arXiv:2301.01976v1 [cs.GR] 5 Jan 2023

一种用于拉格朗日固液耦合的接触代理分裂方法

TIANYI XIE、MINCHEN LI、YIN YANG 和 CHENFANFU JIANG



图1. **踢水**。 人体模型穿着裙子在游泳池中踢水、激起水花时，我们的方法能够精准捕捉水、多层裙子与人体模型之间复杂的相互作用，且不会发生相互穿透。

我们提出了一种基于全新算子分裂策略的稳健高效方法，用于模拟拉格朗日固液耦合。我们采用变分形式近似流体性质和固液相互作用，并利用基于内点障碍函数的摩擦接触，为SPH流体与FEM固体引入统一的双向耦合形式。借助一种结合增广接触代理的新型时间分裂方法，我们将所得优化问题拆分为流体阶段和固体耦合阶段，并提出了高效的定制线性求解器。该技术能够处理流体与不同几何形状和余维数的非线性超弹性物体之间的相互作用，同时在算法上保证满足非穿透判据。全面的基准测试和实验验证了该方法的有效性。

附加关键词和短语：拉格朗日固体-流体耦合、时间分裂、接触代理

1 引言

固体与流体的耦合在自然界中很常见，但模拟起来颇具挑战。固体通常使用拉格朗日网格进行模拟，而流体则常用欧拉网格离散化，以适应拓扑变化。为了准确耦合这些不同的离散化方法，通常需要采用复杂算法，例如 Zarifi 和 Batty [2017]提出的方法。遗憾的是，这类方法计算开销大，且无法处理薄壳。纯欧拉方法[Teng 等，2016；Valkov 等人 2015] 或 SPH 方法 [Akinci 等，2012；Gissler 等人 2019] 通过采用相同视角的离散化，已成功实现双向耦合。但它们难以扩展到非线性弹性动力学。诸如物质点法（MPM） [Jiang 等人2016] 这样的混合方法能够模拟混合材料，但可能产生人为黏滞性，除非采用计算开销更大的方案 [Fang 等人2020]加以解决。此外，这些方法无法确保轨迹不相交，并且在平流过程中，流体意外穿透固体时，往往需要额外的校正程序进行处理。

我们采用拉格朗日路径，并提出一种耦合 FEM固体与 SPH流体的新方法。通过近似固体、流体，以及相互作用项的势能，我们将双向耦合表述为一个优化问题。具体而言，我们从基于位置的流体方法 [Macklin 和 Müller 2013]中汲取灵感，并使用二次能量和一种新的更新拉格朗日更新规则 来模拟弱可压缩性并追踪体积变化。我们进一步对基于离散拉普拉斯算子的黏度进行对称化，并提出一种离散二次势能 ，以提升精度和鲁棒性。我们遵循增量势接触 [Li 等人 2020] 的模型，施加有保证的可分离边界条件，并处理界面处的摩擦接触。

所提出的公式实现了强耦合，但使用牛顿法求解时效率可能极低，这是因为流体部分的海森矩阵规模庞大且稠密，而这又源于为实现精确的光滑粒子流体动力学离散化，需要大量粒子邻居。这会造成显著的计算瓶颈。

为解决这一问题，我们提出了一种稳健的接触能量代理公式，将时间积分拆分为流体阶段和固体耦合阶段。流体阶段每个时间步只需进行一次牛顿迭代，从而提高效率；非线性优化仅在固体耦合阶段进行。二次代理的一项关键优势是能够有效消除时间分裂导致的不稳定性。这得益于它对固液接触力的渐近逼近。此外，在固体耦合阶段通过抵消代理项的贡献，确保时间积分保持一致，因此分裂误差很小。最后，我们设计了矩阵自由共轭梯度求解器和域分解求解器，以进一步提高计算效率。

2 相关工作

传统流体求解器通常采用欧拉网格。许多现有工作致力于通过处理网格与不规则网格边界之间的相互作用，实现欧拉流体与拉格朗日固体的耦合。幽灵流体法 [Fedkiw 2002年；Fedkiw 等人 1999]被提出，用于进一步离散化欧拉/拉格朗日

作者地址：Tianyi Xie，tianyixie77@g.ucla.edu；Minchen Li，minchernl@gmail. com；YinYang，yin.yang@utah.edu；ChenfanfuJiang，chenfanfu.jiang@gmail.com。

界面。早期工作研究了弱耦合 [Guendelman 等人 2005], , 即交替推进固体和流体。另一方面，强耦合 [Klingner 等人 2006] 通过求解整体系统实现，通常更加稳健。切割单元法 [Roble 等人2005] 是另一种广泛使用的方案，通常借助虚拟节点实现。Batty 等人 [2007] 提出了一种变分框架，通过将压力求解转化为最小化问题，实现流体与刚体的强耦合。后续扩展支持可变形物体和薄壳 [Robinson-Mosher 等人 2011, 2008], , 其中弹性力被显式施加，而耦合步骤采用隐式处理。在假设材料满足共旋线性弹性的前提下，Zarifi 和 Batty [2017] 将隐式固体动力学纳入压力投影，从而得到一个对称正定系统。后续工作还研究了刚体—刚体 [Takahashi 和 Batty 2020] 以及刚体—流体 [Takahashi 和 Batty 2021] 之间的摩擦接触。欧拉固体 [Levin 等人 2011]也得到了研究，其耦合可通过纯欧拉方式便捷地实现 [Teng 等人 2016年；Valkov 等人 2015]。然而，这类方法面临数值耗散、体积守恒以及处理薄于一个网格单元的结构等方面的挑战。最近，Brandt 等人 [2019] 基于浸入边界法 [Peskin 2002]提出了一种降阶求解器，用于模拟实时耦合，重点关注不可压缩弹性材料和无滑移边界条件。

流体也可以直接使用拉格朗日网格建模 [Batty 等人，2012年；Clausen 等人，2013年；Klingner 等人，2006年；Wang 等人，2020], 从而实现与固体的显式耦合。不过，重新网格化往往会成为瓶颈。使用粒子是另一种常用策略。光滑粒子流体动力学（SPH） [Koschier 等人 2022] 采用空间采样来近似连续函数，并已被证明在流体动力学中效果显著。开创性工作 [Becker 和 Teschner，2007年；Monaghan1994] 将状态方程（EOS）用于弱可压缩流体，其中压力与密度偏差成正比。显式公式可能会严格限制时间步长。也可以通过求解压力泊松方程（PPE）来实现不可压缩性 [Bender 和 Koschier，2015年；Ihmsen 等，2013年；Solenthaler 和 Pajarola 2009]。这种方法利用压力加速度，消除非压力力造成的密度偏差或速度散度偏差。人们已开发出 SPH 边界处理技术，以防止流体粒子在固体边界附近发生穿透 [Becker 和 Teschner，2007年；Becker 等，2009b；Ihmsen 等人 2010]。Akinci 等人提出的一种方法 [2012], 使用单层边界采样点，并已应用于流体与刚体及弹性固体的耦合 [Akinci 等人 2013]。Gissler 等人 [2019] 提出了一种统一刚体与流体动力学的全局公式，其中流体压力求解器与一个用于刚体粒子的辅助人工压力求解器相连接。Koschier 和 Bender [2017]提出了另一种方法，使用密度图表示动态刚体边界，从而不再需要边界粒子。Bender 等人 [2019] 提出利用边界几何体的体积贡献来计算边界力，这降低了预计算成本，但无法应用于可变形物体。

Solenthaler 等人 [2007]使用 SPH 近似线弹性材料的变形梯度，但得到的梯度不具备旋转不变性。Becker 等人 [2009 a] 通过形状匹配确定方向，并计算旋转构型下的

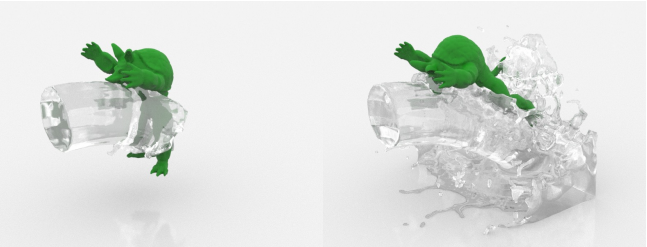


图2. **喷射水流的狒狒** 。

力。Peer 等人 [2018] 提出了一种隐式格式，并应用 Bonet 和 Lok 1999] 提出的核梯度修正 [，从而获得一种针对变形梯度的一阶一致 SPH 公式。将固体粒子纳入现有的流体压力求解器可以解决接触处理问题，但在模拟弹性物体时，SPH 仍面临零模态等数值问题 [Ganzenmüller 2015年；Kugelstadt 等人 2021] 。此外，在受压时，压力求解器会将固体物体视为不可压缩，这可能并不适用于所有情况。

物质点法（MPM） [Jiang 等人，2016年；Sulsky 等人1995] 结合了拉格朗日表示和欧拉表示，以捕捉固液耦合 [Fei 等人，2018年；Stomakhin 等人，2014年；Yan 等人2018] 以及混合物 [Gao 等人，2018年；Tampubolon 等人 2017]。Fang 等人 [2020] 提出了自由滑移处理，但未考虑分离。最近，有人提出了一种基于变分屏障公式的有限元法-物质点法耦合方法 [Li 等人 2020] ，用于耦合具有摩擦的可分离弹性材料 [Li 等人 2022]。我们的固液耦合方法受此方法启发，采用了类似的纯拉格朗日框架。

3 公式推导

在此，我们从控制方程出发，通过离散化推导固体与流体耦合系统的时间积分器。下标 s 和 f 分别表示固体量和流体量。

3.1 控制方程

耦合系统的控制方程为

$$\rho_s \frac{D\mathbf{v}_s}{Dt} = \nabla \cdot \sigma + \rho_s \mathbf{g} + \mathbf{f}_{s\to s} + \mathbf{f}_{f\to s}, \tag{1}$$

$$\rho_f \frac{D\mathbf{v}_f}{Dt} = -\nabla p + \mu \nabla^2 \mathbf{v}_f + \rho_f \mathbf{g} - \mathbf{f}_{f\to s}, \tag{2}$$

$$\nabla \cdot \mathbf{v}_f = 0, \tag{3}$$

其中， ρ 表示密度， $\mathbf{g}$ 表示重力， $\mathbf{f}_{s\to s}$ 表示固体的自接触力， $\mathbf{f}_{f\to s}$ 表示流体施加的接触力， σ 表示柯西应力， p 表示压力， μ 表示动力黏度 [Bridson 2015]。

在固体与流体的界面处，我们施加可分离边界条件

$$0 \le (\mathbf{v}_s - \mathbf{v}_f) \cdot \mathbf{n}_f \perp (\mathbf{f}_{f\to s} \cdot \mathbf{n}_f) \ge 0 \tag{4}$$

以防止穿透，同时允许分离 [Batty 等人 2007]。该条件有助于确定$\mathbf{f}_{f\to s}$的法向分量。对于切向分量（摩擦），令$\mathbf{u} = (\mathbf{I} - \mathbf{n}_f \otimes \mathbf{n}_f)(\mathbf{v}_s - \mathbf{v}_f)$

为切向相对速度，则有

$$\begin{aligned}(\mathbf{I}-\mathbf{n}_f\otimes\mathbf{n}_f)\mathbf{f}_{f\to s} &= \arg\min_{\boldsymbol{\beta}} \boldsymbol{\beta}\cdot\mathbf{u}\\ \text{s.t.}\quad \|\boldsymbol{\beta}\| &\le \mu_t\mathbf{f}_{f\to s}\cdot\mathbf{n}_f \ \text{ and } \ \boldsymbol{\beta}\cdot\mathbf{n}_f = 0\end{aligned} \tag{5}$$

遵循最大耗散原理 [Moreau 2011], 其中 μ_t 为摩擦系数。我们采用拉格朗日方法对两个域进行离散化，从而严格满足质量守恒。

3.2 固体域

我们关注非线性超弹性固体，其弹性力为弹性势能的负梯度。将固体域 Ω_s 离散为拉格朗日线性有限元（三角形用于二维，四面体用于三维）后，总弹性势能是弹性能量密度函数 $\psi_s(\mathbf{F})$（例如新胡克模型）在网格域上的分片常数求和： $\Psi_s(\mathbf{x}) = \sum_e V_e\psi_s(\mathbf{F}e)$，其中 V_e 是四面体 e的初始体积，且 $\mathbf{F} = \frac{\partial \mathbf{x}(\mathbf{X}\,t)}{\partial \mathbf{x}}$ 是变形梯度，其中 $\mathbf{X}$ 和 $\mathbf{x}$ 分别表示材料坐标和世界空间坐标 [Sifakis 和 Barbic 2012]。对于 $\mathbf{f}s\to s$，我们采用 Li 等人 [2020] 提出的光滑屏障方法，以保证不可穿透性。关于 $\mathbf{f}f\to s$的讨论见 § 3.4。

