

Quality-Constrained Predictive Anisotropic Grid Warping for FLIP Liquids

Jong-Hyun Kim

jonghyunkim@inha.ac.kr

Inha University

Research Article

Keywords:

Posted Date: October 29th, 2026

DOI: <https://doi.org/10.21203/rs.3.rs-11219760/v1>

License:   This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Additional Declarations: No competing interests reported.

Quality-Constrained Predictive Anisotropic Grid Warping for FLIP Liquids

Jong-Hyun Kim

Received: date / Accepted: date

Abstract We propose a *quality-constrained predictive anisotropic grid warping* method for FLIP-based liquid simulation. Dynamic grid warping can concentrate resolution near important flow regions, but strong deformation may introduce local compression, shear, and poor cell aspect ratios. Our objective is to preserve substantial adaptive grid motion while explicitly controlling local cell quality. The method predicts near-future particle motion and estimates local velocity gradients to construct a directional anisotropic metric. A determinant-normalized metric redistributes resolution according to predicted deformation without uniformly shrinking the grid. During warping, Jacobian-based regularization and acceptance constraints on the determinant and condition number limit local volume and shape distortion while permitting coherent large-scale grid motion. A metric-change-guided tracking strategy reduces repeated optimization when the target metric varies slowly. We compare the method with Uniform FLIP, conventional scalar dynamic warping, and reactive anisotropic warping without future prediction. Across free-surface deformation, impact, thin-sheet, rotational, and repeated-wall-interaction scenes, the proposed method reduces local grid distortion and post-projection divergence while maintaining comparable or greater warp displacement than conventional warping. The results indicate that predictive directional adaptation can be combined with explicit local quality control in a structured warped-grid FLIP solver.

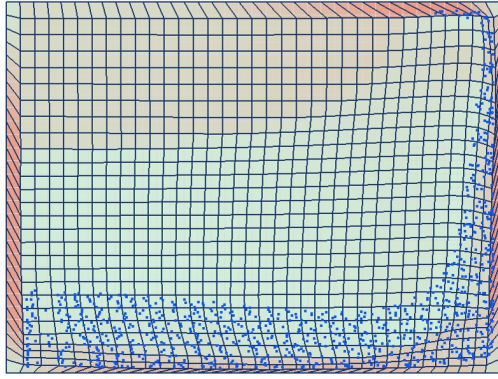
Keywords FLIP liquid simulation · Adaptive grid · Grid warping · Anisotropic deformation · Predictive deformation · Distortion control · Jacobian quality · Free-surface flow

1 Introduction

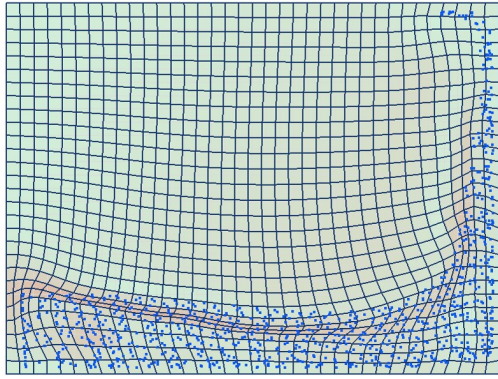
FLIP (Fluid-Implicit-Particle)-based liquid simulation is a representative hybrid fluid simulation technique that combines the Lagrangian characteristics of particles with the Eulerian computational structure of a grid, enabling stable representation of complex free-surface motion and large-scale flow. In FLIP, particles are used to track liquid mass and momentum, while major operations such as pressure projection are performed on the computational grid. Although this structure offers many advantages in terms of numerical stability and efficiency, the use of a fixed uniform grid has the fundamental limitation that the same spatial resolution must be allocated throughout the entire computational domain. In particular, important flow features may not be sufficiently represented under limited grid resolution in regions involving thin liquid sheets, rapidly moving free surfaces, collision-induced jets, or strong localized deformation.

To alleviate this limitation, dynamic grid-warping methods move grid nodes toward regions identified as important by quantities such as free-surface distance or flow velocity. Such methods improve the use of a fixed cell budget, but scalar importance alone does not indicate how the local flow is expected to stretch or compress. The mismatch becomes more pronounced in rapidly deforming sheets, jets, collisions, and rotational flows, where the required resolution can be strongly directional.

Jong-Hyun Kim
College of AI and Convergence (Dept. of Artificial Intelligence, Design Technology), Graduate School of Electrical and Computer Engineering at Inha University
Tel.: +82-32-860-9204
Fax: 0504-434-8210
E-mail: jonghyunkim@inha.ac.kr



(a) Previous method [15]



(b) Our method

Fig. 1: Representative 2D comparison of dynamically warped grids. (a) Our implementation of the dynamic grid-warping method of Ibayashi et al. [15]. (b) Our quality-constrained predictive anisotropic warping. Unlike scalar spatial adaptation, our method redistributes resolution according to predicted deformation directions while maintaining local grid quality.

Fig. 1 illustrates this distinction. The conventional dynamic warping in Fig. 1(a) concentrates nodes according to spatial importance, whereas our method in Fig. 1(b) uses predicted local deformation to orient the adaptation. The resulting anisotropic metric redistributes grid spacing toward directions expected to require finer resolution.

Directional adaptation, however, can degrade cell quality when applied too aggressively. Local compression, shear, or cell inversion can in turn affect particle-grid transfer and pressure projection. We therefore formulate adaptation strength and local cell quality as separate concerns: the grid is allowed to undergo coherent large-scale motion, while the Jacobian of each cell is constrained to avoid locally undesirable deformation.

Our method addresses these issues with a predictive directional metric, local Jacobian-based quality control,

and a lightweight temporal tracking strategy for slowly varying metric states.

The main contributions of this study are as follows.

- A predictive anisotropic metric that combines near-future free-surface importance with local deformation direction and redistributes spatial resolution using determinant normalization.
- A quality-constrained warping scheme that combines Jacobian-based regularization with determinant, condition-number, and cell-volume acceptance tests without directly bounding accumulated grid displacement.
- A metric-change-guided temporal tracking strategy that reduces repeated grid optimization when the target metric varies slowly, with little change in measured grid quality.

2 Problem Statement

In this study, we consider the dynamic grid-warping problem of reallocating a limited number of computational cells to regions that are important to the flow in FLIP liquid simulation. Let $\xi \in \Omega_\xi$ denote the reference coordinates of the regular grid, and define the corresponding warped-grid position in physical space at time t as

$$\mathbf{x} = \mathcal{W}(\xi, t). \quad (1)$$

For a uniform grid, $\mathcal{W}$ corresponds to the identity mapping, whereas in an adaptive grid, internal grid nodes move according to the flow state.

Conventional scalar-based grid adaptation can concentrate the grid in specific regions using local importance measures such as the free surface or velocity. However, a single scalar value cannot distinguish stretching and compression along different directions of the fluid. In particular, for thin liquid sheets, jets, collisions, and rotational flows, the required spatial resolution can vary substantially with direction. Therefore, in this study, we represent the target grid resolution by a directional metric $\mathbf{M}(\xi, t)$ rather than by a single scalar value and aim to construct predictive anisotropic warping that accounts for both the current flow and near-future deformation. The detailed construction of $\mathbf{M}$ and the prediction of future flow deformation are described in Sec. 4.

However, strong anisotropic warping can cause cell compression, shear, or inversion as grid-node displacement increases. The Jacobian representing the local deformation of each cell is defined as

$$\mathbf{J} = \frac{\partial \mathbf{x}}{\partial \xi} \quad (2)$$

Because ξ parameterizes the undeformed regular grid, a uniform grid satisfies $\mathbf{J} = \mathbf{I}$ and $\det(\mathbf{J}) = 1$. The quantity $\det(\mathbf{J})$ represents local volume variation, whereas $\kappa(\mathbf{J}) = \sigma_{\max}(\mathbf{J})/\sigma_{\min}(\mathbf{J})$ represents the degree of cell-shape distortion. Therefore, a small determinant or a high condition number indicates the formation of an unstable grid element.

An important distinction is that *the magnitude of grid warping and grid distortion are different concepts*. Let the regular grid spacings along each axis be h_x , h_y , and h_z , and define a representative grid spacing as $h = \min(h_x, h_y, h_z)$. If the regular position of node i is $\mathbf{x}_i^0$, the warp displacement is measured as

$$d_i = \frac{\|\mathbf{x}_i - \mathbf{x}_i^0\|}{h} \quad (3)$$

A large d_i indicates that the grid has moved far from its regular position, but it does not necessarily imply high cell distortion if neighboring nodes move coherently. Therefore, simply limiting d_i to keep the grid close to its regular position is not the objective of this study. Such an approach may reduce distortion, but it can also weaken adaptive warping itself and reduce its distinction from a uniform grid.

The problem addressed in this study is to compute a warped grid that allows sufficient grid deformation required by the flow while simultaneously satisfying the following local quality conditions:

$$J_{\min} \leq \det(\mathbf{J}) \leq J_{\max}, \quad \kappa(\mathbf{J}) \leq \kappa_{\max}. \quad (4)$$

Here, $J_{\min}$ and $J_{\max}$ define the allowable range of local volume variation, while $\kappa_{\max}$ limits the allowable cell-shape distortion.

The objective is therefore to satisfy Eq. (4) while retaining enough grid motion to represent the spatial and directional variation of the flow. Sec. 4 describes the predictive metric, quality-constrained warping, and temporal tracking used to achieve this objective.

3 Related Work

3.1 FLIP and Particle–Grid Fluid Simulation

Particle-in-Cell (PIC) methods combine material advection using Lagrangian particles with physical computation on an Eulerian grid, thereby exploiting the advantages of both representations. Brackbill and Ruppel [1] proposed the FLIP (Fluid-Implicit-Particle) method by combining particles with an adaptive grid, and FLIP has since been widely used for liquid simulation in computer graphics because of its low numerical dissipation and ability to represent complex free-surface motion. Foster and Fedkiw [2] presented an Eulerian-based

framework for three-dimensional liquid animation with free surfaces, providing an important foundation for subsequent grid-based liquid simulation research.

Research has also continued to improve numerical dissipation and instability arising from information transfer between particles and grids. Jiang et al. [12] proposed APIC, which assigns a local affine velocity to each particle to reduce information loss during particle–grid transfer while preserving the stability of PIC. Fu et al. [14] generalized this approach to a polynomial velocity representation and proposed PolyPIC, which preserves higher-order local motion information on particles. More recently, Qu et al. [22] proposed Power PIC, which uses transfer weights based on power diagrams to achieve stable particle–grid transfer even for nonuniform particle distributions, while Sancho et al. [23] combined an impulse formulation with the particle-in-cell framework to improve the preservation of rotational and vortical structures.

