comput. Animat. Virtual Worlds* 15, 3-4 (July 2004), 159–171. <https://doi.org/10.1002/cav.v15.3/4>
- Rahul Narain, Abhinav Golas, and Ming C Lin. 2010. Free-flowing granular materials with two-way solid coupling. *ACM Transactions on Graphics (TOG)* 29, 6 (2010), 173.
- Seungtaik Oh, Younghee Kim, and Byung-Seok Roh. 2009. Impulse-based Rigid Body Interaction in SPH. *Comput. Animat. Virtual Worlds* 20, 2-3 (June 2009), 215–224. <https://doi.org/10.1002/cav.v20.2/3>
- Karthik Raveendran, Chris Wojtan, and Greg Turk. 2011. Hybrid smoothed particle hydrodynamics. In *Proceedings of the 2011 ACM SIGGRAPH/Eurographics symposium on computer animation*. ACM, 33–42.
- Avi Robinson-Mosher, Craig Schroeder, and Ronald Fedkiw. 2011. A symmetric positive definite formulation for monolithic fluid structure interaction. *J. Comput. Phys.* 230, 4 (2011), 1547–1566.
- Avi Robinson-Mosher, Tamar Shinar, Jon Gretarsson, Jonathan Su, and Ronald Fedkiw. 2008. Two-way Coupling of Fluids to Rigid and Deformable Solids and Shells. *ACM Trans. Graph.* 27, 3, Article 46 (Aug. 2008), 9 pages. <https://doi.org/10.1145/1360612.1360645>
- X Shao, Z Zhou, Nadia Magnenat-Thalmann, and W Wu. 2015. Stable and Fast Fluid–Solid Coupling for Incompressible SPH. In *Computer Graphics Forum*, Vol. 34. Wiley Online Library, 191–204.
- Barbara Solenthaler, Jürg Schläfli, and Renato Pajarola. 2007. A unified particle model for fluid–solid interactions. *Computer Animation and Virtual Worlds* 18, 1 (2007), 69–82.
- Alexey Stomakhin, Craig Schroeder, Chenfanfu Jiang, Lawrence Chai, Joseph Teran, and Andrew Selle. 2014. Augmented MPM for Phase-change and Varied Materials. *ACM Trans. Graph.* 33, 4, Article 138 (July 2014), 11 pages. <https://doi.org/10.1145/2601097.2601176>
- Yun Teng, David I. W. Levin, and Theodore Kim. 2016. Eulerian Solid-fluid Coupling. *ACM Trans. Graph.* 35, 6, Article 200 (Nov. 2016), 8 pages. <https://doi.org/10.1145/2980179.2980229>
- Adrien Treuille, Andrew Lewis, and Zoran Popović. 2006. Model Reduction for Real-time Fluids. *ACM Trans. Graph.* 25, 3 (July 2006), 826–834. <https://doi.org/10.1145/1141911.1141962>
- Jan Vierendeels. 2006. *Implicit Coupling of Partitioned Fluid-Structure Interaction Solvers using Reduced-Order Models*. Springer Berlin Heidelberg, Berlin, Heidelberg, 1–18. https://doi.org/10.1007/3-540-34596-5_1
- Jan Vierendeels, Joris Degroote, Sebastiaan Annerel, and Robby Haelterman. 2011. Stability issues in partitioned FSI calculations. In *Fluid Structure Interaction II*. Springer, 83–102.
- Jan Vierendeels, Lieve Lanoye, Joris Degroote, and Pascal Verdonck. 2007. Implicit coupling of partitioned fluid–structure interaction problems with reduced order models. *Computers & structures* 85, 11 (2007), 970–976.
- Andrew Witkin. 1997. Physically based modeling: principles and practice constrained dynamics. *Computer graphics* (1997).
- Omar Zarifi and Christopher Batty. 2017. A positive-definite cut-cell method for strong two-way coupling between fluids and deformable bodies. In *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*. ACM, 7.