3.3 流体域

遵循光滑粒子流体动力学文献 [Becker 和 Teschner 2007年; Bender和Koschier，2015年；Ihmsen 等人 2013年; Macklin 和 Müller 2013], ，我们使用拉格朗日粒子离散化流体域 Ω_f。为了通过基于优化的时间积分来模拟流体，我们将压力和黏性力都近似为保守力。我们在附录中验证了这些提出的势能均为凸且为二次。

3.3.1 不可压缩势能。压力有助于维持不可压缩流体的体积。因此，我们通过二次能量密度函数 $\psi_{f,\,I}(J) = \frac{k_I}{2}(J-1)^2$ 来建模不可压缩性，该函数会惩罚体积比 $J = \rho_0/\rho$ 偏离 $\mathbf{T}$的情况，其中 ρ_0 为初始密度。在收敛求解中采用较大的刚度值（k_I），视觉压缩几乎可以忽略，因此对于近似不可压缩流体，无需使用高次多项式 [Hyde 等人 2020]。不可压缩势能通过在流体域 Ω_f^0 的材料空间中对 $\psi_{f,\,I}(J)$ 进行积分得到：

$$P_I(\mathbf{x}) = \sum_i \frac{k_I}{2}V_0(J_i(\mathbf{x})-1)^2, \tag{6}$$

其中，我们假设所有流体粒子的初始体积相等 V_0，而 J_i 表示第 i 个粒子的体积比，它是 $\mathbf{x}$ 的函数。

更新拉格朗日。 光滑粒子流体动力学文献通常通过世界空间中的密度求和，将 ρ 与$\mathbf{x}$ 联系起来。为了建立 J与$\mathbf{x}$ 之间的线性关系，使不可压缩势能关于$\mathbf{x}$保持为二次形式，我们以更新拉格朗日 方式跟踪 J。将 Ω^n 视为中间参考空间，并对 Ω^n 与 Ω^{n+1} 之间的变形映射求导，可得更新规则

$$J_i^{n+1} = J_i^n(1+h\nabla\cdot\mathbf{v}_i^{n+1}), \tag{7}$$



图 3. 搅拌**奶油**时，勺子发生了变形。

其中， J_i^n和 $\nabla\cdot\mathbf{v}_i^{n+1}$可近似为

$$J_i^n = \frac{\rho_0}{\sum_j m_j W_{ij}},\ \nabla\cdot\mathbf{v}_i^{n+1} = \sum_j \frac{m_j}{\rho_j^n}(\mathbf{v}_j^{n+1}-\mathbf{v}_i^{n+1})\cdot\nabla_i W_{ij} \tag{8}$$

通过光滑粒子流体动力学，且 $W_{ij} = W(\mathbf{x}_i - \mathbf{x}j)$ 是一个核函数（例如三次样条核 [Monaghan 1992, 2005] 或尖峰核 [Müller 等人 2003]）。这里， J_i^n表示在时间步 n 开始时重新初始化的第 i 个流体粒子的体积比。这种重新初始化避免了其他更新拉格朗日求解器（如物质点法）中常见的密度累积误差和粒子分布误差。

3.3.2 黏度势。可以通过应变率张量 [Bender 和 Koschier 2016年; Peer 等人2015年; Peer 和 Teschner 2016年; Takahashi 等人 2015] 来模拟黏性，但由于表面处粒子不足，可能会产生伪影。我们遵循 Monaghan [2005] 的方法，采用更稳健的速度拉普拉斯算子 [Weiler 等人 2018]，并推导其能量形式。结合 SPH 一阶导数与有限差分，可以计算出黏性力：

$$\mathbf{f}_i(\mathbf{x}) = \nu m_i\nabla^2\mathbf{v}_i^{n+1} = 2\nu(d+2)\sum_j \frac{m_i m_j}{\rho_j}\frac{\nabla_i W_{ij}(\mathbf{x}_{ij}^n)^T}{\|\mathbf{x}_{ij}^n\|^2+0.01\hbar^2}\mathbf{v}_{ij}^{n+1},$$

其中，$\mathbf{x}_{ij}^n = \mathbf{x}_i^n - \mathbf{x}_j^n$、$\mathbf{v}_{ij}^{n+1} = \mathbf{v}_i^{n+1} - \mathbf{v}_j^{n+1}$，$\hbar$ 为核函数的支撑半径， ν和 $d\in\{2,3\}$分别表示运动黏度和空间维度。直接应用该力会违反动量守恒，因为相互作用力并非大小相等、方向相反。因此，我们进一步进行了近似

$$\mathbf{f}_i(\mathbf{x}) \approx 4\nu(d+2)\sum_j \frac{m_i m_j}{\rho_i+\rho_j}\frac{\nabla_i W_{ij}(\mathbf{x}_{ij}^n)^T}{\|\mathbf{x}_{ij}^n\|^2+0.01\hbar^2}\mathbf{v}_{ij}^{n+1} \tag{9}$$

为了解决这一问题，并使力也可积。令$\mathbf{v}ij$ =4$(d+2)$ $\frac{m_i m_j}{\rho_i+\rho_j}$($-\nabla_i W_{ij}$ $\frac{n}{ij}$ ^{T})$(\mathbf{x})\mathbf{x}\hbar^2$ ’，现在我们可以汇集并积分所有黏性力，从而得 $\|\frac{n}{ij}\|^2$+0.01到一个二次黏性势能

$$P_V(\mathbf{x}) = \frac{1}{4}\nu\hat{h}\sum_i\sum_j \|\mathbf{v}_{ij}^{n+1}\|^2_{\mathbf{V}_{ij}}, \tag{10}$$

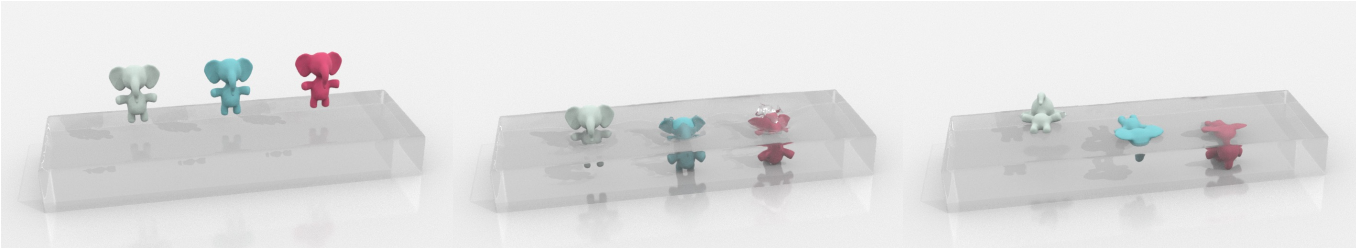


图 4. **浮力**。三个密度不同的弹性大象（从左到右：200、700 和 1200 kg/m^3）落入水中，展示了浮力。

其中，h 是一个与时间积分方案有关的常数标量。例如，对于隐式欧拉法，$h = h$ 为$\mathbf{v}^{n+1} = (\mathbf{x}^{n+1} - \mathbf{x}^n)/h$。

3.4 耦合

3.4.1 非穿透障碍势。为了耦合固体域 Ω_s与流体域 Ω_f，我们使用可分离边界条件（式 4），以强制这两个域之间满足非互穿约束。为了对这些约束进行建模，我们首先定义一个距离函数

$$d(\partial\Omega_s^t, \mathbf{x}_f) = \min_{\mathbf{x}_s} \|\mathbf{x}_s - \mathbf{x}_f\|, \quad \mathbf{x}_s \in \partial\Omega_s^t,\ \mathbf{x}_f \in \Omega_f^t, \tag{11}$$

该函数用于测量流体域中的点 $\mathbf{x}f$ 与固体域表面之间的距离。然后，约束的原始分量可表示为

$$d(\partial\Omega_s^t, \mathbf{x}_f) \geq 0, \quad \forall t \geq 0,\ \forall \mathbf{x}_f \in \Omega_f^t. \tag{12}$$

随后，我们采用 Li 等人 [2020]提出的障碍形式，对式 4 中固体与流体之间的所有约束进行建模，并得到一个障碍势

$$\int_{\partial\Omega_f^0} b(d(\partial\Omega_s^t, \mathbf{x}_f), \hat{d}) d\mathbf{X}_f, \tag{13}$$

ˆ其中，障碍能量密度 $b(d,\ d)$ 是分段光滑的，且仅在 $d < d$ 时激活，从而提高效率并近似满足互补松弛条件。随着 d趋近于 ˆ 0，$b(d,\ d)$ 的值单调增至无穷大，提供任意大的排斥力以避免相互穿透。

由于我们的固体域和流体域分别离散为网格和粒子，因此三维情况下的障碍势（式 13）可通过数值积分计算为

$$\begin{aligned} B_{\mathrm{sf}}(\mathbf{x}_s, \mathbf{x}_f) &= \sum_{q\in Q_f} s_q b(\min_{e\in\mathcal{B}_s} d^{PT}(\mathbf{x}_q, e), \hat{d}) \\ &= \sum_{q\in Q_f} s_q \max_{e\in\mathcal{B}_s} b(d^{PT}(\mathbf{x}_q, e), \hat{d}), \end{aligned} \tag{14}$$

其中，Qf 是所有 SPH 流体粒子的集合，Bs 是所有固体边界三角形的集合，$s_q = \pi(\frac{3V_0}{4\pi})^{\frac{2}{3}}$是每个流体粒子的积分权重（边界面积），而 $d_{PT}(\mathbf{x}q,\ e)$表示粒子$\mathbf{x}q$与三角形 e之间的距离。这里的最小-最大变换基于障碍函数的非递增性质。然而，此处的最大值运算符使障碍势难以通过基于梯度的方法高效优化。幸运的是，由于障碍函数的局部支撑性质

ˆ ˆ 函数 $b(d,\ d)$在 d较小时，我们可以简单地将障碍势近似为

$$B_{\mathrm{sf}}(\mathbf{x}_s, \mathbf{x}_f) = \sum_{q\in Q_f} \sum_{e\in\mathcal{B}_s} s_q b(d^{PT}(\mathbf{x}_q, e), \hat{d}), \tag{15}$$

这可能会导致网格边界上的边缘和节点附近的接触力被高估，但我们在实验中未观察到任何伪影。

3.4.2 摩擦势。按照 Li 等人 [2020],的方法，我们对每个处于激活状态的固体-流体接触对 k 建立局部摩擦力 $\mathbf{f}_k$ 模型。形式上，摩擦力定义为

$$\mathbf{f}_k(\mathbf{x}_s, \mathbf{x}_f) = -\mu_t \lambda_k T_k(\mathbf{x}_s, \mathbf{x}_f) f_1(\|\mathbf{u}_k\|) \frac{\mathbf{u}_k}{\|\mathbf{u}_k\|}, \tag{16}$$

其中，λ_k是接触力大小，$T_k(\mathrm{x}s, \mathrm{x}f) \in \mathbb{R}^{3n\times 2}$是方向一致的滑动基，$\mathbf{u}_k$ 是相对滑动位移，可计算为 u

$$_k = T_k \quad _s \quad _f{}^T \ [\ _s^T \quad _f^T]^T - (\mathrm{x}\ ,$$

$[\ _s^n \qquad\qquad _f^1]^T$)。这里的 是一个经过平滑近似的函数，用于实现粘着模式与滑动模式之间的平滑过渡。为了使这一摩擦公式适用于优化时间积分，Li 等人 [2020] 进一步将滑动基 $T(\mathrm{x}s, \mathrm{x}f)$和接触力 $\lambda_k(\mathrm{x}s, \mathrm{x}f)$显式近似为 $T(\mathrm{x}_s^n, \mathrm{x}_f^n)$和 $\lambda_k(\mathrm{x}_s^n, \mathrm{x}_f^n)$。随后，半隐式摩擦力可通过以下摩擦势进行积分计算：

$$D_{\mathrm{sf}}(\mathbf{x}_s, \mathbf{x}_f) = \sum_{k\in\mathcal{A}^n} \mu_t \lambda_k^n f_0(\|\mathbf{u}_k\|), \tag{17}$$