These studies primarily focus on improving the accuracy of particle–grid transfer, momentum conservation, and numerical dissipation. In contrast, while retaining the basic FLIP particle–grid structure, our study focuses on adaptively controlling the spatial placement and deformation of the computational grid itself according to the flow state.

3.2 Spatially Adaptive Grids for Liquid Simulation

Uniform Cartesian grids are efficient for fluid simulation because of their simple data structures and regular memory-access patterns, but they have the limitation that the same resolution must be allocated throughout the entire computational domain. To alleviate this limitation, various adaptive spatial discretizations have been studied. Losasso et al. [3] used an octree to selectively place high resolution near the free surface and in detailed flow regions. Klingner et al. [4] concentrated computational elements near fluid boundaries and regions of interest using a tetrahedral mesh that is dynamically reconstructed at each time step, while Chentanez et al. [5] used a lattice-based tetrahedral mesh to place fine elements near the free surface and relatively large elements in the interior.

Structural adaptivity for large-scale liquid simulation has also been investigated. Chentanez and Müller [6] reduced the degrees of freedom required for pressure computation by maintaining regular cells near the liquid surface while representing deep interior regions with tall cells. Ando et al. [7] combined tetrahedral discretization with spatially adaptive FLIP to simulate liquids with strongly varying spatial resolution. Zhu et al. [8] proposed a grid structure that can dynamically extend

the computational domain while maintaining high resolution in regions of interest, and English et al. [9] achieved dynamic spatial adaptivity using Chimera grids that overlap and move Cartesian grids of different resolutions.

Research on octree-based methods has likewise continued to improve computational efficiency and free-surface accuracy. Ferstl et al. [10] combined an adaptive hexahedral grid with finite-element discretization to reduce computation in the interior of large-scale liquids. Setaluri et al. [11] proposed SPGrid to improve memory-access efficiency for sparse Cartesian grids, and Aanjaneya et al. [13] combined it with power-diagram-based pressure discretization to realize adaptive liquid simulation in which resolution changes extend to the free surface. Ando and Batty [17] proposed a practical adaptive liquid solver that robustly handles free-surface pressure around octree T-junctions.

More recently, methods have been proposed to obtain adaptivity while preserving as much as possible the computational advantages of regular-grid structures. Xiao et al. [18] proposed an adaptive staggered-tilted grid that selectively activates tilted cells on a background Cartesian grid, combining a regular data layout with spatial adaptivity. Nakanishi et al. [20] proposed an adaptive PIC solver based on RBF-FD for pressure projection and particle-grid interpolation on dynamically changing quadtree/octree structures. Zhai et al. [16] also implemented adaptive staggered power particles on GPUs, combining spatially varying sampling with parallel computation.

3.3 Dynamic Grid Adaptation and Anisotropic Representation

Unlike methods that directly modify grid topology, approaches that move grid positions while preserving the existing regular-grid connectivity can retain regularity in both data structures and memory access. Most closely related to our work, Ibayashi et al. [15] proposed a dynamically warping grid in which the nodes of a regular Cartesian grid move over time. Their method defines grid deformation using a scalar potential and concentrates spatial resolution in required regions based on free-surface distance and flow information, thereby providing spatial adaptivity without explicitly reconstructing an unstructured adaptive mesh.

However, local fluid deformation generally exhibits different characteristics depending on direction. For example, in a thin sheet or jet, strong stretching may occur along one direction while relatively little change occurs along another. To represent such directionality, Parreiras et al. [21] proposed anisotropic particle-

in-cell simulation by combining a positive-definite tensor field with FLIP advection and projection. Whereas their study uses a tensor field to introduce directionality into the fluid motion itself, the anisotropy in our work is not intended to modify the physical properties of the fluid, but rather to determine *the directional allocation of computational grid resolution*.

In addition to spatial adaptivity, studies have also considered controlling computational cost in the temporal domain. Koike et al. [19] proposed an Eulerian liquid simulation using asynchronous time steps in different regions, thereby reducing computational cost according to spatially varying flow speeds. These studies demonstrate that the local flow state can be used to allocate computational resources, but they do not predict future deformation or explicitly incorporate such prediction into directional grid adaptation.

3.4 Recent Adaptive Liquid Discretizations and Our Position

Adaptive-grid research aimed at concentrating computational degrees of freedom near liquid surfaces has continued in recent years. Narita et al. [24] extended the conventional tall-cell structure with horizontal quadtree subdivision and proposed Quadtree Tall Cells, which more flexibly controls spatial resolution near the free surface. Subsequently, Narita and Kanai [25] proposed Adaptive Optical Layers, in which the thickness of the optical layer near the liquid surface varies spatially according to liquid motion, thereby dynamically reducing the number of cells used for pressure projection. These recent studies indicate that concentrating computational resources in important regions remains an important problem in liquid simulation.

Our method retains the regular-grid topology of dynamic warping [15], but extends scalar spatial adaptation with predicted deformation direction and explicit local quality constraints. Temporal metric coherence is used only to reduce repeated optimization and does not alter the underlying FLIP discretization.

4 Algorithm

In this section, we describe the proposed *quality-constrained predictive anisotropic grid warping* and its integration with FLIP liquid simulation. The proposed method preserves the connectivity of a regular Cartesian grid while deforming only the positions of internal nodes over time. Therefore, no separate remeshing or topology update is required, and each warped cell retains its original structured-grid index.

Fig. 2 summarizes the pipeline. The current FLIP state is used to estimate a predictive anisotropic metric; the grid is then warped under local quality constraints and passed to the FLIP transfer, projection, and advection stages.

At time step n , the particle position and velocity are denoted by $\mathbf{x}_p^n$ and $\mathbf{v}_p^n$, respectively, the position of the i th node in the regular grid by $\mathbf{x}_i^0$, and its current warped position by $\mathbf{x}_i^n$. For a computational domain of size $W \times H \times D$ and grid resolution $N_x \times N_y \times N_z$, the regular grid spacings defined above are computed as $h_x = W/N_x$, $h_y = H/N_y$, and $h_z = D/N_z$. At each time step, the proposed method proceeds in the following order: (1) prediction of the future free-surface position, (2) construction of a deformation-aware anisotropic metric, (3) quality-constrained grid warping, (4) update of warped cell geometry and particle references, (5) FLIP particle-to-grid transfer and body-force/boundary update, (6) pressure projection, and (7) grid-to-particle transfer and particle advection. Table 1 summarizes the default simulation and grid-warping parameters used in this study.

4.1 Predictive Free-Surface Importance

When grid adaptation is performed using only the current free-surface position, the grid response to rapidly moving or strongly deforming fluids may lag by one time step. To alleviate this issue, we first predict the near-future position of each FLIP particle. The particle position at prediction time τ is computed as

$$\hat{\mathbf{x}}_p = \mathbf{x}_p^n + \tau \mathbf{v}_p^n + \frac{1}{2} \tau^2 \mathbf{g}, \quad (5)$$

where $\mathbf{g}$ is the gravitational acceleration. To prevent an excessively large instantaneous velocity from dominating the entire metric, the predicted displacement is limited by

$$\|\hat{\mathbf{x}}_p - \mathbf{x}_p^n\| \leq d_{\text{pred}} h \quad (6)$$

and we use $d_{\text{pred}} = 2.4$ in our implementation. This bound does not constrain *grid warp displacement*; it is used only when estimating the future free-surface position.

The predicted particle positions are splatted onto grid nodes using a Gaussian kernel to construct the predicted density field ρ_i^p :

$$\rho_i^p = \sum_p K_\sigma (\|\mathbf{x}_i^0 - \hat{\mathbf{x}}_p\|), \quad (7)$$

where $\mathbf{x}_i^0$ is the regular-grid node position, and ρ_i^p is not a physical mass density but a surface-importance

signal constructed from the predicted particle distribution. The kernel support and standard deviation are set to $1.65h$ and $0.72h$, respectively. The density field is smoothed twice using the six neighboring nodes, and its gradient is used to compute local free-surface variation.

The normalized initial surface importance is defined as

$$\tilde{q}_i = S \left(\text{clamp} \left(\frac{\frac{\|\nabla \rho_i^p\|}{\max(\max_j \|\nabla \rho_j^p\|, \epsilon_g)} - q_0}{1 - q_0}, 0, 1 \right) \right), \quad (8)$$

where $q_0 = 0.08$, $\epsilon_g = 10^{-12}$, and $S(a) = a^2(3 - 2a)$ is the smoothstep function. Here, ϵ_g is a small constant used to prevent division by zero when the maximum gradient is close to zero. To reduce discontinuities in the metric caused by abrupt local changes in importance, $\tilde{q}_i$ is additionally smoothed twice together with its six neighboring nodes, and the result is used as the final importance q_i . The resulting q_i is a smooth interface-importance weight used to modulate the anisotropic metric near the predicted free surface.

4.2 Predictive Deformation Estimation

The free-surface position alone is insufficient to determine the direction in which spatial resolution should be concentrated in a given region. Therefore, we estimate the local velocity gradient using fluid-cell velocities around each grid node.

Let $\mathcal{N}_i$ denote the set of fluid cells around node i . In the implementation, only valid fluid cells are selected within a local $4 \times 4 \times 4$ region containing up to four cells along each axis. Let the cell center and velocity be $\mathbf{x}_c$ and $\mathbf{u}_c$, respectively. The velocity gradient $\mathbf{G}_i$ is computed by the following weighted least-squares problem:

$$\mathbf{G}_i = \arg \min_{\mathbf{G}} \sum_{c \in \mathcal{N}_i} w_{ic} \|(\mathbf{u}_c - \bar{\mathbf{u}}_i) - \mathbf{G}(\mathbf{x}_c - \bar{\mathbf{x}}_i)\|^2, \quad (9)$$

where $\bar{\mathbf{x}}_i$ and $\bar{\mathbf{u}}_i$ are the weighted averages of the neighboring fluid-cell centers and velocities, respectively, using w_{ic} . If fewer than four valid fluid cells are available, we set $\mathbf{G}_i = \mathbf{0}$, and a small diagonal regularization is applied to the normal equation to alleviate ill-conditioning. The distance-based weight is defined as