其中，f_0 由关系式 $f\prime_0 = f_1$ 定义，A^n 是包含上一个时间步所有已激活粒子-三角形接触对的集合 n。

3.5 优化时间积分器

上述势能已对所有固体力和流体力进行建模，现在我们可以构建统一的双向固液耦合框架。将 SPH 粒子和有限元节点的所有节点位置和速度依次堆叠为 $\mathbf{x} = [\mathbf{x}_f^T, \mathbf{x}_s^T]^T$和 $\mathbf{v} = [\mathbf{v}_f^T, \mathbf{v}_s^T]^T$，我们定义 $\Psi(\mathbf{x}) = \Psi_s(\mathbf{x})$、$P(\mathbf{x}) = P_I(\mathbf{x}) + P_V(\mathbf{x})$ 和 $C\mathrm{sf}(\mathbf{x}) \mathrm{sf}(\mathrm{x}\ _s, \mathrm{x}f) \mathrm{sf}(\mathrm{x}\ _s, \mathrm{x}f)$。结合 IPC 中的固体间接触势能$C\mathrm{ss}(\mathbf{x})$，我们的固液耦合问题可以

$B \qquad + D$

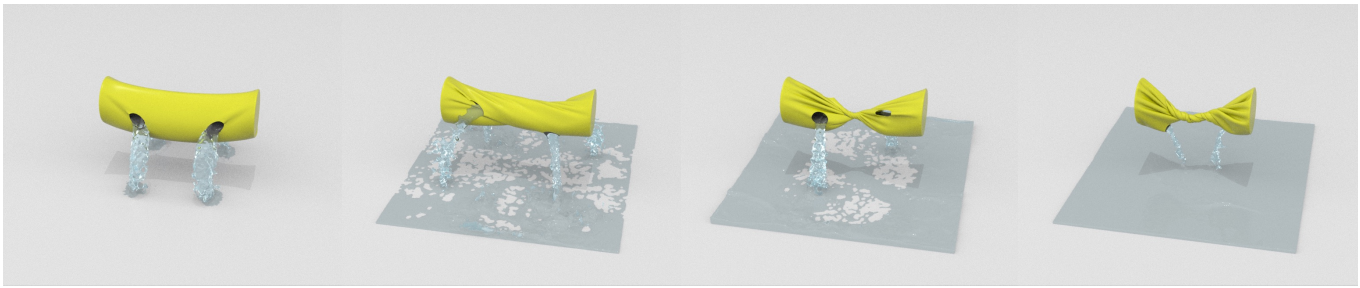


图5. **扭转圆柱**。一个带有四个孔洞的圆柱形布料被扭转，将内部的水挤出。

采用隐式欧拉法进行时间积分，以整体式方式求解

$$\begin{cases} \mathbf{v}^{n+1} = \mathbf{v}^n + h\mathbf{M}^{-1}(\mathbf{f}_{\text{ext}} - \nabla P(\mathbf{x}^{n+1}) - \nabla \Psi(\mathbf{x}^{n+1}) - \nabla C(\mathbf{x}^{n+1})) \\ \mathbf{x}^{n+1} = \mathbf{x}^n + h\mathbf{v}^{n+1} \end{cases}, \tag{18}$$

这等价于

$$\mathbf{x}^{n+1} = \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{x} - \hat{\mathbf{x}}^n\|_{\mathbf{M}}^2 + h^2(P(\mathbf{x}) + \Psi(\mathbf{x}) + C(\mathbf{x})) \tag{19}$$

其中，质量矩阵为 $\mathbf{M}$，时间步长为 h，预测位置为 $\hat{\mathbf{x}}^n = \mathbf{x}^n + h\mathbf{v}^n + h^2\mathbf{M}^{-1}\mathbf{f}$ ext，总接触势能为 $C(\mathbf{x})$ $= C \quad + C$ $\text{sf}(\mathbf{x})$ $\text{ss}(\mathbf{x})$。

4 高效求解器

稳健求解时间步进优化问题（式 19）的一种直接方法是采用带线搜索的投影牛顿法 [Li 等人 2020]。在每次迭代中，搜索方向 $\mathbf{p}$可通过求解线性系统来计算

$$\begin{bmatrix} \mathbf{H}_f & \mathbf{G} \\ \mathbf{G}^T & \mathbf{H}_s \end{bmatrix} \mathbf{p} = \begin{bmatrix} \mathbf{g}_f \\ \mathbf{g}_s \end{bmatrix}. \tag{20}$$

这里，$\mathbf{H}_f$和$\mathbf{H}_s$ 分别表示关于流体和固体位置的（投影）海森矩阵，$\mathbf{G} = {}_{\partial}{}^{\partial^2 E}{}_{\mathbf{x}_f \partial \mathbf{x}_s}$ 表示耦合子矩阵。然而，在实践中求解该线性系统可能会成为严重的瓶颈。原因之一是，光滑粒子流体动力学技术需要足够多的邻居，才能准确近似物理量，因此流体海森矩阵$\mathbf{H}_f$比固体海森矩阵大得多、也稠密得多。此外，由于屏障能的陡峭性，优化可能需要许多次迭代才能收敛，在接触频繁的情况下尤其如此。

由于流体能量均为二次形式，我们通过一种稳健的时间分裂方案（§ 4.1）将其与高度非线性的固体和接触能量分离，从而使流体部分在每个时间步内只需进行一次牛顿迭代即可求解。随后，我们提出了求解域分解线性系统的高效方法（§ 4.2）。

4.1 时间分裂

4.1.1 基线时间分裂。直观而言，我们可以将原始时间积分拆分为一个**流体阶段**

$$\begin{cases} \mathbf{v}_f^{n+1/2} = \mathbf{v}_f^n + h\mathbf{M}_f^{-1}(-\nabla_f P([(\mathbf{x}_f^{n+1/2})^T, (\mathbf{x}_s^n)^T]^T) + \mathbf{f}_f) \\ \mathbf{x}_f^{n+1/2} = \mathbf{x}_f^n + h\mathbf{v}_f^{n+1/2} \end{cases} \tag{21}$$

和一个**固体耦合阶段**

$$\begin{cases} \mathbf{v}^{n+1} = \begin{bmatrix} \mathbf{v}_f^{n+1/2} \\ \mathbf{v}_s^n \end{bmatrix} + h\mathbf{M}^{-1}(-\nabla \Psi(\mathbf{x}^{n+1}) - \nabla C(\mathbf{x}^{n+1}) + \begin{bmatrix} \mathbf{0} \\ \mathbf{f}_s \end{bmatrix}) \\ \mathbf{x}^{n+1} = \mathbf{x}^n + h\mathbf{v}^{n+1} \end{cases}, \tag{22}$$

其中，$\mathbf{f}_f$和$\mathbf{f}_s$ 分别表示作用于流体和固体的外力。在流体阶段，我们通过一次牛顿迭代求解流体粒子的中间状态，并忽略接触。随后，在固体耦合阶段同时考虑弹性并处理高度非线性的障碍力，此时流体海森矩阵$\mathbf{H}_f$会简化为块对角矩阵$\partial^2 C(\mathbf{x}) \, \partial \mathbf{x}_f^2$。在这种设定下，非线性优化仅在固体耦合阶段针对流体边界和固体自由度进行。该基线时间分裂方案的详细内容见附录。

尽管这种基线分裂策略确实显著提升了性能，但如果时间步长不够小，固液界面处仍可能出现严重的不稳定现象，尤其是在模拟黏性流体时（图7）。例如，流体粒子可能会黏附在固体边界上。这种伪影在现有的SPH流体求解器中也能观察到，通常通过在固体边界处采样粒子来施加边界压力予以解决 [Akinci 等，2012；Becker 等，2009b；Ihmsen 等人 2010]。鉴于此，我们始终为流体阶段添加用于固液接触的代理力，以提高稳定性，同时避免粒子采样带来的额外开销。

4.1.2 带接触代理的时间分裂。我们将固液接触代理能量 $\hat{C}$ $\text{sf}(\mathbf{x})$ 引入流体阶段，以高效地施加固体与流体边界之间的近似相互作用力。在下面的讨论中，为了清晰起见，我们也将接触能量 $C(\mathbf{x})$ 表示为固液部分（$C\text{sf}(\mathbf{x})$）和固体-固体部分（$C\text{ss}(\mathbf{x})$）之和。为确保与原始



图6. **Bob** 分别采用（A）联合优化、（B）带接触代理的时间分裂和（C）基线时间分裂进行模拟。对于此示例，基线时间分裂也能产生视觉上合理的结果，而我们的代理辅助方案比联合优化快 3×。

偏微分方程保持一致，我们在固液耦合阶段中消去该接触代理的贡献。由此得到的时间积分为

$$\begin{cases} \mathbf{v}^{n+1/2} = \mathbf{v}^n + h\mathbf{M}^{-1}(-\nabla P(\mathbf{x}^{n+1/2}) + \mathbf{f}_{\text{ext}} - \nabla \hat{C}_{\text{sf}}(\mathbf{x}^{n+1/2})) \\ \mathbf{x}^{n+1/2} = \mathbf{x}^n + h\mathbf{v}^{n+1/2} \end{cases}$$
$$\begin{cases} \mathbf{v}^{n+1} = \mathbf{v}^{n+1/2} + h\mathbf{M}^{-1}(-\nabla \Psi(\mathbf{x}^{n+1}) - \nabla C(\mathbf{x}^{n+1}) + \nabla \hat{C}_{\text{sf}}(\mathbf{x}^{n+1})) \\ \mathbf{x}^{n+1} = \mathbf{x}^n + h\mathbf{v}^{n+1} \end{cases} \tag{23}$$

此时，流体阶段也会隐式地将流体附近的固体边界节点更新至一个中间状态，并显式更新所有其他固体节点。

对于 $\hat{C}\mathbf{sf}(\mathbf{x})$，一种直接的选择是 $\hat{C}\mathbf{sf}(\mathbf{x}) = \frac{1}{2}C\mathbf{sf}(\mathbf{x})$。但为了确保流体阶段仍然只包含线性力，我们在 $\mathbf{x}^n$ 处对 $\frac{1}{2}C\mathbf{sf}(\mathbf{x})$ 进行二阶泰勒展开，以用于流体阶段中的近似，即

$$\hat{C}_{\text{sf}}(\mathbf{x}) = \frac{1}{2}\left(C_{\text{sf}}(\mathbf{x}^n) + \nabla C_{\text{sf}}(\mathbf{x}^n)(\mathbf{x} - \mathbf{x}^n) + \frac{1}{2}\|\mathbf{x} - \mathbf{x}^n\|^2_{\nabla^2 C_{\text{sf}}(\mathbf{x}^n)}\right), \tag{24}$$

而在固体耦合阶段，我们直接使用 $\hat{C}\mathbf{sf}(\mathbf{x}) = \frac{1}{2}C\mathbf{sf}(\mathbf{x})$。在附录 . 中，我们证明，与隐式欧拉解相比，我们的带接触代理的时间分裂方案仅存在 $O(h^4)$ 的偏差。将两个阶段（式 23）重新表述为优化问题后，我们得到

$$\begin{aligned} \mathbf{x}^{n+1/2} &= \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{x} - \hat{\mathbf{x}}^n\|^2_{\mathbf{M}} + h^2(P(\mathbf{x}) + \hat{C}_{\text{sf}}(\mathbf{x})), \\ \mathbf{x}^{n+1} &= \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{x} - \mathbf{x}^{n+1/2}\|^2_{\mathbf{M}} + h^2(\Psi(\mathbf{x}) + \frac{1}{2}C_{\text{sf}}(\mathbf{x}) + C_{\text{ss}}(\mathbf{x})), \end{aligned} \tag{25}$$

其中 $\hat{\mathbf{x}}^n = \mathbf{x}^n + h\mathbf{v}^n + h^2\mathbf{M}^{-1}\mathbf{fext}$ 与 $\mathbf{fext} = [\mathbf{f}_f^T, \mathbf{f}_s^T]^T$。

除了无需额外昂贵开销即可避免流体粒子粘连问题外，我们的方法还有另一项优势：有助于减少求解问题所需的牛顿迭代次数。通常，障碍法在处理高速冲击时需要进行许多次牛顿迭代。采用我们的方案时，高速流体粒子与可变形物体发生碰撞后，会因接触代理的作用而在流体阶段结束时显著减速。随后，这一较低的速度会被带入固体耦合阶段，从而使非线性优化更容易求解（因为接触约束集合的变化更少）。基于代理的时间分裂方案的具体细节见算法 1。

算法1 带接触代理的时间分裂

$\mathbf{x} \leftarrow \mathbf{x}^n$, $\hat{\mathbf{x}}^n \leftarrow \mathbf{x}^n + h\mathbf{v}^n + h^2\mathbf{M}^{-1}\mathbf{f}_{\text{ext}}$
SPH Neighbor Search & Density Update
$\hat{C}_{\text{sf}}(\mathbf{x}) \leftarrow$ 2nd Taylor Expansion of $\frac{1}{2}C_{\text{sf}}(\mathbf{x})$ at $\mathbf{x} = \mathbf{x}^n$
// Fluid Phase
$\mathbf{H} \leftarrow h^2\left(\nabla^2 P(\mathbf{x}) + \nabla^2 \hat{C}_{\text{sf}}(\mathbf{x})\right) + \mathbf{M}$
$\mathbf{p} \leftarrow -\mathbf{H}^{-1}\left(h^2(\nabla P(\mathbf{x}) + \nabla \hat{C}_{\text{sf}}(\mathbf{x})) + \mathbf{M}(\mathbf{x} - \hat{\mathbf{x}}^n)\right)$
$\mathbf{x} \leftarrow \mathbf{x} + \mathbf{p}$
$\mathbf{x}^{n+1/2} \leftarrow \mathbf{x}$
// Solid-Coupling Phase
do
  $\mathbf{H} \leftarrow h^2\left(\nabla^2\Psi(\mathbf{x}) + \frac{1}{2}\nabla^2 C_{\text{sf}}(\mathbf{x}) + \nabla^2 C_{\text{ss}}(\mathbf{x})\right) + \mathbf{M}$
  $\mathbf{g} \leftarrow h^2\left(\nabla\Psi(\mathbf{x}) + \frac{1}{2}\nabla C_{\text{sf}}(\mathbf{x}) + \nabla C_{\text{ss}}(\mathbf{x})\right) + \mathbf{M}(\mathbf{x} - \mathbf{x}^{n+1/2})$
  $\mathbf{p} \leftarrow -\mathbf{H}^{-1}\mathbf{g}$
  $\alpha \leftarrow$ Backtracking Line Search with CCD
  $\mathbf{x} \leftarrow \mathbf{x} + \alpha\mathbf{p}$
while $\frac{1}{h}\|\mathbf{p}\| > \epsilon$
$\mathbf{x}^{n+1} \leftarrow \mathbf{x}$, $\mathbf{v}^{n+1} \leftarrow (\mathbf{x} - \mathbf{x}^n)/h$
return $\mathbf{x}^{n+1}, \mathbf{v}^{n+1}$

同样，也可以利用接触代理将弹性与接触能量分离。这样一来，我们便得到一个三阶段（流体、固体和接触）时间分裂方案

$$\begin{aligned} \mathbf{x}^{n+1/3} &= \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{x} - \hat{\mathbf{x}}^n\|^2_{\mathbf{M}} + h^2(P(\mathbf{x}) + \hat{C}_{\text{sf}}(\mathbf{x})), \\ \mathbf{x}^{n+2/3} &= \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{x} - \mathbf{x}^{n+1/3}\|^2_{\mathbf{M}} + h^2(\Psi(\mathbf{x}) + \hat{C}_{\text{sf}}(\mathbf{x}) + \hat{C}_{\text{ss}}(\mathbf{x})), \\ \mathbf{x}^{n+1} &= \arg\min_{\mathbf{x}} \frac{1}{2}\|\mathbf{x} - \mathbf{x}^{n+2/3}\|^2_{\mathbf{M}} + h^2(\frac{1}{3}C_{\text{sf}}(\mathbf{x}) + \frac{1}{2}C_{\text{ss}}(\mathbf{x})), \end{aligned} \tag{26}$$

其中，$\hat{C}\mathbf{sf}(\mathbf{x})$ 和 $\hat{C}\mathbf{ss}(\mathbf{x})$ 分别是 $\frac{1}{3}C\mathbf{sf}(\mathbf{x})$ 和 $\frac{1}{2}C\mathbf{ss}(\mathbf{x})$ 的二阶泰勒展开。然而，-这种激进的分裂方案仅适用于抗反转本构模型，例如固定共旋模型 [Stomakhin 等人 2012]。虽然在考虑弹性能的固体相中可以保证避免反转，但在接触阶段可能无法满足这一点。尽管存在这一限制，三阶段分裂方案在实践中仍能针对抗反转本构模型正常工作，从而进一步加速模拟。

4.2 求解线性系统

在我们的时间分裂方案中，求解大型稀疏线性系统占据了每个阶段的主要计算和内存开销。因此，我们设计了基于无矩阵和舒尔补的策略，以高效地求解这些系统。

4.2.1 流体阶段。 由于我们的公式需要考虑 SPH 粒子的二环邻居，因此构造海森矩阵并对其直接进行因式分解都会耗费大量时间和内存。为此，我们设计了一种无矩阵共轭梯度（CG）求解器，以高效求解流体的中间状态。

由于此阶段的所有能量势都是二次的，能量梯度 $\mathbf{g}(\mathbf{x})$ 只是关于 $\mathbf{x}$ 的线性函数，其系数矩阵 $\mathbf{H}(\mathbf{x})$ 为常数。因此，$\mathbf{H}(\mathbf{x})$ 与任意向量$\mathbf{p}$ 的乘积可表示为

$$\mathbf{H}(\mathbf{x})\mathbf{p} = \mathbf{g}(\mathbf{p}) - \mathbf{g}(\mathbf{0}). \tag{27}$$

这样我们便可以通过计算梯度来求得矩阵-向量乘积；对于共轭梯度求解器中的块雅可比预条件，我们只需获取海森矩阵的 3×3 对角块。

4.2.2 固体耦合阶段。 由于此阶段不包含流体势能，海森矩阵的各组成部分变为

$$\mathbf{H}_f = \frac{\partial^2 C(\mathbf{x})}{\partial \mathbf{x}_f^2}, \quad \mathbf{G} = \frac{\partial^2 C(\mathbf{x})}{\partial \mathbf{x}_s \partial \mathbf{x}_f}, \quad \mathbf{H}_s = \frac{\partial^2 C(\mathbf{x})}{\partial \mathbf{x}_s^2} + \frac{\partial^2 \Psi(\mathbf{x})}{\partial \mathbf{x}_s^2}. \tag{28}$$

尽管这个线性系统已不再那么难以求解，但当流体分辨率较高时，鉴于$\mathbf{H}f$和$\mathbf{G}$中的非零元素数量相当多，直接对整个系统进行分解并非最优选择。

因此，我们设计了一种域分解线性求解器，分别处理$\mathbf{H}f$和$\mathbf{H}_s$。基于舒尔补 [Zhang 2006]，我们的海森矩阵的逆可表示为

$$\mathbf{H}^{-1} = \begin{bmatrix} \mathbf{H}_f^{-1} + \mathbf{H}_f^{-1}\mathbf{G}(\mathbf{H}/\mathbf{H}_f)^{-1}\mathbf{G}^T\mathbf{H}_f & -\mathbf{H}_f^{-1}\mathbf{G}(\mathbf{H}/\mathbf{H}_f)^{-1} \\ -(\mathbf{H}/\mathbf{H}_f)^{-1}\mathbf{G}^T\mathbf{H}_f & (\mathbf{H}/\mathbf{H}_f)^{-1} \end{bmatrix}, \tag{29}$$

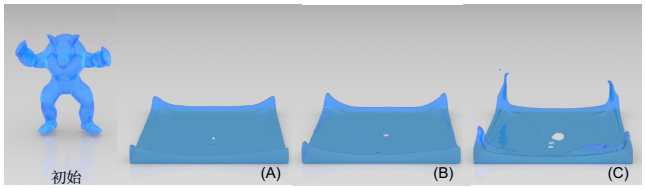
其中$\mathbf{H}/\mathbf{H}f = \mathbf{H}_s - \mathbf{G}^T\mathbf{H}_f^{-1}\mathbf{G}$ 是块$\mathbf{H}f$的舒尔补。由于$\mathbf{H}f$的非零元素仅存在于对角块中，因此很容易求得其逆矩阵$\mathbf{H}_f^{-1}$。随后，我们可以使用 CHOLMOD [Chen 等人 2008] 提出的 LLT求解器，对$\mathbf{H}/\mathbf{H}f$进行分解；该矩阵的维数仅与固体自由度的数量相当。之后，可通过矩阵-向量乘积和回代计算搜索方向。没有固液相互作用时，$\mathbf{H}/\mathbf{H}f$的稀疏模式与$\mathbf{H}_s$保持一致。只有当两个固体节点 i和 j与同一个流体粒子发生相互作用时， 3×3 块 $(\mathbf{H}/\mathbf{H}f)_{i,j}$（在 3D 中）才会变为非零。通常，这种情况只会发生在相邻的网格图元之间，因此其稀疏模式

$\mathbf{H}/\mathbf{H}f$大多不错。

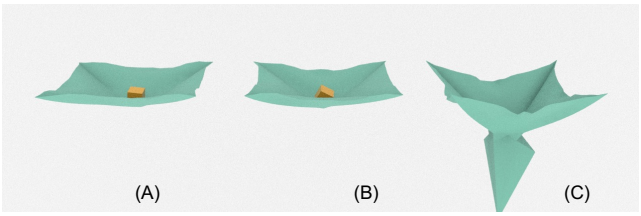
请注意，使用三阶段时间分裂方案（公式 26）时，我们的域分解求解器也可应用于固体阶段和接触阶段，因为它们的系统与此处的固体耦合阶段具有相似的结构。

5 实验与评估

我们的代码使用 C++ 实现，并采用 Eigen 执行基本的线性代数运算，采用 Intel TBB 实现多线程。所有模拟的时间步长均根据 SPH CFL 条件和用户定义的上限自适应选取。我们将 SPH 核函数的支撑半径设为 $2d$，其中 d 为粒子直径。在实现中，我们使用三次样条核进行密度估计，并使用尖峰核进行梯度计算。对于图5、4、2和6，我们采用三阶段时间分裂方案，展示其在本构模型与网格反转兼容时的有效性。对于其余模拟，我们采用两相时间分裂方案。除与 ElastoMonolith [Takahashi 和 Batty 2022]进行比较研究外，大多数实验均在一台 24 核 3.50GHz Intel i9-10920X 机器上完成。我们证明了该方法能够实现高效且稳健的固液耦合。所有模拟的参数和计时分解分别列于表1和图12。



（a）一只**黏性犰狳**被扔到地面上。



（b）**布上的立方体**。一个弹性立方体被扔到一块四角固定的方形布上。

图7. (A) 联合优化、(B) 带接触代理的时间分裂和 (C) 基线时间分裂的模拟结果。直接应用时间分裂会导致边界处不稳定，而使用接触代理得到的结果与联合优化一致。

5.1 消融研究

5.1.1 时间分裂评估。 进行了三项模拟（图6和图7），以展示时间分裂的效率，以及我们提出的接触代理在维持稳定性方面的有效性。

首先，我们需要妥善选择时间步长 h。首先，它必须受到CFL条件的限制。否则，由于SPH近似误差，可能会出现严重的体积损失。此外，与联合优化（公式 19）相比，时间分裂方案通常需要更小的时间步长才能保持稳定，这带来了第二个时间步长约束。然而，我们观察到，在实践中，即使使用最大的CFL时间步长，我们的代理辅助时间分裂方案仍能正常运行，并产生稳定的仿真结果。因此，为了进行比较，我们对两种方案都使用最大的CFL时间步长，以最大限度地发挥其性能，因为较小的 h 通常需要进行更多次牛顿迭代才能完成一帧的仿真。对于联合优化，由于直接分解难以处理，我们使用流体部分矩阵自由的块雅可比预条件共轭梯度求解器求解公式 20。