$$w_{ic} = \frac{1}{0.20h^2 + \|\mathbf{x}_c - \mathbf{x}_i\|^2} \quad (10)$$

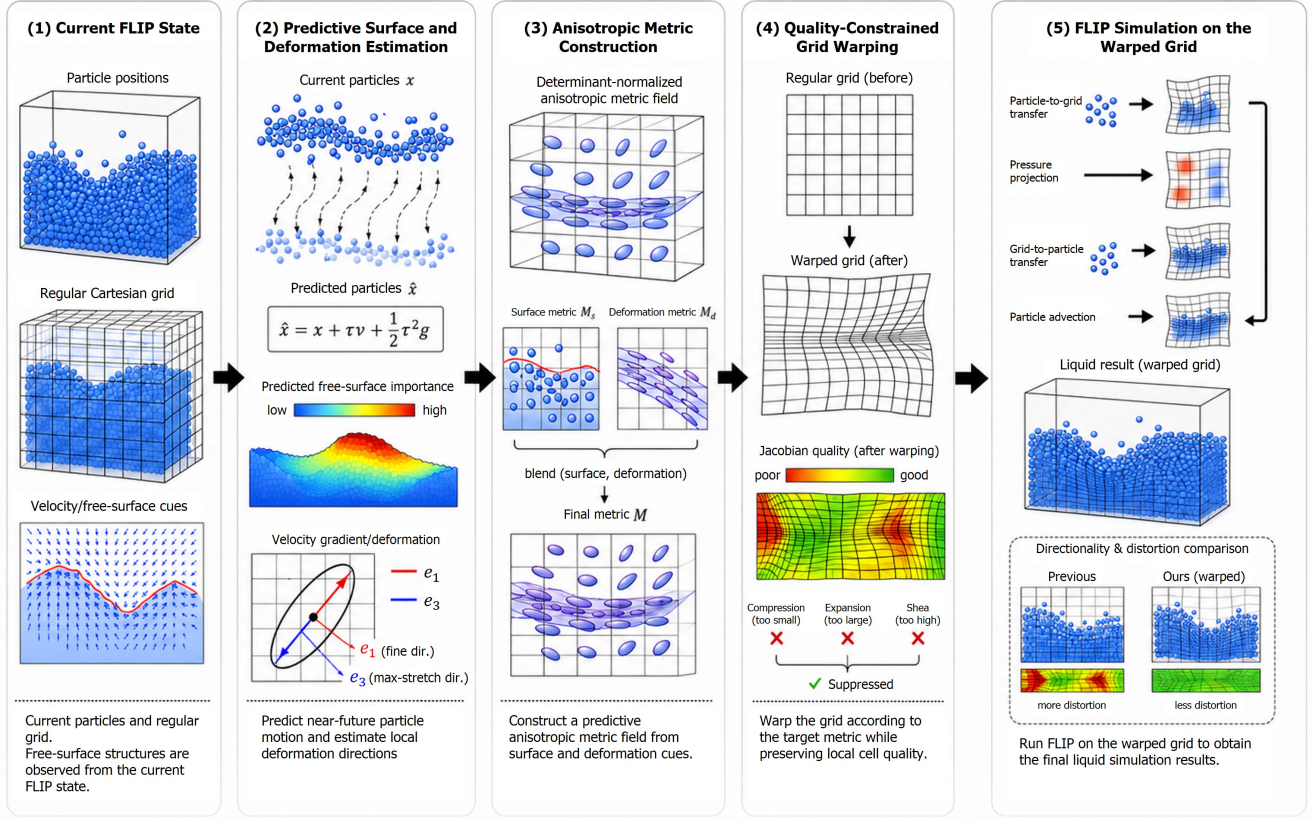


Fig. 2: Overview of the proposed quality-constrained predictive anisotropic grid warping framework. Starting from the current FLIP state, near-future particle motion is predicted to estimate free-surface importance and local deformation directions. These cues are combined to construct a determinant-normalized anisotropic metric field, which drives adaptive grid warping while Jacobian-based quality control suppresses excessive local compression, expansion, and shear distortion. The resulting warped grid is then used for particle-to-grid transfer, pressure projection, grid-to-particle transfer, and particle advection in the FLIP simulation.

The local deformation gradient over the prediction interval τ is approximated as

$$\mathbf{F}_i = \mathbf{I} + \tau \mathbf{G}_i \quad (11)$$

and, to robustly extract directional deformation, we construct

$$\mathbf{C}_i = \mathbf{F}_i^T \mathbf{F}_i \quad (12)$$

Let the eigenvalues of $\mathbf{C}_i$ be $\lambda_1 \leq \lambda_2 \leq \lambda_3$, with corresponding eigenvectors $\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$. The proposed method uses $\mathbf{e}_1$ as the preferred refinement direction and measures deformation directionality as

$$s_i = \log \sqrt{\frac{\lambda_3}{\max(\lambda_1, \epsilon_\lambda)}} \quad (13)$$

where $\epsilon_\lambda = 10^{-10}$ is a small constant for numerical stability of the eigenvalue ratio. A larger s_i indicates stronger predicted directional deformation. We use $\mathbf{e}_1$,

the eigenvector associated with the smallest principal stretch, as the fine direction because compression or thinning reduces the represented feature thickness along this direction. The principal-stretch ratio $\sqrt{\lambda_3/\lambda_1}$ controls deformation directionality, while $\mathbf{e}_1$ determines where the smallest physical grid spacing is assigned.

4.3 Determinant-Normalized Anisotropic Metric

If directional refinement is implemented simply by shrinking the entire cell, the local volume of the warped grid can decrease rapidly. Therefore, we redistribute anisotropic resolution among directions while keeping the metric determinant constant.

Let $\mathbf{n}$ be a unit refinement direction and $r \geq 1$ the target aspect ratio. We define the three-dimensional di-

rectional metric as

$$\mathbf{M}(\mathbf{n}, r) = m_c \mathbf{I} + (m_f - m_c) \mathbf{n} \mathbf{n}^T, \\ m_f = \frac{r^{4/3}}{h^2}, \quad m_c = \frac{r^{-2/3}}{h^2}, \quad \det(\mathbf{M}) = \frac{1}{h^6}. \quad (14)$$

Thus, $\det(\mathbf{M})$ is independent of r . This normalization preserves the overall scale of the *target metric*; it does not by itself guarantee preservation of the geometric volume of a warped cell.

We first define the free-surface normal as

$$\mathbf{n}_i^s = \begin{cases} \frac{\nabla \rho_i^p}{\|\nabla \rho_i^p\|}, & \|\nabla \rho_i^p\| > \epsilon_g, \\ \mathbf{e}_y, & \text{otherwise,} \end{cases} \quad (15)$$

where $\mathbf{e}_y = (0, 1, 0)^T$ is a fallback direction used when the gradient is nearly zero. Using

$$r_i^s = 1 + \gamma_s q_i \quad (16)$$

we construct the surface metric $\mathbf{M}_i^s = \mathbf{M}(\mathbf{n}_i^s, r_i^s)$. We use $\gamma_s = 2.2$ in our implementation.

The activation of the prediction-based deformation metric is defined as

$$a_i = q_i \frac{s_i}{s_i + \epsilon_s}, \quad \epsilon_s = 0.10 \quad (17)$$

. The corresponding deformation aspect ratio is

$$r_i^d = \exp[\text{clamp}(\gamma_d a_i, 0, \log r_{\max})], \quad (18)$$

where $\gamma_d = 4.0$ and $r_{\max} = 4.0$ are used.

Let the deformation metric be $\mathbf{M}_i^d = \mathbf{M}(\mathbf{e}_1, r_i^d)$. The final target metric is computed as

$$\mathbf{M}_i = \mathcal{R}_M[(1 - \beta_i) \mathbf{M}_i^s + \beta_i \mathbf{M}_i^d], \quad \beta_i = \min(\beta_d a_i, \beta_d), \quad (19)$$

where $\beta_d = 0.92$. The operator $\mathcal{R}_M[\cdot]$ renormalizes the determinant to h^{-6} and limits the metric aspect ratio to at most $r_{\max}$.

Because strongly moving only a subset of nodes near the free surface can itself produce high grid distortion, the generated metric is smoothly extended toward the six neighboring node directions. Specifically, at each node,

$$\overline{\mathbf{M}}_i^e = \frac{2\mathbf{M}_i + \sum_{j \in \mathcal{N}_6(i)} \mathbf{M}_j}{2 + |\mathcal{N}_6(i)|}, \\ w_i^e = \lambda_e (1 - 0.8 q_i), \quad (20) \\ \mathbf{M}_i \leftarrow \mathcal{R}_M[(1 - w_i^e) \mathbf{M}_i + w_i^e \overline{\mathbf{M}}_i^e].$$

Thus, the influence of the original metric is retained more strongly at nodes with high importance. In our implementation, $\lambda_e = 0.68$ is used and four diffusion steps are applied.

4.4 Quality-Constrained Grid Warping

Once the target metric is determined, the nodes are moved so that each grid edge has a length similar to that of the regular grid in metric space. For two adjacent nodes i and j ,

$$\mathbf{d}_{ij} = \mathbf{x}_j - \mathbf{x}_i, \quad \overline{\mathbf{M}}_{ij} = \frac{\mathbf{M}_i + \mathbf{M}_j}{2}, \quad \ell_{ij}^M = \sqrt{\mathbf{d}_{ij}^T \overline{\mathbf{M}}_{ij} \mathbf{d}_{ij}}. \quad (21)$$

We define the dimensionless relative distance between two nodes in the regular grid as $\ell_{ij}^0 = \|\mathbf{x}_j - \mathbf{x}_i\|/h$ and use it as the target value, moving grid nodes in the direction that reduces

$$E_{\text{metric}} = \frac{1}{2} \sum_{(i,j)} (\ell_{ij}^M - \ell_{ij}^0)^2 \quad (22)$$

. In the three-dimensional implementation, each internal node uses up to 26 neighboring nodes in its $3 \times 3 \times 3$ neighborhood, excluding itself. To preserve the computational-domain boundary, grid nodes on the domain boundary are fixed at their regular positions $\mathbf{x}_i^0$, and warping updates are applied only to internal nodes.

Using only the metric constraint can cause cell collapse or excessive shear under strong anisotropic deformation. Therefore, we compute the deformation Jacobian $\mathbf{J}_c$ at the center of each hexahedral cell c and apply a separate distortion regularization. Specifically, $\mathbf{J}_c$ is computed by differentiating the trilinear mapping of the cell at $(s, t, u) = (0.5, 0.5, 0.5)$ and normalizing its columns by the regular grid spacings h_x , h_y , and h_z .

First, local volume distortion is controlled by

$$E_{\text{vol}} = \frac{w_v}{2} \sum_c (\det(\mathbf{J}_c) - 1)^2 \quad (23)$$

. The default value of w_v is 1.05, and the regularization weight is further increased in regions where $\det(\mathbf{J}_c) < 0.72$ or $\det(\mathbf{J}_c) > 1.38$.

To directly control the discrepancy between the target anisotropy and the actual grid deformation, we define, using $\overline{\mathbf{M}}_c$, the average metric of the eight nodes of cell c ,

$$\mathbf{B}_c = h^2 \overline{\mathbf{M}}_c, \quad \mathbf{E}_c = \mathbf{J}_c^T \mathbf{B}_c \mathbf{J}_c - \mathbf{I} \quad (24)$$

and use

$$E_{\text{shape}} = \frac{1}{2} \sum_c w_c^q \|\mathbf{E}_c\|_F^2 \quad (25)$$

. Here,

$$w_c^q = \lambda_q (0.2 + 0.8 q_c) \quad (26)$$

with $\lambda_q = 0.32$, where q_c is the average surface importance q_i of the eight corner nodes of cell c . Accordingly, the agreement between the metric and actual cell deformation is enforced more strongly near the free surface and in regions of strong deformation.

No strong rest-position penalty is applied to $\|\mathbf{x}_i - \mathbf{x}_i^0\|$. Consequently, coherent multi-node motion can accumulate over time, while local distortion is controlled through the Jacobian-based terms.

4.5 Quality-Constrained Update

At each iteration, the metric force and distortion-regularization force are combined to compute the node displacement. Let the current node position be $\mathbf{x}_i$ and the warping-start position at the current time step be $\mathbf{x}_i^{\text{prev}}$. The quantity $\mathbf{f}_i$ denotes the accumulated warping force from the gradients of E_{metric} , E_{vol} , and E_{shape} , and the node update is

$$\Delta \mathbf{x}_i = \eta h^2 \left[\mathbf{f}_i - \frac{\lambda_t}{h^2} (\mathbf{x}_i - \mathbf{x}_i^{\text{prev}}) \right], \quad (27)$$

where $\eta = 0.042$ and $\lambda_t = 0.045$ are used. The second term alleviates grid jitter across consecutive time steps.

For numerical stability, the displacement in a single iteration is limited to $0.13h$, but this does not limit the *accumulated warp displacement*. Therefore, over multiple iterations and time steps, the grid can move sufficiently far from its regular position.

Once a candidate grid is generated, the following conditions are checked for every cell:

$$\frac{V_c}{V_c^0} \geq r_V, \quad J_{\min} \leq \det(\mathbf{J}_c) \leq J_{\max}, \quad \kappa(\mathbf{J}_c) \leq \kappa_{\max}. \quad (28)$$

In our implementation, $r_V = 0.30$, $J_{\min} = 0.42$, $J_{\max} = 1.90$, and $\kappa_{\max} = 4.20$. Here, $V_c^0 = h_x h_y h_z$ is the reference volume of cell c in the regular grid. The volume ratio and Jacobian determinant are used as complementary quality measures: the former evaluates signed-volume variation of the full hexahedral cell, whereas the latter evaluates local deformation at the cell center.

If any cell fails to satisfy these conditions, the current incremental step size is halved and reevaluated. Up to eight line-search trials are performed, and if an acceptable candidate is found, its position is adopted as the new grid. If no acceptable candidate is obtained after eight trials, the current warping iteration is terminated and the last accepted grid state is retained. The acceptance test therefore acts on the current incremental update rather than explicitly pulling the accumulated warp back toward the regular grid.

4.6 Metric-Change-Guided Temporal Tracking

The predictive metric need not undergo a full warping optimization at every frame during intervals in which the flow changes only slightly. To exploit this, we measure the change between the current metric $\mathbf{M}_i^n$ and the metric $\mathbf{M}_i^*$ stored at the most recent non-tracking warping update as

$$\delta_M = \sqrt{\frac{\sum_i \|\mathbf{M}_i^n - \mathbf{M}_i^*\|_F^2}{\max(\sum_i \|\mathbf{M}_i^*\|_F^2, \epsilon_M)}} \quad (29)$$

where $\epsilon_M = 10^{-30}$ is a small constant used to prevent the denominator from approaching zero.

Given three metric-change thresholds, $\tau_L = 0.008$, $\tau_M = 0.030$, and $\tau_H = 0.080$, the warping iteration budget N_w in predictive mode is adjusted as

$$N_w = \begin{cases} 9, & \delta_M \geq \tau_H, \\ 6, & \tau_M \leq \delta_M < \tau_H, \\ 3, & \tau_L \leq \delta_M < \tau_M, \\ 1, & \delta_M < \tau_L. \end{cases} \quad (30)$$

When $\delta_M < \tau_L$, the single iteration is used as a lightweight tracking update rather than a full re-optimization. Tracking is limited to two consecutive frames; afterward, a nine-iteration update is forced regardless of δ_M .

In the reactive formulation without prediction, temporal tracking is not applied, and a fixed maximum iteration budget is used at every time step. In reactive mode, we construct $\mathbf{D}_i = \frac{1}{2}(\mathbf{G}_i + \mathbf{G}_i^T)$ and, denoting its eigenvalues by $\mu_1 \leq \mu_2 \leq \mu_3$, compute the directional magnitude of the current deformation as $s_i^{\text{react}} = \max(0, (\mu_3 - \mu_1)\tau_r)$. Here, $\tau_r = \max(2\Delta t, 0.012)$, and the eigenvector corresponding to μ_1 is used as the fine direction. The subsequent activity and anisotropic metric construction are applied in the same manner as in the predictive formulation. Thus, comparison with the predictive formulation allows us to analyze the effect of directional grid adaptation using future flow information.

4.7 FLIP on the Warped Grid

After grid warping is completed, the center and volume of each hexahedral cell are recomputed from the updated node positions. The basic configuration for particle reference mapping, particle-grid transfer, and pressure projection on the warped grid follows the structured warped-grid formulation of Ibayashi et al. [15]. Because particles are stored in physical space, their logical coordinates must be updated to perform particle-to-grid transfer on the warped grid.

The position inside each warped hexahedron is represented by the trilinear mapping

$$\mathbf{x} = \mathcal{T}_c(s, t, u), \quad 0 \leq s, t, u \leq 1 \quad (31)$$

and the (s, t, u) corresponding to particle position $\mathbf{x}_p$ is computed using inverse trilinear mapping with Newton iteration. The logical coordinate from the previous time step is reused as the initial estimate, allowing the inverse mapping for most particles to be solved in the current cell or an immediately adjacent cell.

Particle velocities are transferred from logical coordinates to the eight neighboring cell centers using trilinear weights. For pressure projection, the actual warped-cell face unit normal $\mathbf{n}_f$ and area A_f are used. The velocity $\mathbf{u}_f$ at an internal fluid face is computed as the arithmetic average of the cell-centered velocities of the two fluid cells sharing the face, while the velocity of the current fluid cell is used for a face adjacent to a non-fluid region. Let d_f denote the face-normal distance between adjacent cell centers. The finite-volume pressure equation for warped cell c is

$$\sum_{f \in \partial c} \frac{A_f}{d_f} (p_c - p_{n(f)}) = -\frac{\rho}{\Delta t} \sum_{f \in \partial c} (\mathbf{u}_f \cdot \mathbf{n}_f) A_f. \quad (32)$$

Here, $p_{n(f)}$ is the pressure of the neighboring fluid cell sharing face f , and a zero-pressure condition is applied to non-fluid cells adjacent to the free surface. To prevent stiffness caused by excessively small distances, $d_f \geq 0.15h$ is enforced. The resulting linear system is solved using PCG with a diagonal preconditioner, with a maximum of 160 iterations and a relative tolerance of 2×10^{-5} . The pressure from the previous time step is used as the initial guess to improve convergence.

After the first pressure projection, an additional defect-correction solve is performed only when the mean absolute divergence remains at or above 70% of its value before projection. This maintains projection quality under strong grid deformation or violent free-surface motion without performing two pressure solves at every frame.

After pressure projection, particle velocities are updated using conventional FLIP/PIC blending:

$$\mathbf{v}_p^{\text{FLIP}} = \mathbf{v}_p^n + (\mathbf{u}^{n+1}(\mathbf{x}_p) - \mathbf{u}^n(\mathbf{x}_p)), \quad (33)$$

$$\mathbf{v}_p^{n+1} = (1 - \alpha)\mathbf{v}_p^{\text{FLIP}} + \alpha\mathbf{u}^{n+1}(\mathbf{x}_p), \quad (34)$$

where $\alpha = 0.08$. Finally, particle positions are advected using midpoint RK2 with the warped-grid velocity.

4.8 GPU Implementation

The entire solver is implemented using CUDA. Particle prediction, density splatting, metric construction,

grid warping, Jacobian evaluation, particle-grid transfer, pressure projection, and particle advection are parallelized as separate CUDA kernels. Each operation is independently dispatched on a per-particle, per-grid-node, or per-cell basis, and because the warped-grid topology remains identical to that of a regular structured grid, no separate dynamic connectivity data structure is required.

Particle rendering is implemented using CUDA-OpenGL interoperability, directly transferring particle positions and colors from GPU memory to a vertex buffer. The Jacobian determinant and condition number used for grid-quality visualization are treated as diagnostic quantities independent of the simulation state and therefore do not affect the simulation results themselves.

5 Experimental Results

5.1 Experimental Setup

To evaluate the grid-adaptation characteristics and numerical stability of the proposed method, we constructed a set of FLIP liquid simulation scenes involving different types of free-surface deformation. All experiments were performed using the CUDA-based implementation described in Sec. 4.8, while preserving regular Cartesian grid connectivity throughout the simulation.

The proposed formulation is inherently three-dimensional. Several experiments are nevertheless presented in a two-dimensional simulation environment to facilitate visualization and controlled analysis of grid adaptation and distortion; this changes only the experimental setting, not the underlying adaptation and quality-control principles. The default simulation and grid-warping parameters are summarized in Table 1. Unless otherwise stated, the dimension-independent parameters were kept fixed across the experiments rather than being tuned separately for individual scenes. For the volumetric configuration used in the three-dimensional experiments, the default grid resolution was $36 \times 28 \times 24$. The time step was $\Delta t = 0.006$, the prediction time was $\tau = 0.050$, and the maximum metric aspect ratio was $r_{\max} = 4.0$. To maintain stable grid quality, $J_{\min} = 0.42$, $J_{\max} = 1.90$, and $\kappa_{\max} = 4.20$ were used as quality constraints.