场景	$\Delta t_{\text{帧}}$	Δt	$N_{\text{流体}}$	$N_{\text{固体}}$	k_I	ν_f	d	ρ_f	E	ν_s	ρ_s	T
图6 Bob	1/24	4×10^{-3}	97K	2.3K	2×10^5	0	15	1000	1×10^5	0.3	500	0.3
图 7a 黏性犰狳	1/48	4×10^{-3}	238K	0	1×10^5	100	10	1200	-	-	-	0.4
图2 射击犰狳	1/24	4×10^{-3}	103K	16K	1×10^5	0	10	1000	1×10^5	0.3	200	1.3
图9 溃坝	1/24	5×10^{-3}	280K	0	2×10^5	0.005	25	1000	-	-	-	0.4
图10a 液体兔子	1/50	4×10^{-3}	52K	3.7K	1×10^5	0	10	1000	4×10^3	0.49	200	0.4
图10b 液体兔子	1/50	4×10^{-3}	101K	4.5K	6×10^4	0	6.4	1000	1×10^3	0.49	200	1.0
图4 浮力	1/24	5×10^{-3}	787K	66K	2×10^5	1	10	1000	1×10^5	0.4	200/700/1200	5.9
图5 扭转圆柱*	1/24	5×10^{-3}	486K	12K	4×10^4	0	5	1000	-	-	500	7.9
图3 奶油	1/24	4×10^{-3}	159K	9K	3×10^4	25	3	1000	5×10^8	0.49	1000	1.8
图13 愤怒的奶牛*	1/24	5×10^{-3}	789K	13K	1×10^5	0.2	10	1000	1×10^5	0.45	100/700	4.9
图1 踢水*	1/24	6×10^{-3}	1M	43K	2.5×10^5	0.1	25	1000	-	-	500	37.9

表1. **仿真统计**包括每帧的时长（Δt_{frame}，$[s]$）、时间步长上限（Δt，$[s]$）、流体粒子数量（N_{fluid}）、固体顶点数量（N_{solid}）、不可压缩系数（k_I，$[Pa]$）、动力黏度（$^{\nu}f$，$[Pa\cdot s]$）、流体粒子直径（d，$[mm]$）、流体密度（ρf，$[kg/m^3]$）、杨氏模量（E，$[Pa]$）、泊松比（ν_s）、固体密度（ρ_s，$[kg/m^3]$），以及每帧的平均仿真时间（T，$[min]$）。除图10外，计时统计均在一台 24 核 3.50GHz Intel i9-10920X 机器上测得；图10则在 Google Compute Engine 的“e2-standard-8”（8 核，32GB 内存）上测试。请注意，标有 * 的示例包含余维材料，其参数设置未在此处列出。

场景	方案	秒/帧	# 牛顿迭代次数/帧
图6	联合优化/时间分裂/接触代理时间分裂	66.1 / 38.0 / 22.5	63.5 / 117.3 / 37.1
图 7a	联合优化/时间分裂/接触代理时间分裂	41.3 / 32.3 / 25.5	16.5 / 29.0 / 10.5

表 2. **不同时间步进方案的统计数据**：联合优化（Joint）、基线时间分裂（TS）和带接触代理的时间分裂（TSCP）。我们提出的 TSCP 比 Joint 和 TS 都快得多。

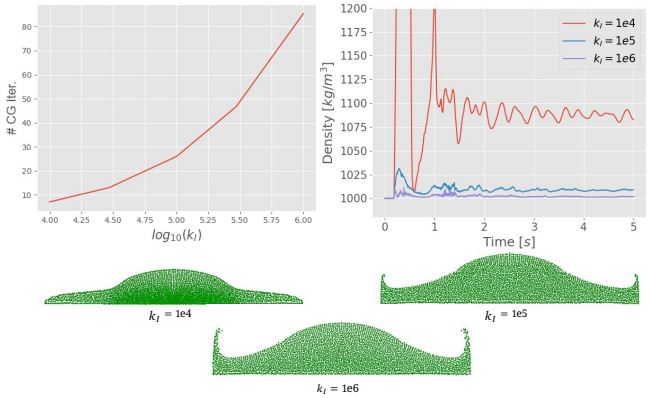


图 8. **不同刚度参数下模拟的统计数据** k_I。较大的 k_I 能更好地保持体积，但会导致更多共轭梯度迭代。在这种情况下，可以设置合适的 $k_I = 10^5$ Pa，以平衡计算成本和视觉伪影。

如表 2所示，即使在这些简单的示例中，我们的时间分裂方案也明显比联合优化快（最高可达 3×），尤其是在涉及流体与可变形固体之间接触的情况（例如图6）中。这一改进源于不再需要在一个时间步内反复求解流体的不可压缩性。此外，还可以发现，采用我们的代理辅助时间分裂方案时，牛顿迭代次数大幅减少。正如

如 § 4.1.2 所述，这是因为具有挑战性的高速冲击已在流体阶段得到部分求解。时间分裂的另一个优点是可以为两个阶段设置不同的误差容限。流体阶段的误差源于共轭梯度求解器的解偏差，而固体相中的误差则由牛顿法的容限直接控制。通常，为流体设置稍高的容限可以提升性能，同时仍能产生视觉上合理的结果。

除了提升效率外，我们提出的接触代理还能提高时间分裂方案的稳定性。基线时间分裂方案在无黏流体的情况下模拟效果尚可，但应用于黏性流体时，情况会变得更糟。在图 7a 中，一只黏性犰狳被扔到地面上。在这个例子中，基线时间分裂方案会在边界处产生严重的黏滞伪影，流体表面最终也无法平静下来。正如图 7a 所示，在流体阶段持续使用我们的接触代理施加边界压力，可以很好地消除这些伪影。同样，我们的接触代理思路也适用于

在保持稳定性的同时进一步将弹性与 IPC 接触分离，从而形成我们的三相方案（图 7b）。

5.1.2 线性求解器评估。 对于流体阶段，我们设计了一种无矩阵共轭梯度（CG）求解器，通过梯度计算矩阵-向量乘积，从而避免直接分解（§ 4.2.1）所带来的高昂计算和内存开销。然而，如果收敛所需的共轭梯度迭代次数过多，这种方法带来的性能提升将不再显著，因为此时计算梯度的成本会高于一次性构建海森矩阵的成本。在我们的流体阶段中，共轭梯度迭代次数与不可压缩性能量的刚度 k_I 成正比。较大的 k_I 能够更好地保持流体的体积，但也会导致系统的条件变差，需要更多迭代才能收敛（图 8）。实际上，通过将 k_I 设为一个

求解器	流体阶段		固体相	接触阶段	Mem.
	hess	求解	求解	求解	
共轭梯度 + LLT	14.9	0.49	1.45	0.43	12375
Ours	0.15	0.59	1.11	0.25	1469

表3。**示例2中不同求解器的时间和内存开销**。开销按每个时间步测量，单位分别为 [s] 和 [MB] 。基线方法在流体阶段使用共轭梯度（CG）方法，在固体阶段和接触阶段使用 CHOLMOD LLT。相比之下，我们的方法在流体阶段采用矩阵自由共轭梯度求解器，在固体阶段和接触阶段采用域分解求解器，从而提高效率并节省内存。

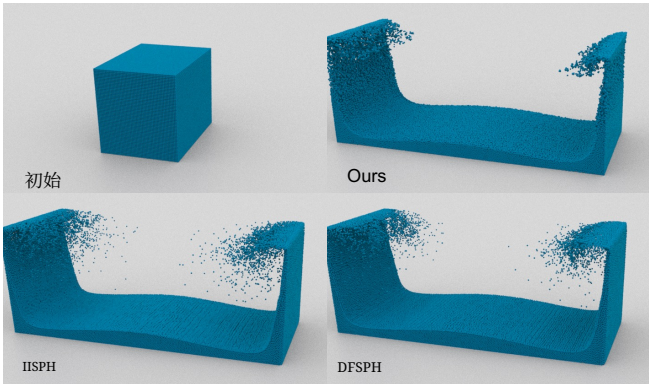


图9。**溃坝**，包含 280K 个 SPH 粒子。我们的弱可压缩公式能够产生稳定的流体动力学效果，且没有明显的体积损失。与不可压缩 SPH 求解器 IISPH [Ihmsen 等人 2013]和 DFSPH [Bender 和 Koschier 2015], 相比，我们的模拟结果展现出更均匀的粒子分布。

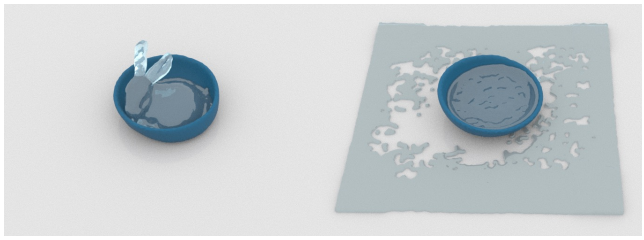
合适的值，我们可以在 50 次共轭梯度迭代内高效地求解系统，同时不会出现明显的流体体积损失。

我们在射击犰狳示例（图2）中测试了矩阵自由共轭梯度求解器，以及我们为固体耦合阶段设计的域分解求解器的性能，并在表3中展示了结果。矩阵自由共轭梯度求解器通过避免构建海森矩阵，显著提高了效率（快 20× ）并降低了内存开销。另一方面，我们的域分解求解器比直接对固体和接触系统进行分解快 40%。

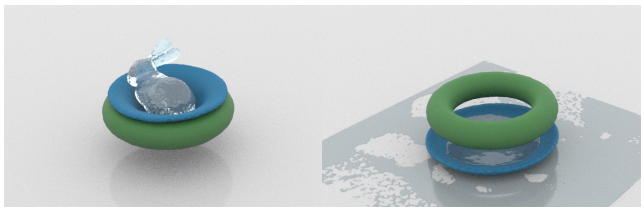
5.2 比较

在本节中，我们将所提方法与几种流行的光滑粒子流体动力学流体求解器，以及一种最先进的固液耦合方法 Elas-toMonolith [Takahashi 和 Batty 2022]进行比较。我们利用开源库 SPlisHSPlasH[1] 实现了 SPH 流体模拟器。为了将所提方法与 ElastoMonolith 进行比较，我们根据其论文中的设置搭建了两个场景，采用相同的参数，并为确保公平，在 Google Compute Engine 的“e2-standard-8”（8核、32GB RAM）实例上运行所有模拟。

[1]https://github.com/InteractiveComputerGraphics/SPlisHSPlasH



(a) 将一只液体兔子放入碗中。



(b) 将一只液体兔子和一个弹性碗放到静止圆环上。

图10. **液体兔子**。与 ElastoMonolith [Takahashi 和 Batty 2022], 相比，在完全相同的场景设置下，我们的方法在这两个示例中均实现了超过 5× 的加速。

5.2.1 流体动力学。尽管现有的大多数 SPH 流体求解器都专注于不可压缩流体，但我们的方法将流体视为弱可压缩流体，因此能够在统一框架中实现流体与可变形固体的耦合。我们进行了一项溃坝模拟，将所提方法与两种 SPH 流体求解器 IISPH [Ihmsen 等人 2013] 和 DFSPH [Bender 和 Koschier 2015]进行比较。这些方法通常采用粒子重采样 [Akinci 等人 2013, 2012] 或隐式表示 [Bender 等人 2019年；Koschier 和 Bender 2017] 来施加边界反作用力。相比之下，我们的方法采用 IPC [Li 等人2020] 实现更稳健的固液耦合，并保证无穿透。我们对所有方法统一施加相同的 CFL条件，并设置上限 5*ms*，同时使用体积映射 [Bender 等人 2019] 进行边界处理。如图9所示，尽管我们的方法没有严格保证不可压缩性，但能够产生自然的流体动力学效果，且视觉上观察不到体积损失。另一方面，由于边界处理策略更为复杂，我们的方法（0.45分钟/帧）比 IISPH（0.31分钟/帧）和 DFSPH（0.15分钟/帧）运行更慢。然而，我们提出的方法能够将 SPH流体与采用任意本构模型的弹性固体耦合，而现有的大多数 SPH 方法 [Kugelstadt 等人 2021年；Peer 等人 2018] 都将弹性固体视为不可压缩，这通常并不适用。