For comparison, we used Uniform FLIP on a regular Cartesian grid, the scalar dynamic-warping formulation of Ibayashi et al. [15], a reactive anisotropic variant using only current local deformation, and the proposed predictive quality-constrained formulation. Because Uniform FLIP does not deform the grid, it was used as a reference for adaptive-resolution allocation, while comparison with the previous method was used to analyze the difference between scalar-based dynamic

Parameter	Symbol	Value	Parameter	Symbol	Value
3D grid resolution	$N_x \times N_y \times N_z$	$36 \times 28 \times 24$	Time step	Δt	0.006
3D domain size	$W \times H \times D$	$10.0 \times 7.5 \times 6.0$	PIC blending	α	0.08
Prediction time	τ	0.050	Max. prediction displacement	d_{pred}	$2.40h$
Surface aspect gain	γ_s	2.20	Deformation aspect gain	γ_d	4.00
Maximum metric aspect	r_{max}	4.00	Deformation blending	β_d	0.92
Metric extension	λ_e	0.68	Metric extension iterations	N_e	4
Warp step	η	0.042	Maximum incremental step	—	0.13h
Temporal weight	λ_t	0.045	Metric-quality weight	λ_q	0.32
Volume-preservation weight	w_v	1.05	Minimum volume ratio	r_V	0.30
Minimum Jacobian determinant	J_{min}	0.42	Maximum Jacobian determinant	J_{max}	1.90
Maximum Jacobian condition	κ_{max}	4.20	Maximum warp iterations (reactive)	N_{max}	12
Metric-change thresholds	τ_L, τ_M, τ_H	0.008, 0.030, 0.080	Maximum hold frames	N_h	2
PCG tolerance	ϵ_p	2×10^{-5}	Maximum PCG iterations	N_p	160

Table 1: Default simulation and grid-warping parameters of the proposed method.

warping and the proposed predictive anisotropic adaptation. The reactive anisotropic formulation was included as a non-predictive directional baseline for evaluating the benefit of using anticipated deformation information.

The experimental scenes were designed to induce different forms of grid deformation. The dam-break scene includes large-scale translational motion and wall interaction, whereas the droplet-splash scene contains strong impact, compression, and surface reorientation over a short time interval. In the vortex-patch scene, the principal deformation direction changes continuously due to rotation and stretching, while the annular-collapse scene produces strong spreading along the bottom after rapid collapse. The sloshing-tank scene includes repeated wall impacts and flow-direction reversals and was used to evaluate whether grid distortion accumulates over long simulations. In addition, 3D dam break, two-block collision, and droplet splash scenes were used to verify that the proposed method operates robustly on volumetric hexahedral grids.

The evaluation was performed from both qualitative and quantitative perspectives. In the qualitative comparisons, the FLIP particle distribution and warped-grid deformation were visualized together to analyze how grid resolution is redistributed along the free surface. For the quantitative evaluation, normalized warp displacement, Jacobian determinant deviation, Jacobian condition number, high-distortion cell ratio, and grid-cell volume error were measured to analyze the relationship between adaptive deformation and local grid quality. In addition, post-projection divergence was used to evaluate the effect of grid distortion on FLIP pressure projection, and the per-frame computational time was measured to analyze the computational overhead introduced by predictive anisotropic warping and temporal tracking. Unless otherwise specified, all sequence-based

metrics were averaged over the complete simulation sequence, while global maxima or minima were additionally reported when worst-case behavior was important.

5.2 Importance and Metric Direction Visualization

The adaptation is driven by complementary cues: the predicted surface importance q_i identifies *where* resolution is needed, and the deformation metric specifies *in which direction* it should be redistributed.

Fig. 3 visualizes the surface importance computed at different time steps together with the FLIP particle distribution. The blue points represent FLIP particles, and the colors near the interface indicate the magnitude of the computed importance. Warmer colors from orange to red indicate higher importance.

In the initial state, high importance is formed primarily along the liquid free surface, and as the predicted free-surface shape changes, the high-importance region moves with the interface. Relatively high values occur near pronounced predicted interface variations, whereas low importance is observed in the fluid interior or in regions with weak density-gradient variation. Thus, q_i is not simply a liquid-occupancy measure; it is used as a weight for selecting the spatial regions on which subsequent adaptive grid warping should focus.

Fig. 4 shows the principal refinement directions of the anisotropic metric around the selected free-surface regions. The blue points represent FLIP particles, and the line segments near the interface indicate the metric direction computed at each location.

The metric direction varies with the local interface configuration and rotates as the free surface changes orientation, for example when the liquid spreads along the bottom or rises along a wall. Together, Figs. 3 and 4 show the spatial support and directional orientation of the adaptation. Through the determinant-normalized

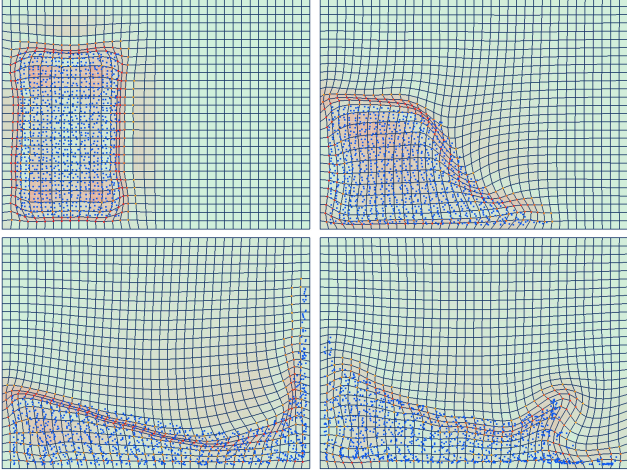


Fig. 3: Visualization of the predicted free-surface importance at four representative simulation states. Blue points represent FLIP particles, and interface cells are colored according to their importance, with warmer colors indicating larger values.

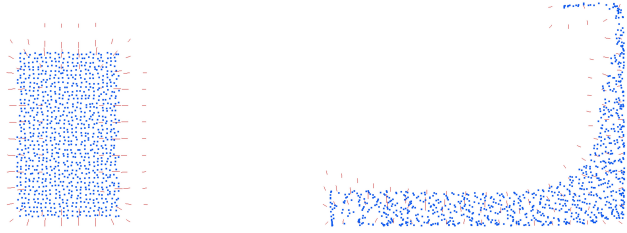


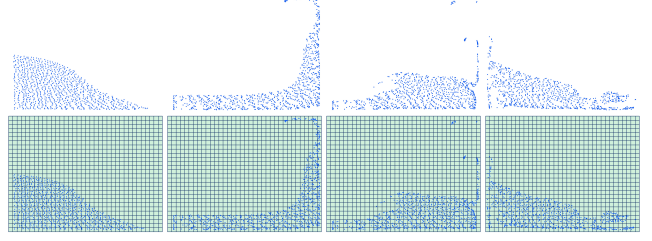
Fig. 4: Visualization of the anisotropic metric directions at two representative simulation states. Blue points represent FLIP particles, and line segments near the interface indicate the local principal refinement directions.

metric, higher resolution is assigned along the selected principal direction while compensating spacing changes occur in the remaining directions.

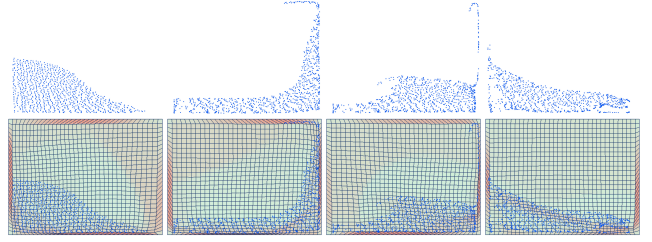
5.3 Qualitative Comparison of Grid Adaptation

In this section, we qualitatively compare basic FLIP, the conventional dynamic grid-warping method [15], and the proposed method in several two-dimensional test environments used for clearer visualization of grid adaptation and distortion. In the following comparisons, blue points denote FLIP particles and the warped-grid color indicates local distortion, with darker red denoting larger values. Uniform FLIP provides the non-adaptive reference, while the two warped-grid methods are compared under similar grid budgets.

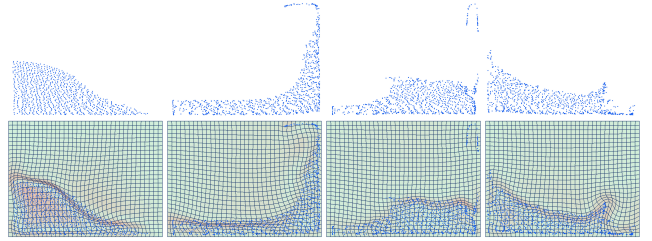
Fig. 5 compares grid adaptation under large-scale liquid motion and wall rise in the dam-break scene.



(a) FLIP



(b) Previous method [15]



(c) Our method

Fig. 5: Qualitative comparison of grid adaptation in the dam-break scene.

With basic FLIP, the same spatial resolution is maintained throughout the computational domain even when the liquid rapidly moves along the bottom or rises along the right wall.

The conventional method concentrates grid nodes along the free surface and allocates more computational degrees of freedom to the liquid region, but cell compression and shear increase substantially near the domain boundary and free surface as the flow moves to the right. In particular, relatively high distortion forms over broad regions near the upper and side boundaries and around the leading front of the liquid.

The proposed method redistributes grid lines more continuously as the liquid spreads along the bottom and rises along the wall. Boundary distortion is reduced while the grid remains visibly adapted to the free surface, and the particle distribution retains finer spatial support along the thin bottom flow and wall-rise regions.

Fig. 6 shows a falling droplet colliding with the free surface, producing strong compression, expansion, and surface reorientation over a short time interval. With

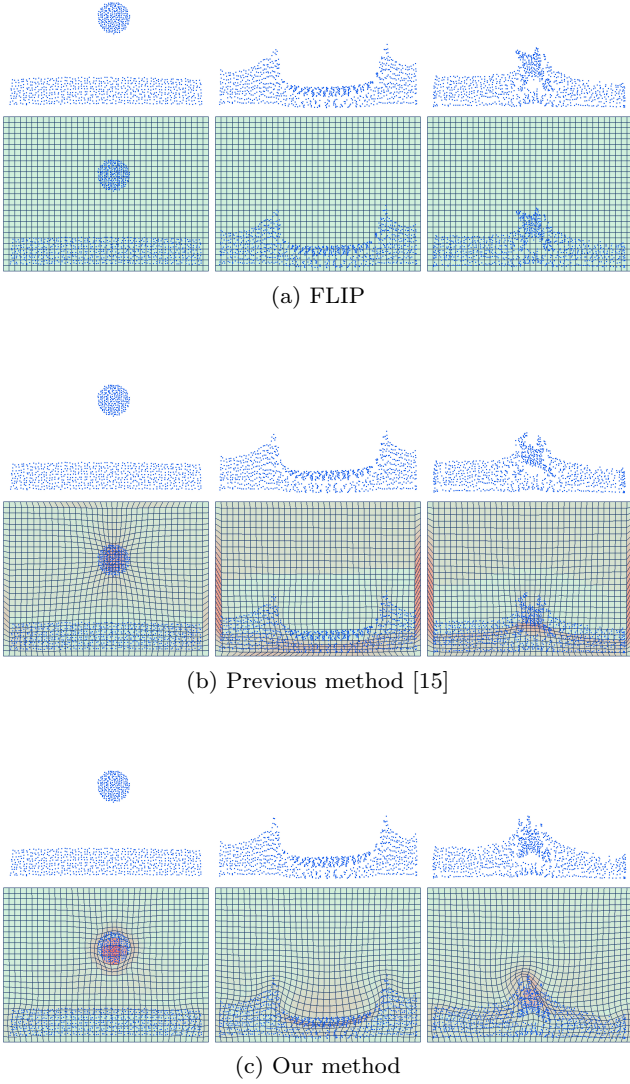


Fig. 6: Qualitative comparison of grid adaptation in the droplet-splash scene.

basic FLIP, a uniform spatial resolution is maintained even around the impact region and the subsequently formed cavity and splash.

The conventional dynamic warping method concentrates grid resolution around the droplet and the lower free surface, but as the collision progresses, grid spacing and orientation change abruptly in multiple regions, including the center and domain boundary, producing high distortion.

With the proposed method, the grid concentrates around the expected impact region before collision and then reorients with the cavity and rising splash. High distortion remains more localized than with conventional warping, while grid lines vary smoothly between

the central impact region and the surrounding free surface.

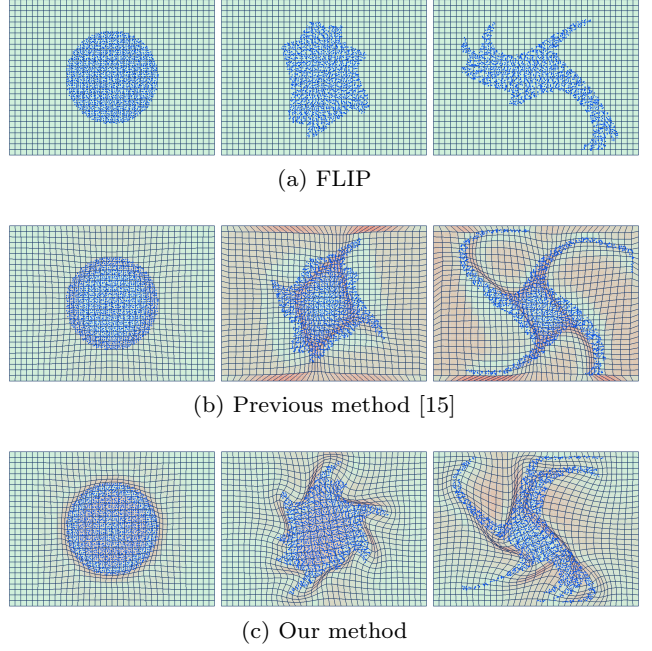


Fig. 7: Qualitative comparison of grid adaptation in the vortex-patch scene.

Fig. 7 shows a vortex-patch scene in which the deformation direction changes continuously due to rotation and stretching. This scene is particularly suitable for examining the difference between scalar spatial adaptation and predictive anisotropic adaptation.

With basic FLIP, the grid orientation and resolution remain unchanged even as the fluid deforms from an initially circular state through an asymmetric structure into a long and thin spiral-like structure. The conventional method moves the grid along the evolving fluid region, but because it is based on scalar importance, it cannot directly represent spatial changes in the stretching direction. Consequently, in later frames, some grid lines become sharply compressed or bent, and high distortion forms near the rotation center and domain boundary.

With the proposed method, the grid orientation rotates with the evolving stretch direction and remains aligned with the elongated particle distribution in later frames. Extreme local compression and shear are reduced relative to conventional warping, indicating that the adaptation follows temporally varying directions rather than only scalar spatial concentration.

Fig. 8 shows the difference between the conventional and proposed methods during large-scale collapse and bottom-directed spreading in the annular-collapse scene.

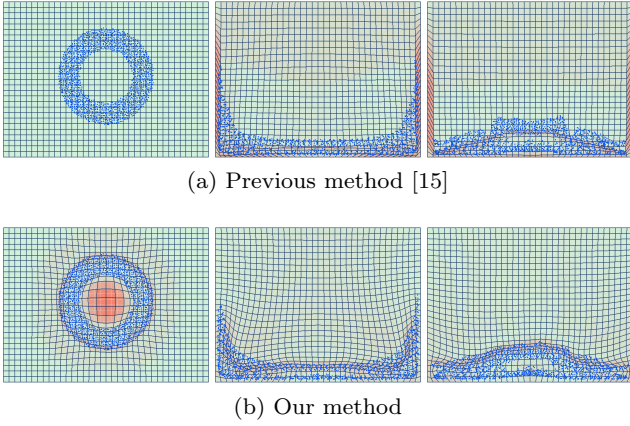


Fig. 8: Qualitative comparison of grid adaptation during annular collapse.

In the conventional method, as the fluid collapses, grid nodes become strongly concentrated near the bottom and side boundaries, and cell compression and distortion in these regions increase over time. In later frames, together with the severely deformed grid, the particle distribution near the center is observed to rise nonuniformly or become concentrated in specific regions. Although this behavior cannot be conclusively attributed directly to grid distortion, the spatial coincidence between low-grid-quality regions and abnormal particle distributions suggests that excessive warped-cell deformation may affect the stability of particle-grid computations.

The proposed method reduces abrupt compression near the bottom and side boundaries while retaining the large-scale deformation required by the collapse. The resulting grid remains more continuous, and the particle distribution spreads more consistently along the bottom in later stages.

Fig. 9 compares temporally accumulated grid deformation in a sloshing scene where the liquid repeatedly moves between the left and right walls. Because the same regions are repeatedly compressed and relaxed, this scene allows us to evaluate the long-term stability of grid adaptation.

In the conventional method, when the liquid rises along one wall, the grid becomes strongly concentrated near that wall and the bottom. Even after the direction of liquid motion reverses, deformation formed in the previous stage remains over a broad region and combines with new warping, tending to increase nonuniformity in cell spacing and orientation. In particular, high distortion repeatedly forms near the side and upper boundaries during repeated wall impacts.

With the proposed method, the grid follows each flow-direction reversal more continuously: deformation

Scene	Previous method [15]		Our method	
	$\bar{d}_{\text{avg}}$	d_{max}^*	$\bar{d}_{\text{avg}}$	d_{max}^*
Dam break	1.42	3.87	1.51	3.96
Falling liquid	1.28	3.41	1.34	3.52
Thin sheet	1.55	4.02	1.63	4.11
Annular collapse	1.37	3.76	1.44	3.82
Sloshing tank	1.61	4.18	1.67	4.24
Average	1.45	3.85	1.52	3.93

Table 2: Quantitative comparison of grid adaptation strength. The displacement is normalized by the representative grid spacing h .

in regions no longer occupied by the liquid relaxes, and resolution shifts toward the new free surface. This behavior reduces the accumulation of distortion near repeatedly impacted boundaries.

5.4 Quantitative Evaluation of Grid Adaptation

To quantify the differences observed in the qualitative comparisons, we evaluate complete simulation sequences for the major scenes in Fig. 5–9. The metrics include grid displacement, Jacobian-based quality, high-distortion cell ratio, cell-volume error, temporal stability, and post-projection divergence, allowing adaptation strength and local distortion to be assessed separately.

5.4.1 Grid Adaptation Strength

We first measure grid displacement to separate adaptation strength from local distortion. The normalized warp displacement d_i for each node follows the definition in Eq. (3), and to summarize adaptation strength over time we compute

$$\bar{d}(t) = \frac{1}{N_g} \sum_{i=1}^{N_g} d_i(t), \quad d_{\text{max}}(t) = \max_i d_i(t) \quad (35)$$

. In Table 2, $\bar{d}_{\text{avg}}$ denotes the temporal average of $\bar{d}(t)$, and d_{max}^* is the global maximum over the entire sequence.

Table 2 compares the grid displacement produced by the two methods. The proposed method yields an average normalized displacement of 1.52, which is approximately 4.8% larger than the 1.45 obtained with the conventional method. The global maximum is also similar or slightly larger, increasing from 3.85 to 3.93. In particular, for the thin-sheet and sloshing-tank scenes, the proposed method produces average normalized displacements of 1.63 and 1.67, respectively, confirming that grid nodes are moved sufficiently to respond to strong flow deformation.

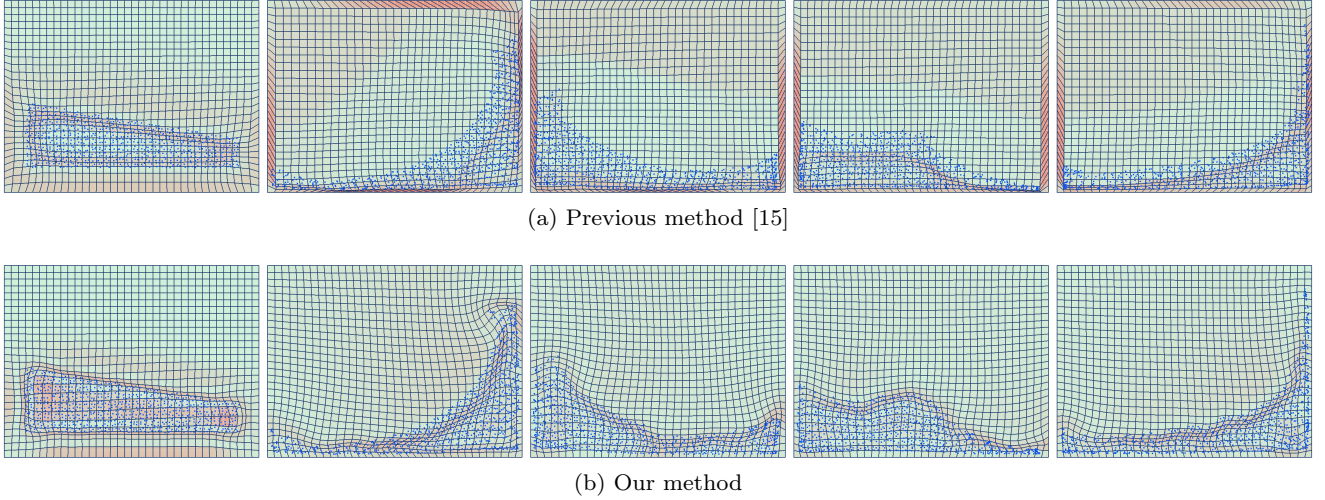


Fig. 9: Qualitative comparison of grid adaptation in the sloshing-tank scene.

The displacement statistics therefore show that the proposed grid remains at least as adaptive as conventional warping; the following metrics evaluate the corresponding local cell quality.