5.2.2 固液耦合。 随后，我们将我们的方法与 ElastoMonolith [Takahashi 和 Batty 2022], 进行比较。该方法以整体式方式耦合欧拉流体与拉格朗日固体。按照他们的实验设置，我们使用我们的方法，以相同的参数运行了两个固液耦合模拟（图10）。我们的方法在这两个场景中的耗时分别为 24.1 秒/帧和 62.8 秒/帧；根据他们报告的耗时（253.2 秒/帧和 352.0 秒 / 帧），这两者都比 ElastoMonolith 快 5× 倍以上。耦合欧拉流体与拉格朗日固体需要处理几何差异，而且通常

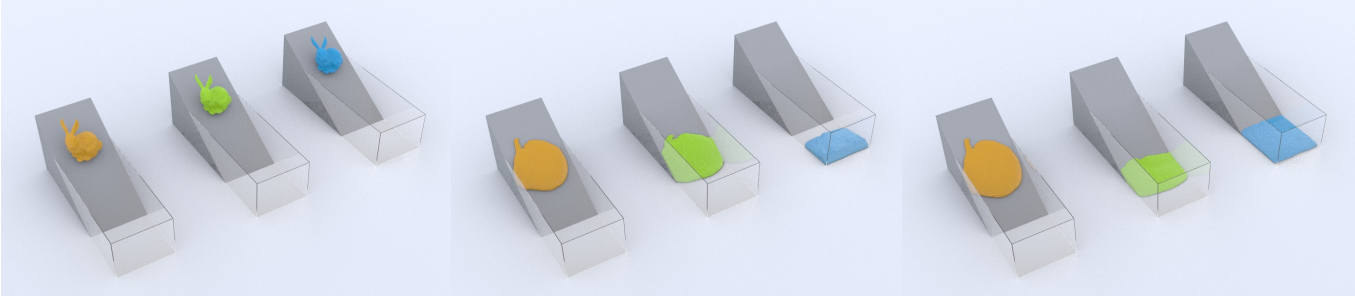


图11. **变化摩擦**。将三只黏性兔子从不同的斜坡滑下，其摩擦系数各不相同 μ（从左到右：0.5、0.03、0）。我们的方法支持调节固液边界摩擦。

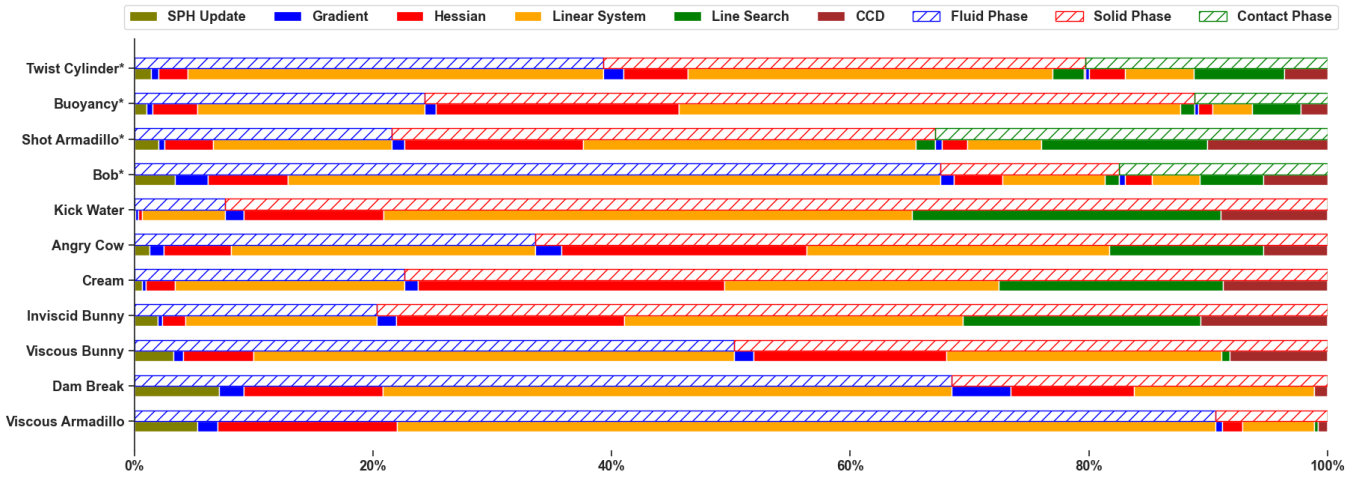


图12. **计时分解**。我们展示了不同模拟阶段的计时情况，并绘制了各主要程序例程所占的比例。标有 * 的示例使用我们的三阶段时间分裂方案进行模拟。其他示例则使用两阶段方案生成。具体而言，光滑粒子流体动力学更新（包括邻域搜索和密度更新）仅发生在流体阶段；非线性优化的线搜索发生在固体阶段和接触阶段；考虑 IPC 接触能时，则会统计连续碰撞检测（CCD）的耗时。

需要进行 SPD 重构，才能使线性系统易于求解。正如 ElastoMonolith 中所述，这种 SPD 重构可能会给系统引入许多额外的非零元，尤其是在接触丰富且固体形状复杂的情况下。相比之下，我们的方法从统一的拉格朗日视角处理固体和流体，以统一的方式处理固体间接触和固液接触。

5.3 复杂场景

随后，我们在更复杂的场景中评估了该方法的效率和鲁棒性。图4展示了该方法能够正确捕捉浮力行为。该方法也能自然地模拟黏性流体（图3），甚至支持调节边界摩擦（图11）。此外，该方法也能很好地支持与薄壳的双向耦合（图5和图1），并保证无穿透。在图13中，我们展示了该框架甚至可以模拟由任意余维数（0、1、2 和 3）的几何体构成的紧密耦合系统。详细的参数设置见附录。

6 结论

我们提出了一种适用于弱可压缩 SPH 流体和非线性弹性有限元固体的统一双向强耦合框架。为此，我们将固液相互作用建模为 SPH 粒子与有限元边界单元之间的接触力，并采用 IPC 来保证不可穿透性和稳定性。由于我们以更新拉格朗日的方式追踪 SPH 粒子的体积变化，不可压缩性能量始终保持为二次形式，同时还能维持良好的粒子分布。通过对离散黏性力进行对称近似，我们提出了一种适用于优化时间积分的黏度势。随后，我们提出了一种带有接触代理的时间分裂方案，以便在保持鲁棒性的同时高效求解时间积分优化问题。我们还通过无矩阵共轭梯度法和基于舒尔补的域分解求解器进一步提升了性能。

与现有将欧拉流体与拉格朗日弹性固体耦合的方法相比 [Takahashi 和 Batty 2022；Zarifi 和 Batty 2017],



图1 13. **愤怒的奶牛**。我们展示了该方法能够模拟任意余维数下材料之间的耦合，包括流体粒子、杆（橡皮筋）、薄壳（皮革垫）、可变形固体（奶牛）和刚体（立方体）。我们用弹弓发射一头愤怒的奶牛，奶牛穿墙而过，随后落入水中。各种材料之间的相互作用都得到了准确捕捉。

我们的方法以拉格朗日方式处理流体和固体，避免了处理不同空间离散的需要。在这种统一视角下，我们的方法能够实现更便捷、更稳健的双向耦合，即使是在流体与余维固体之间也是如此。同样，与现有将所有材料都视为 SPH 粒子的方法不同 [Kugelstadt 等人 2021；Peer 等人 2018] ，我们的公式兼具 SPH 流体的效率和 FEM 固体的精度。

未来有许多值得探索的研究方向。首先，当流体自由度占主导时，为每个流体粒子构建并查询空间哈希可能会带来相当大的开销。事实上，由于内部粒子不会发生固液接触，为了提高效率，我们可以仅在流体的扩展包围盒与各固体的交集区域内构建空间数据结构。此外，固体与流体之间的黏附也是一种值得建模的有趣现象。与屏障能类似，黏附力可以施加于彼此接近的固液基本体对上，但方向相反。通过解析流体的表面张力来模拟黏附，也是一个有趣的未来研究方向。

参考文献

Nadir Akinci、Jens Cornelis、Gizem Akinci 和 Matthias Teschner。2013年。将弹性固体与光滑粒子流体动力学流体耦合。*ComputerAnimation and Virtual Worlds* 24，3-4（2013年），195–203。Nadir Akinci、Markus Ihmsen、Gizem Akinci、Barbara Solenthaler 和 Matthias Teschner。2012年。用于不可压缩 SPH 的多功能刚体-流体耦合。*ACM Trans- actions on Graphics (TOG)* 31，4（2012年），1–8。Christopher Batty、Florence Bertails 和 Robert Bridson。2007年。用于精确固液耦合的快速变分框架。*ACM Transactions on Graphics (TOG)* 26，3（2007年），100–es。Christopher Batty、Andres Uribe、Basile Audoly 和 Eitan Grinspun。2012年。离散黏性薄片。*ACM Transactions on Graphics (TOG)* 31，4（2012年），1–7。Markus Becker、Markus Ihmsen 和 Matthias Teschner。2009a。用于可变形固体的共旋光滑粒子流体动力学。。载于 *NPH*。27–34。Markus Becker 和 Matthias Teschner。2007年。用于自由表面流动的弱可压缩光滑粒子流体动力学。载于 *2007*年 *ACM SIGGRAPH/Eurographics* 研讨会论文集

计算机动画。209–217。Markus Becker、Hendrik Tessendorf 和 Matthias Teschner。2009b。用于拉格朗日刚体-流体耦合的直接强制法。*IEEE Transactions on Visualization and Computer Graphics* 15, 3（2009），493–503。Jan Bender 和 Dan Koschier。2015。无散度光滑粒子流体动力学。载于*ACM SIGGRAPH/Eurographics*计算机动画研讨会论文集。147–155。Jan Bender 和 Dan Koschier。2016。用于不可压缩黏性流体的无散度 SPH。*IEEE Transactions on Visualization andComputer Graphics* 23, 3（2016），1193–1206。Jan Bender、Tassilo Kugelstadt、Marcel Weiler 和 Dan Koschier。2019。体积映射：SPH 的隐式边界表示。载于*Motion, Interaction and Games*。1–10。Javier Bonet 和 T-SL Lok。1999。光滑粒子流体动力学公式的变分与动量守恒方面。*ComputerMethods inapplied mechanics and engineering* 180, 1-2（1999），97–115。Christopher Brandt、Leonardo Scandolo、Elmar Eisemann 和 Klaus Hildebrandt。2019。用于实时流体-弹性固体相互作用与接触模拟的降阶浸入法。*ACM Transactions on Graphics (TOG)* 38, 6（2019），1–16。Robert Bridson。2015。面向计算机图形学的流体模拟。AK Peters/CRC Press。Yanqing Chen、Timothy A Davis、William W Hager 和 Sivasankaran Rajamanickam。2008。算法887：CHOLMOD，超节点稀疏乔列斯基分解及更新/降阶更新。*ACM Transactions onMathematicalSoftware(TOMS)* 35, 3（2008），1–14。Pascal Clausen、Martin Wicke、Jonathan R Shewchuk 和 James F O'Brien。2013。使用拉格朗日网格模拟液体及固液相互作用。*ACM Transac- tions on Graphics (TOG)* 32, 2（2013），1–15。Yu Fang、Ziyin Qu、Minchen Li、Xinxin Zhang、Yixin Zhu、Mridul Aanjaneya 和 Chenfanfu Jiang。2020。IQ-MPM：一种用于无黏附强双向耦合非线性固体与流体的界面求积物质点法。*ACM Transactions on Graphics (TOG)* 39, 4（2020），51–1。Ronald P Fedkiw。2002。使用幽灵流体法将欧拉流体计算与拉格朗日固体计算耦合。*J. Comput. Phys.* 175, 1（2002），200–224。Ronald P Fedkiw、Tariq Aslam、Barry Merriman 和 Stanley Osher。1999。一种处理多材料流中界面的非振荡欧拉方法（幽灵流体法）。*Journal of computational physics* 152, 2（1999），457–492。Yun Fei、Christopher Batty、Eitan Grinspun 和 Changxi Zheng。2018。用于模拟液体-织物相互作用的多尺度模型。*ACM Transactions on Graphics (TOG)* 37, 4（2018），1–16。Georg C Ganzenmüller。2015。一种用于拉格朗日光滑粒子流体动力学的沙漏控制算法。*Computer Methods i n Applied Mechanics and Engineering* 286（2015），87–106。