5.4.2 Jacobian-Based Grid Quality

Local cell deformation is evaluated using the deformation Jacobian $\mathbf{J}_c$ defined in the Problem Statement. The quantity $\det(\mathbf{J}_c)$ represents local compression/expansion, whereas $\kappa(\mathbf{J}_c)$ represents anisotropic shape distortion. The average deviation of the determinant from 1 over the complete sequence and its worst-case range are measured as

$$E_J = \frac{1}{N_f N_c} \sum_t \sum_{c=1}^{N_c} |\det(\mathbf{J}_c(t)) - 1|, \quad J_{\min}^* = \min_{t,c} \det(\mathbf{J}_c(t)), \quad J_{\max}^* = \max_{t,c} \det(\mathbf{J}_c(t)). \quad (36)$$

For shape distortion, the condition number defined in the Problem Statement is computed for each cell, and the sequence average $\bar{\kappa}$ and global maximum $\kappa_{\max}^*$ are reported.

As shown in Table 3, the proposed method exhibits lower Jacobian distortion than the conventional method in all scenes. The average determinant deviation E_J over all scenes decreases from 0.204 for the conventional method to 0.112 for the proposed method, corresponding to an improvement of approximately 45.1%. This indicates that excessive compression or expansion of cell volume relative to the reference state is substantially reduced during adaptive warping.

The difference between the two methods is also clear in the minimum determinant. For the conventional method, $J_{\min}^*$ decreases to 0.22 in the sloshing-tank scene and

reaches a similarly low value of 0.25 in the annular-collapse scene. These values indicate that some cells are strongly compressed relative to the regular state and are consistent with the local distortion regions shown in dark red in Fig. 8(a) and Fig. 9(a). In contrast, with the proposed method, the minimum determinant remains at or above 0.43 in all scenes, including $J_{\min}^* = 0.43$ for the strongly deforming sloshing-tank scene.

The maximum determinant reaches as high as 2.57 with the conventional method, whereas it remains below 1.90 in all scenes with the proposed method. The determinant bounds therefore limit both strong local compression and excessive expansion.

A similar trend is observed for shape distortion. The average condition number is 1.99 for the conventional method and decreases by approximately 25.1% to 1.49 with the proposed method. In particular, the maximum condition number increases to 5.73 for dam break, 6.31 for thin sheet, 6.67 for annular collapse, and 7.18 for sloshing tank with the conventional method. In contrast, the proposed method limits the corresponding values to 3.61, 3.89, 3.96, and 4.08, respectively. The condition-number results likewise show lower anisotropic shape distortion.

Combined with the displacement results in Table 2, these measurements indicate that the lower Jacobian distortion is obtained at comparable adaptation strength.

5.4.3 High-Distortion Cell Ratio

Because average Jacobian metrics may not sufficiently reveal a small number of severely deformed cells, we separately measured the fraction of regions exhibiting locally large distortion. The high-distortion cells and

Scene	Previous method [15]					Our method				
	E_J	$J_{\min}^*$	$J_{\max}^*$	$\bar{\kappa}$	$\kappa_{\max}^*$	E_J	$J_{\min}^*$	$J_{\max}^*$	$\bar{\kappa}$	$\kappa_{\max}^*$
Dam break	0.184	0.31	2.24	1.82	5.73	0.103	0.48	1.76	1.43	3.61
Falling liquid	0.163	0.34	2.12	1.71	5.21	0.094	0.51	1.71	1.39	3.44
Thin sheet	0.211	0.27	2.38	2.06	6.31	0.118	0.46	1.82	1.51	3.89
Annular collapse	0.226	0.25	2.46	2.14	6.67	0.121	0.44	1.84	1.53	3.96
Sloshing tank	0.238	0.22	2.57	2.23	7.18	0.126	0.43	1.87	1.57	4.08
Average	0.204	–	–	1.99	–	0.112	–	–	1.49	–

Table 3: Quantitative comparison of local grid quality over the complete simulation sequences. Lower E_J , $\bar{\kappa}$, and $\kappa_{\max}^*$ indicate lower grid distortion.

Scene	Previous method		Our method	
	$\bar{R}_{\text{dist}}$	$R_{\text{dist}}^{\max}$	$\bar{R}_{\text{dist}}$	$R_{\text{dist}}^{\max}$
Dam break	7.84%	15.62%	2.31%	6.42%
Falling liquid	6.47%	13.85%	1.94%	5.73%
Thin sheet	9.36%	18.91%	2.87%	7.11%
Annular collapse	10.82%	21.46%	3.02%	7.58%
Sloshing tank	12.31%	24.73%	3.46%	8.17%
Average	9.36%	18.91%	2.72%	7.00%

Table 4: Percentage of highly distorted cells during the simulation. Lower values indicate a smaller spatial extent of severe grid distortion.

their ratio are defined as

$$\mathcal{C}_{\text{dist}}(t) = \{c \mid \det(\mathbf{J}_c) < 0.60 \vee \det(\mathbf{J}_c) > 1.60 \vee \kappa(\mathbf{J}_c) > 3.0\}, \quad (37)$$

These diagnostic thresholds are intentionally stricter than the hard acceptance bounds used during grid warping, allowing the metric to capture cells that remain valid but already exhibit relatively strong distortion. Table 4 reports both the temporal average $\bar{R}_{\text{dist}}$ of $R_{\text{dist}}(t)$ and the maximum value $R_{\text{dist}}^{\max}$ over the entire sequence.

Table 4 quantitatively analyzes the red distortion regions observed in the qualitative visualizations. With the previous method, an average of 9.36% of the cells across all scenes satisfied the high-distortion criterion, whereas the proposed method reduced this ratio to 2.72%. This corresponds to a reduction of approximately 70.9%.

The largest differences occur in the annular-collapse and sloshing-tank scenes. For annular collapse, the average/maximum ratios decrease from 10.82%/21.46% to 3.02%/7.58%. For sloshing, the maximum ratio decreases from 24.73% to 8.17%. These values are consistent with the localized distortion patterns observed in the corresponding qualitative comparisons.

Scene	Previous method		Our method	
	E_V	$R_V^{\min}$	E_V	$R_V^{\min}$
Dam break	0.172	0.28	0.091	0.47
Falling liquid	0.151	0.31	0.084	0.50
Thin sheet	0.196	0.25	0.103	0.44
Annular collapse	0.209	0.23	0.108	0.42
Sloshing tank	0.221	0.20	0.114	0.41
Average	0.190	0.25	0.100	0.45

Table 5: Comparison of grid-cell volume preservation.

5.4.4 Grid-Cell Volume Preservation

In addition to the Jacobian determinant, we directly measured the volume variation of the actual warped hexahedral cells. Let the regular cell volume be V_c^0 and the warped volume at time t be $V_c(t)$. The warped volume ratio is defined as

$$e_V(c, t) = \frac{|V_c(t) - V_c^0|}{V_c^0}, \quad E_V = \frac{1}{N_f N_c} \sum_t \sum_c e_V(c, t), \quad R_V^{\min} = \min_c e_V(c, t). \quad (38)$$

Here, E_V denotes the average volume error over the entire sequence, and $R_V^{\min}$ denotes the worst-case volume ratio of the most strongly compressed cell.

As shown in Table 5, the average cell-volume error of the previous method was 0.190, whereas the proposed method reduced it to 0.100, corresponding to an improvement of approximately 47.4%. In particular, in the sloshing-tank scene, the minimum volume ratio of the previous method decreased to 0.20, whereas the proposed method maintained it at 0.41. A similar difference was observed in the annular-collapse scene, with values of 0.23 and 0.42, respectively.

The proposed formulation therefore reduces both average volume error and worst-case local compression during directional grid refinement.

5.4.5 Temporal Stability

In scenes such as sloshing, where the flow direction changes repeatedly, not only the quality of an individual frame but also the temporal accumulation of grid deformation and frame-to-frame variation are important. To evaluate these effects, we measure

$$\kappa_{\max}(t) = \max_c \kappa(\mathbf{J}_c(t)), \quad E_{\text{temp}}(t) = \frac{1}{N_g h} \sum_{i=1}^{N_g} \|\mathbf{x}_i^t - \mathbf{x}_i^{t-1}\| \quad (39)$$

The former represents the worst-case shape distortion in each frame, whereas the latter represents the normalized grid variation between consecutive frames.

The average temporal variation decreases from 0.118 for the previous method to 0.082 for the proposed method, a reduction of approximately 30.5%. Because Table 2 reports slightly larger overall displacement for the proposed method, this reduction reflects smoother temporal evolution rather than weaker spatial adaptation.

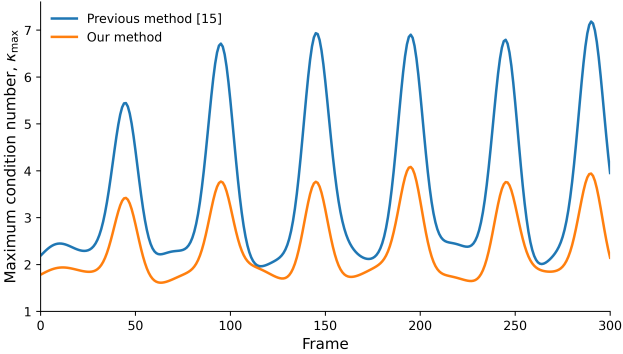


Fig. 10: Temporal evolution of the maximum Jacobian condition number in the sloshing tank scene.

Fig. 10 compares the temporal variation of $\kappa_{\max}(t)$ in the sloshing-tank scene. With the previous method, the condition number repeatedly increases sharply when the liquid collides with the left and right walls, and some peaks exceed 6.0–7.0. Moreover, even after the flow direction reverses, deformation from the previous stage is not fully relaxed and tends to overlap with newly generated distortion at the next wall impact.

In contrast, with the proposed method, the maximum condition number remains mostly within the range of 2.5–4.0 even under the same strong sloshing motion, and temporary peaks caused by wall impacts rapidly decrease. This behavior is consistent with Fig. 9(b), where grid deformation does not continuously accumulate near a specific boundary as the liquid repeatedly

Scene	Previous	Ours	Reduction
Dam break	0.0187	0.0138	26.2%
Falling liquid	0.0161	0.0124	23.0%
Thin sheet	0.0214	0.0152	29.0%
Annular collapse	0.0248	0.0166	33.1%
Sloshing tank	0.0265	0.0171	35.5%
Average	0.0215	0.0150	30.2%

Table 6: Average post-projection divergence for the warped-grid FLIP simulation.

moves from side to side, but is instead redistributed according to the current free surface and deformation direction.

The repeated wall impacts therefore produce smaller and shorter-lived peaks in worst-case cell distortion.

5.4.6 Influence of Grid Quality on FLIP Projection

To analyze how the reduction in grid distortion affects the stability of the actual fluid computation, we measured the residual divergence after pressure projection. Letting the set of fluid cells be $\mathcal{F}(t)$, the mean absolute divergence is defined as

$$E_{\text{div}}(t) = \frac{1}{|\mathcal{F}(t)|} \sum_{c \in \mathcal{F}(t)} |\nabla \cdot \mathbf{u}_c| \quad (40)$$

For each scene, the average value over the entire sequence was measured.