Ming Gao、Andre Pradhana、Xuchen Han、Qi Guo、Grant Kot、Eftychios Sifakis 和 Chenfanfu Jiang。2018。为含颗粒流模拟流体-沉积物混合物的动画。*ACM Transactions on Graphics (TOG)* 37, 4（2018），1–11。Christoph Gissler、Andreas Peer、Stefan Band、Jan Bender 和 Matthias Teschner。2019。用于强流体-刚体耦合的互联 SPH 压力求解器。*ACM Transactions on Graphics (TOG)* 38, 1（2019），1–13。Eran Guendelman、Andrew Selle、Frank Losasso 和 Ronald Fedkiw。2005。将水和烟雾与薄可变形壳体及刚性壳体耦合。*ACM Transactions on Graphics (TOG)* 24, 3（2005），973–981。David AB Hyde、Steve n W Gagniere、Alan Marquez-Razon 和 Joseph Teran。2020。一种适用于具有较大表面能液体的隐式更新拉格朗日公式。*ACM Transactions on Graphics (TOG)* 39, 6（2020），1–13。Markus Ihmsen、Nadir Akinci、Marc Gissler 和 Matthias Teschner。2010。PCISPH 的边界处理与自适应时间步进。*VRIPHYS'10*，79–88。Markus Ihmsen、Jens Cornelis、Barbara Solenthaler、Christopher Horvath 和 Matthias Teschner。2013。隐式不可压缩 SPH。*IEEE transactions on visualization and computer graphics* 20, 3（2013），426–435。Chenfanfu Jiang、Craig Schroeder、Joseph Teran、Alexey Stomakhin 和 Andrew Selle。2016。用于连续介质材料模拟的物质点法。载于*ACM SIGGRAPH 2016 Courses*。1–52。Bryan M Klingner、Bryan E Feldman、Nuttapong Chentanez 和 James F O'Brien。2006。使用动态网格进行流体动画。载于*ACM SIGGRAPH 2006 Papers*。820–825。Dan Koschier 和 Jan Bender。2017。用于改进 SPH 边界处理的密度图。载于*ACM SIGGRAPH/Eurographics* 计算机动画研讨会论文集。1–10。Dan Koschier、Jan Bender、Barbara Solenthaler 和 Matthias Teschner。2022。计算机图形学中的 SPH 方法综述。载于《计算机图形学论坛》，第 41 卷。Wiley在线图书馆，737–760。Tassilo Kugelstadt、Jan Bender、José Antonio Fernández-Fernández、Stefan Rhys Jeske、Fabian Löschner 和 Andreas Longva。2021。具有隐式零能模态控制的快速共旋弹性 SPH 固体。*ACM* 计算机图形学与交互技术论文集 4, 3（2021），1–21。David IW Levin、Joshua Litven、Garrett L Jones、Shinjiro Sueda 和 Dinesh K Pai。2011。带接触的欧拉固体模拟。*ACM Transactions on Graphics (TOG)* 30, 4（2011），1–10。Minchen Li、Zachary Ferguson、Teseo Schneider、Timothy R Langlois、Denis Zorin、Daniele Panozzo、Chenfanfu Jiang 和 Danny M Kaufman。2020。增量势接触：无相交、无反转的大变形动力学。*ACM Trans. Graph.* 39, 4（2020），49。Xuan Li、Yu Fang、Minchen Li 和 Chenfanfu Jiang。2022。BFEMP：采用屏障接触的无互穿 MPM–FEM 耦合。*Computer Methods in Applied Mechanics and Engineering* 390（2022），114350。Miles Macklin 和 Matthias Müller。2013。基于位置的流体。*ACM Transactions onGraphics (TOG)* 32, 4（2013），1–12。Joe J Monaghan。1992。光滑粒子流体动力学。*Annual review of astronomy and astrophysics* 30（1992），543–574。Joe J Monaghan。1994。使用 SPH 模拟自由表面流。*Journal of computational physics* 110, 2（1994），399–406。Joe J Monaghan。2005。光滑粒子流体动力学。*Reports on progress in physics* 68, 8（2005），1703。Jean Jacques Moreau。2011。关于单边约束、摩擦与塑性。载于《数学物理中的新变分技术》。施普林格，171–322。Matthias Müller、David Charypar 和 Markus H Gross。2003。面向交互式应用的基于粒子的流体模拟。。载于计算机动画研讨会，第 2 卷。Andreas Peer、Christoph Gissler、Stefan Band 和 Matthias Teschner。2018。一种用于不可压缩线弹性固体的隐式 SPH 公式。载于《计算机图形学论坛》，第 37 卷。Wiley在线图书馆，135–148。Andreas Peer、Markus Ihmsen、Jens Cornelis 和 Matthias Teschner。2015。一种适用于 SPH 流体的隐式黏度公式。*ACM Transactions on Graphics (TOG)* 34, 4（2015），1–10。Andreas Peer 和 Matthias Teschner。2016。为具有涡量扩散的高黏度 SPH 流体指定速度梯度。*IEEE transactions on visualization and computer graphics* 23, 12（2016），2656–2662。Charles S Peskin。2002。浸入边界法。《数值学报》11（2002），479–517。Avi Robinson-Mosher、Craig Schroeder 和 Ronald Fedkiw。2011。用于单体流固相互作用的对称正定公式。*J. Comput. Phys.* 230, 4 2011 1547 1566 ( ), – 。Avi Robinson-Mosher、Tamar Shinar、Jon Gretarsson、Jonathan Su 和 Ronald Fedkiw。2008。流体与刚性、可变形固体及壳体的双向耦合。*ACM Transactions on Graphics (TOG)* 27, 3（2008），1–9。Doug Roble、Nafees bin Zafar 和 Henrik Falt。2005。具有不规则边界体素的笛卡尔网格流体模拟。载于*ACM SIGGRAPH 2005 Sketches*。138–es。Eftychios Sifakis 和 Jernej Barbic。2012。三维可变形固体的有限元模拟：理论、离散化与模型降阶实用指南。载于*ACM SIGGRAPH 2012 courses*。1–50。

Barbara Solenthaler 和 Renato Pajarola。2009年。预测-校正不可压缩SPH。载于 *ACM SIGGRAPH 2009*论文集。1–6。Barbara Solenthaler、Jürg Schläfli 和 Renato Pajarola。2007年。流体-固体相互作用的统一粒子模型。《计算机动画与虚拟世界》18，1（2007年），69–82。Alexey Stomakhin、Russell Howes、Craig A Schroeder 和 Joseph M Teran。2012年。能量一致的可逆弹性。载于 计算机动画研讨会，第1卷。Alexey Stomakhin、Craig Schroeder、Chenfanfu Jiang、Lawrence Chai、Joseph Teran 和 Andrew Selle。2014年。用于相变和多种材料的增强型MPM。*ACM Transactions on Graphics*（*TOG*）33，4（2014年），1–11。Deborah Sulsky、Shi-Jian Zhou 和 Howard L Schreyer。1995年。粒子-网格法在固体力学中的应用。《计算机物理通讯》87，1-2（1995年），236–252。Tetsuya Takahashi 和 Christopher Batty。2020年。Monolith：用于强双向刚体-刚体、刚体-流体耦合的单体压力-黏度-接触求解器。（2020年）。Tetsuya Takahashi 和 Christopher Batty。2021年。FrictionalMonolith：一种基于单体优化的方法，用于处理具有接触感知刚体耦合的颗粒流。*ACM Transactions on Graphics*（*TOG*）40，6（2021年），1–20。Tetsuya Takahashi 和 Christopher Batty。2022年。ElastoMonolith：用于接触感知弹性固体耦合的单体基于优化的液体求解器。*ACM Transactions on Graphics*（*TOG*）41，6（2022年），1–19。Tetsuya Takahashi、Yoshinori Dobashi、Issei Fujishiro、Tomoyuki Nishita 和 Ming C Lin。2015年。基于SPH的黏性流体的隐式公式。载于 《计算机图形学论坛》，第34卷。Wiley在线图书馆，493–502。Andre Pradhana Tampubolon、Theodore Gast、Gergely Klár、Chuyuan Fu、Joseph Teran、Chenfanfu Jiang 和 Ken Museth。2017年。多物种多孔砂水混合物模拟。*ACM Transactions on Graphics*（*TOG*）36，4（2017年），1–11。Yun Teng、David IW Levin 和 Theodore Kim。2016年。欧拉固体-流体耦合。*ACM Transactions on Graphics*（*TOG*）35，6（2016年），1–8。Boris Valkov、Chris H Rycroft 和 Ken Kamrin。2015年。流体中软固体多相相互作用的欧拉方法。《应用力学杂志》82，4（2015年），041011。Hui Wang、Yongxu Jin、Anqi Luo、Xubo Yang 和 Bo Zhu。2020年。使用移动最小二乘粒子的余维表面张力流。*ACM Transactions on Graphics*（*TOG*）39，4（2020年），42–1。Marcel Weiler、Dan Koschier、Magnus Brand 和 Jan Bender。2018年。用于SPH流体的物理一致隐式黏度求解器。载于 《计算机图形学论坛》，第37卷。Wiley在线图书馆，145–155。Xiao Yan、C-F Li、X-S Chen 和 S-M Hu。2018年。相互作用的流体与固体的MPM模拟。载于 《计算机图形学论坛》，第37卷。Wiley在线图书馆，183–193。Omar Zarifi 和 Christopher Batty。2017年。用于流体与可变形物体之间强双向耦合的正定切割单元法。载于 *ACM SIGGRAPH/Eurographics*计算机动画研讨会论文集。1–11。Fuzhen Zhang。2006年。舒尔补及其应用。第4卷。Springer Science & Business Media。

流体势能的导数

不可压缩势能关于流体粒子位置的梯度和海森矩阵为

$$\begin{aligned}\frac{\partial P_I(\mathbf{x})}{\partial \mathbf{x}} &= \sum_i k_I V_0 (J_i - 1) J_i^n h \frac{\partial (\nabla \cdot \mathbf{v}_i^{n+1})}{\partial \mathbf{x}}, \\ \frac{\partial^2 P_I(\mathbf{x})}{\partial \mathbf{x}^2} &= \sum_i k_I V_0 (J_i^n)^2 h^2 \frac{\partial (\nabla \cdot \mathbf{v}_i^{n+1})}{\partial \mathbf{x}} \left(\frac{\partial (\nabla \cdot \mathbf{v}_i^{n+1})}{\partial \mathbf{x}} \right)^T .\end{aligned} \tag{30}$$

这个常数海森矩阵显然是半正定的（PSD），因为它只是若干向量与正系数相乘后所得外积之和。

同理，黏度势关于流体粒子位置的梯度和海森矩阵为

$$\frac{\partial P_V(\mathbf{x})}{\partial \mathbf{x}_i} = \nu \sum_j \mathbf{V}_{ij} \mathbf{v}_{ij}^{n+1}, \quad \frac{\partial^2 P_V(\mathbf{x})}{\partial \mathbf{x}_i \partial \mathbf{x}_j} = \begin{cases} \frac{\nu}{h} \sum_k \mathbf{V}_{ik}, j = i \\ -\frac{\nu}{h} \mathbf{V}_{ij}, j \neq i \end{cases} . \tag{31}$$

由于$\mathbf{V}_{ij} = -4(d+2)\frac{m_i m_j}{\rho_i+\rho_j}\frac{\nabla_i W_{ij}}{\|\mathbf{x}_{ij}^n\|^2+0.01}\mathbf{x}_{ij}^n(\mathbf{x}_{ij}^n)^T$对于时间步内任意粒子对都是一个 3×3常数半正定矩阵，因此黏度势的海森矩阵也是一个常数半正定矩阵。

B 时间分裂

B.1 基线时间分裂

应用时间分裂，我们可以将原始时间积分分解为**流体阶段**

$$\begin{cases}\mathbf{v}_f^{n+1/2} = \mathbf{v}_f^n + h\mathbf{M}_f^{-1}(-\nabla_f P([(\mathbf{x}_f^{n+1/2})^T,(\mathbf{x}_s^n)^T]^T) + \mathbf{f}_f)\\ \mathbf{x}_f^{n+1/2} = \mathbf{x}_f^n + h\mathbf{v}_f^{n+1/2}\end{cases} \tag{32}$$

和一个**固体耦合阶段**

$$\begin{cases}\mathbf{v}^{n+1} = \begin{bmatrix}\mathbf{v}_f^{n+1/2}\\ \mathbf{v}_s^n\end{bmatrix} + h\mathbf{M}^{-1}(-\nabla\Psi(\mathbf{x}^{n+1}) - \nabla C(\mathbf{x}^{n+1}) + \begin{bmatrix}\mathbf{0}\\ \mathbf{f}_s\end{bmatrix})\\ \mathbf{x}^{n+1} = \mathbf{x}^n + h\mathbf{v}^{n+1}\end{cases}, \tag{33}$$

其中$\mathbf{f}f$和$\mathbf{f}_s$分别是作用于流体和固体的外力。该基线时间分裂方案的两个阶段具有等价的优化形式

$$\boxed{\begin{aligned}\mathbf{x}_f^{n+1/2} &= arg\min_{\mathbf{x}_f}\frac{1}{2}\|\mathbf{x}_f - \hat{\mathbf{x}}_f^n\|^2_{\mathbf{M}_f} + h^2P([\mathbf{x}_f^T,(\mathbf{x}_s^n)^T]^T)\\ \mathbf{x}^{n+1} &= arg\min_{\mathbf{x}}\frac{1}{2}\|\mathbf{x} - \hat{\mathbf{x}}^{n+1/2}\|^2_{\mathbf{M}} + h^2(\Psi(\mathbf{x}) + C(\mathbf{x}))\end{aligned}} \tag{34}$$

其中 $\hat{}\mathbf{x}_f^n = \mathbf{x}_f^n + h\mathbf{v}_f^n + h^2\mathbf{M}_f^{-1}\mathbf{f}f$和 $\hat{}\mathbf{x}^{n+1/2} = \mathbf{x}^n + h[(\mathbf{v}_f^{n+1\ 2}/)^T,(\mathbf{v}_s^n)^T]^T + h^2\mathbf{M}^{-1}[\mathbf{0}^T,\mathbf{f}_s^T]^T$，其中 $\mathbf{v}_f^{n+1\ 2} = /(\mathbf{x}_f^{n+1\ 2} - /\mathbf{x}_f^n)/h$。优化算法的详情见算法 2。

Algorithm 2 Baseline Time Splitting

$\mathbf{x}\leftarrow\mathbf{x}^n, \hat{\mathbf{x}}_f^n\leftarrow\mathbf{x}_f^n + h\mathbf{v}_f^n + h^2\mathbf{M}_f^{-1}\mathbf{f}_f$
SPH Neighbor Search & Density Update
// Fluid Phase
$\mathbf{H}_f\leftarrow h^2\nabla_f^2P([\mathbf{x}_f^T,(\mathbf{x}_s^n)^T]) + \mathbf{M}_f$
$\mathbf{p}_f\leftarrow -\mathbf{H}_f^{-1}\left(h^2\nabla_f P([\mathbf{x}_f^T,(\mathbf{x}_s^n)^T]^T) + \mathbf{M}_f(\mathbf{x}_f - \hat{\mathbf{x}}_f^n)\right)$
$\mathbf{x}_f\leftarrow\mathbf{x}_f + \mathbf{p}_f$
$\mathbf{x}_f^{n+1/2}\leftarrow\mathbf{x}_f, \mathbf{v}_f^{n+1/2} = (\mathbf{x}_f - \mathbf{x}_f^n)/h$
// Solid Coupling Phase
$\hat{\mathbf{x}}^{n+1/2}\leftarrow\mathbf{x}^n + h[(\mathbf{v}_f^{n+1/2})^T,(\mathbf{v}_s^n)^T]^T + h^2\mathbf{M}^{-1}[\mathbf{0}^T,\mathbf{f}_s^T]^T$
do
  $\mathbf{H}\leftarrow h^2\left(\nabla^2\Psi(\mathbf{x}) + \nabla^2C(\mathbf{x})\right) + \mathbf{M}$
  $\mathbf{p}\leftarrow -\mathbf{H}^{-1}\left(h^2(\nabla\Psi(\mathbf{x}) + \nabla C(\mathbf{x})) + \mathbf{M}(\mathbf{x} - \hat{\mathbf{x}}^{n+1/2})\right)$
  $\alpha\leftarrow$ Backtracking Line Search with CCD
  $\mathbf{x}\leftarrow\mathbf{x} + \alpha\mathbf{p}$
while $\frac{1}{h}\|\mathbf{p}\| > \epsilon$
$\mathbf{x}^{n+1}\leftarrow\mathbf{x}, \mathbf{v}^{n+1}\leftarrow(\mathbf{x} - \mathbf{x}^n)/h$
return $\mathbf{x}^{n+1}, \mathbf{v}^{n+1}$

B.2 误差分析

隐式欧拉法和我们采用代理增强的时间分裂方案的位置更新可分别表示为

$$\begin{aligned}\mathbf{x}_{\mathrm{IE}}^{n+1} &= \mathbf{x}^n + h\mathbf{v}^n + h^2\mathbf{M}^{-1}\Big(-\nabla P(\mathbf{x}_{\mathrm{IE}}^{n+1})\\ &\quad - \nabla\Psi(\mathbf{x}_{\mathrm{IE}}^{n+1}) - \nabla C_{\mathrm{sf}}(\mathbf{x}_{\mathrm{IE}}^{n+1}) - \nabla C_{\mathrm{ss}}(\mathbf{x}_{\mathrm{IE}}^{n+1}) + \mathbf{f}_{\mathrm{ext}}\Big)\\ \mathbf{x}^{n+1} &= \mathbf{x}^n + h\mathbf{v}^n + h^2\mathbf{M}^{-1}\Big(-\nabla P(\mathbf{x}^{n+1/2}) - \nabla\Psi(\mathbf{x}^{n+1})\\ &\quad - \nabla\hat{C}_{\mathrm{sf}}(\mathbf{x}^{n+1/2}) - \frac{1}{2}\nabla C_{\mathrm{sf}}(\mathbf{x}^{n+1}) - \nabla C_{\mathrm{ss}}(\mathbf{x}^{n+1}) + \mathbf{f}_{\mathrm{ext}}\Big).\end{aligned} \tag{35}$$

如果我们定义

$$\begin{aligned}e(\mathbf{x}) = \Big\|\mathbf{x}^n + h\mathbf{v}^n + h^2\mathbf{M}^{-1}\Big(&-\nabla P(\mathbf{x}) - \nabla\Psi(\mathbf{x})\\ &- \nabla C_{\mathrm{sf}}(\mathbf{x}) - \nabla C_{\mathrm{ss}}(\mathbf{x}) + \mathbf{f}_{\mathrm{ext}}\Big) - \mathbf{x}\Big\|,\end{aligned} \tag{36}$$

$\mathbf{x}^{n+1}$由隐式欧拉法给出的 IE 满足 $e(\mathbf{x}^{n+1}\mathrm{IE}) = 0$，而对于来自我们方案的 $\mathbf{x}^{n+1}$，有 $e(\mathbf{x}^{n+1}) = \mathrm{O}(h^4)$。具体而言，

$$\begin{aligned}e(\mathbf{x}^{n+1}) &= h^2\Big\|\mathbf{M}^{-1}(\nabla P(\mathbf{x}^{n+1/2}) - \nabla P(\mathbf{x}^{n+1})\\ &\quad + \nabla\hat{C}_{\mathrm{sf}}(\mathbf{x}^{n+1/2}) - \frac{1}{2}\nabla C_{\mathrm{sf}}(\mathbf{x}^{n+1}))\Big\|\\ &= O\Bigg(h^3\Big\|\mathbf{M}^{-1}\Big(\nabla^2P(\mathbf{x}^n)(\mathbf{v}^{n+1/2} - \mathbf{v}^{n+1}) +\\ &\quad \frac{1}{2}\nabla^2C_{\mathrm{sf}}(\mathbf{x}^n)(\mathbf{v}^{n+1/2} - \mathbf{v}^{n+1})\Big)\Big\|\Bigg)\\ &= O\Bigg(h^4\Big\|\mathbf{M}^{-1}\Big(\nabla^2P(\mathbf{x}^n) + \frac{1}{2}\nabla^2C_{\mathrm{sf}}(\mathbf{x}^n)\Big)\mathbf{M}^{-1}\Big(\nabla\Psi(\mathbf{x}^n)\\ &\quad + \frac{1}{2}\nabla C_{\mathrm{sf}}(\mathbf{x}^n) + \nabla C_{\mathrm{ss}}(\mathbf{x}^n)\Big)\Big\|\Bigg)\\ &= O(h^4).\end{aligned} \tag{37}$$

这里假设，在我们的离散化域中，任意一对基本体（粒子-粒子对、粒子-三角形对和三角形-三角形对）之间的距离都有一个下界 ϵ。因此，$\nabla^2P(\mathbf{x}^n)$、$\nabla^2C\mathrm{sf}(\mathbf{x}^n)$、$\nabla\Psi(\mathbf{x}^n)$、$\nabla C\mathrm{sf}(\mathbf{x}^n)$ 和 $\nabla C\mathrm{ss}(\mathbf{x}^n)$ 均有界，这表明与隐式欧拉解相比，我们的方法存在 $\mathrm{O}(h^4)$ 的差异。由于隐式欧拉法相对于偏微分方程解的误差为 $\mathrm{O}(h^2)$，因此，当所提出的时间分裂方案稳定时，其精度阶与隐式欧拉法相同。

C 复杂场景实验

本节介绍我们在各种复杂场景下进行模拟的实验设置，并简要讨论实验结果。

浮力。 我们将三只密度不同的弹性大象放入水中（$1000\ kg/m^3$）。浅灰色的大象（$200\ kg/m^3$）浮在水面上；蓝色的大象（$700\ kg/m^3$）约有一半浸在水中；红色的大象（$1200\ kg/m^3$）则沉入底部。这表明我们的方法能够正确捕捉浮力行为。

变化摩擦。 我们将三只黏性兔子放到斜坡上，它们的摩擦系数各不相同（橙色兔子：0.5，绿色兔子：0.03，蓝色兔子：0）。三只兔子的动力黏度系数均为 $100\ kg/m^3$，斜坡角度为 30◦。

扭转圆柱。 由于稍有不慎就容易发生穿透，因此将流体与薄壳耦合颇具挑战性。正如 [Zarifi 和 Batty 2017],所指出的，如果固体的厚度小于网格单元尺寸，欧拉流体可能会穿过固体。相比之下，我们的方法采用统一的拉格朗日视角，并通过 IPC 保证无穿透。在此示例中，我们模拟一个装满水的圆柱发生扭转的过程。该圆柱建模为厚度为 $2mm$ 的薄壳，其前后两侧分别有一个孔洞。圆柱的左侧和右侧以 $72◦/s$ 的角度旋转，并以 $2cm/s$ 的速度缓慢相向移动。随着圆柱扭转，水从孔洞中被挤出。该模拟展示了我们的方法能够生成稳定的模拟结果，并保证无穿透。

奶油。 此示例展示了黏性流体与弹性固体之间的耦合行为。我们用一把弹性勺子在瓷碗中搅拌奶油。勺柄绕 y 轴旋转，$360◦/s\ (0.2m/s)$，而瓷碗固定在桌面上。正如模拟结果所示，搅拌时，弹性勺子受到黏性奶油施加的阻力，因此发生了变形。

愤怒的奶牛。 接下来，我们展示该框架能够模拟任意余维数（0、1、2 和 3）几何体的自然物理行为及其相互作用。在此场景中，余维数分别为 0、1、2 的对象指流体粒子、橡皮筋和皮革垫。一头可变形奶牛被弹弓发射出去，撞上由刚性立方体组成的墙，随后落入水池中，产生了有趣的物理行为。刚性立方体和奶牛的密度分别为 $100kg/m^3$ 和 $700kg/m^3$。

踢水。在这个示例中，我们展示了这样一个场景：一个穿着多层裙子的人体模型在大型水池中踢水，涉及流体粒子与衣物之间的复杂交互。当人体模型在水中移动时，我们的方法能够自然地模拟裙子与水接触所产生的形变；当它高速踢出水面时，也能准确捕捉由此产生的水花。我们的方法能够在保证无穿透的前提下，有效处理流体粒子、轻薄衣物与快速移动的复杂边界之间的接触。