As shown in Table 6, the proposed method exhibits lower post-projection divergence in all test scenes. The overall average divergence decreases from 0.0215 with the previous method to 0.0150 with the proposed method, corresponding to an improvement of approximately 30.2%.

The largest reductions, 33.1% and 35.5%, occur in the annular-collapse and sloshing-tank scenes, which also exhibit the largest differences in grid distortion. The correlation is consistent with the expectation that better-conditioned warped-cell geometry benefits face-based flux evaluation and pressure projection.

A related qualitative difference appears in the annular-collapse result: the previous method shows severe bottom-region compression together with localized upward particle accumulation, whereas the proposed method exhibits lower distortion and more continuous spreading along the bottom. Although this visual correspondence does not by itself establish causality, it is consistent with the lower post-projection divergence measured for the proposed grid.

5.5 Computational Performance

Compared with conventional scalar grid warping, the proposed method additionally performs near-future particle prediction, velocity-gradient estimation, anisotropic metric construction, and Jacobian-based quality evaluation. We therefore analyzed the computational overhead introduced by these additional operations. All timings were measured using the same hardware and simulation configuration, and the average execution time per frame was computed over the entire sequence of each scene.

Table 7 shows the average computational cost of each method. Because Uniform FLIP does not perform grid adaptation, it has the lowest computation time at 8.43 ms per frame. The previous dynamic-warping method requires 12.47 ms due to the additional metric computation and grid warping.

Ours (no tracking), which performs predictive anisotropic warping with full iterations in every frame, incurs the highest cost at 15.20 ms. Of this cost, prediction and metric construction account for 0.68 ms and 1.38 ms, respectively, while grid warping accounts for 4.27 ms.

In contrast, the final method with metric-change-guided temporal tracking reduces the total time to 13.63 ms. This corresponds to an approximately 10.3% reduction in total computation time compared with the formulation without tracking, while the warping stage alone is reduced by approximately 31.9%, from 4.27 ms to 2.91 ms.

The final method is approximately 9.3% slower than the previous dynamic-warping method, while the measured determinant deviation, high-distortion cell ratio, and post-projection divergence are lower by approximately 45.1%, 70.9%, and 30.2%, respectively. Because the structured-grid topology is preserved, no remeshing or dynamic connectivity reconstruction is required; the additional work is concentrated in prediction, metric evaluation, and quality-aware warping, which are parallelized on the GPU.

5.6 Ablation Study

We examine the roles of prediction, local quality control, and temporal tracking using four configurations: reactive anisotropy without prediction, predictive anisotropy without quality control, prediction with quality control but without tracking, and the full method.

The reactive baseline has the smallest average displacement among the anisotropic methods and yields $E_J = 0.158$ with a high-distortion ratio of 6.21%. Compared with this baseline, the predictive-only configura-

tion increases the average displacement to 1.56, indicating a stronger anticipatory response.

Prediction alone, however, increases local distortion to $E_J = 0.191$ and $\bar{R}_{\text{dist}} = 8.37\%$. This comparison shows that anticipatory anisotropy by itself is insufficient; the stronger directional response must be paired with explicit local quality control.

Adding quality control reduces E_J to 0.112 and the high-distortion ratio to 2.73%, while the average displacement remains 1.53. Thus, the quality constraints reduce local distortion without materially suppressing adaptation. Temporal tracking leaves the reported grid-quality metrics essentially unchanged but reduces the total time from 15.20 ms to 13.63 ms.

These results separate the roles of the three components: prediction strengthens anticipatory adaptation, quality control stabilizes the resulting warped cells, and temporal tracking reduces optimization cost.

5.7 Three-Dimensional Simulation Results

We further evaluate the method in 3D dam-break, two-block collision, and droplet-splash scenes. These cases introduce volumetric combinations of compression, stretching, and shear and therefore test whether the directional metric and Jacobian-based quality control remain stable on deformed hexahedral grids.

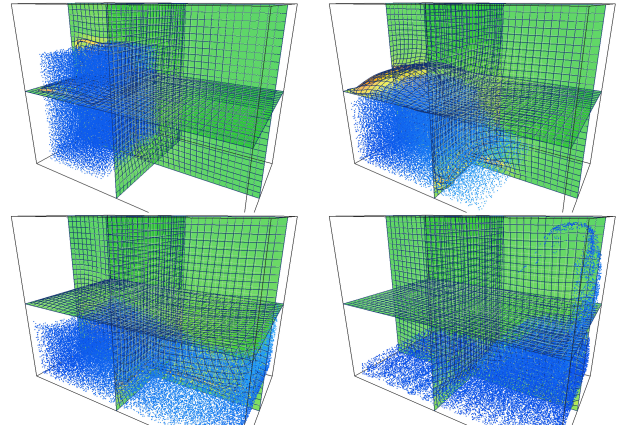


Fig. 11: Three-dimensional dam-break simulation with adaptive grid warping.

Fig. 11 shows a 3D dam-break scene in which an initially tall liquid column collapses, rapidly spreads along the bottom, and subsequently rises along the opposite wall. During the initial collapse, grid resolution is selectively concentrated near the leading front and free surface, and the grid is continuously redistributed along the predicted deformation direction as the liquid

Method	Prediction	Metric	Warping	Pressure	Others	Total
Uniform FLIP	–	–	–	4.82	3.61	8.43
Previous method [15]	–	0.74	2.83	5.16	3.74	12.47
Reactive anisotropic	–	1.12	3.46	5.09	3.79	13.46
Ours (no tracking)	0.68	1.38	4.27	5.04	3.83	15.20
Ours	0.68	1.21	2.91	5.01	3.82	13.63

Table 7: Average computational cost per simulation frame in milliseconds.

Method	$\bar{d}_{\text{avg}}$	E_J	$\bar{\kappa}$	$\bar{R}_{\text{dist}}$	Time (ms)
Reactive anisotropic baseline	1.43	0.158	1.78	6.21%	13.46
Predictive anisotropic (no quality control)	1.56	0.191	1.94	8.37%	13.92
Predictive + quality control	1.53	0.112	1.49	2.73%	15.20
Full method	1.52	0.112	1.49	2.72%	13.63

Table 8: Ablation study of prediction, local quality control, and temporal tracking.

rapidly moves along the bottom. Even when the liquid subsequently collides with the wall and rises upward, the grid deforms three-dimensionally with the changing free surface without exhibiting abrupt local cell collapse.

After wall impact, the principal deformation direction changes from predominantly horizontal to vertical; the metric reorients accordingly while the quality constraints prevent abrupt local collapse.

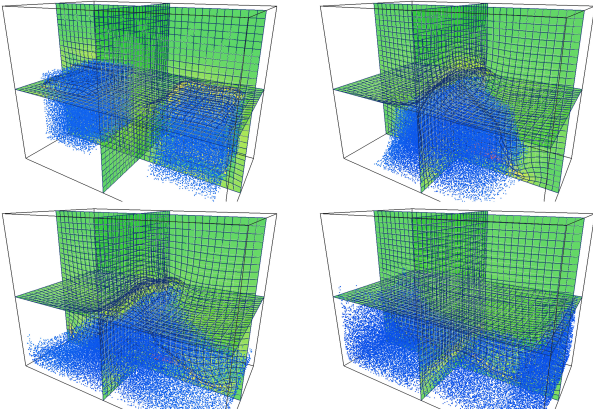


Fig. 12: Three-dimensional two-block collision simulation with adaptive grid warping.

Fig. 12 shows a scene in which two opposing liquid blocks collide in the central region. As the two liquids moving in opposite directions collide, strong compression first occurs in the center, followed by complex deformation as the liquid spreads laterally and upward. Consequently, the grid cannot be effectively controlled using a single dominant direction, and the principal deformation direction changes rapidly over time.

Before collision, the two liquid blocks induce different local deformation directions. After impact, resolution shifts toward the compressed center and then follows the subsequent lateral spreading, demonstrating that the metric responds to changing deformation direction rather than liquid occupancy alone.

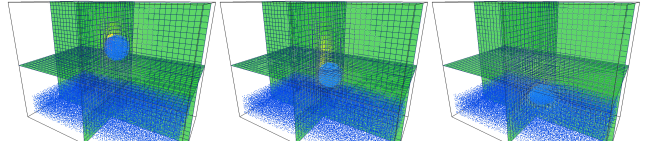


Fig. 13: Three-dimensional droplet-splash simulation with adaptive grid warping.

Fig. 13 shows a falling droplet colliding with the liquid surface below, producing a splash and localized upward motion. This scene differs from the previous two in that strong compression occurs in the impact region over a short time interval, after which the liquid rapidly spreads radially while stretching develops simultaneously in multiple directions.

The predicted particle distribution concentrates resolution near the impending impact region. After impact, the metric follows radial and vertical deformation, while the Jacobian constraints limit excessive aspect-ratio growth around thinning free-surface structures.

For the three-dimensional scenes, grid quality was also evaluated using the Jacobian-based metrics defined in the preceding section. The average Jacobian determinant deviations E_J for the three scenes were 0.126, 0.139, and 0.147, respectively, while the average condition numbers $\bar{\kappa}$ were 1.56, 1.63, and 1.68. The maximum condition numbers over the entire sequences were maintained at 4.03 for the 3D dam break, 4.11 for the

two-block collision, and 4.16 for the droplet splash, indicating that grid deformation remains stable within the quality-constraint range used in this study even in three dimensions.

The high-distortion cell ratios were also limited to 3.18%, 3.64%, and 3.92%, respectively, indicating that relatively large distortion occurs only in a small fraction of the overall grid despite complex three-dimensional deformation. The post-projection divergence values were measured as 0.0168, 0.0179, and 0.0186 for the three scenes, respectively, indicating that pressure projection remains numerically well behaved in the tested three-dimensional scenes.

Together, the three scenes show that the same adaptation and quality-control mechanisms extend to volumetric hexahedral grids under large-scale motion, bidirectional collision, and localized splashes.

6 Discussion

The experiments suggest that predictive directionality and geometric quality control play complementary roles. Prediction is most useful when the free surface moves or reorients rapidly, because the metric can respond before the deformation is fully expressed in the current geometry. The ablation also shows the limitation of using prediction alone: stronger anticipatory anisotropy can increase local distortion unless the warped cells are explicitly regularized.

The Jacobian constraints address this issue locally rather than by pulling all nodes toward their regular-grid positions. This distinction is reflected in the displacement and quality measurements: the proposed grid moves at least as far as the conventional warped grid on average, while exhibiting substantially smaller determinant and condition-number errors. Temporal tracking affects a different aspect of the method, reducing repeated optimization when the metric changes slowly with little measurable effect on grid quality.

The 3D experiments show the same qualitative behavior under volumetric compression, stretching, and shear. The method is therefore best viewed as a structured-grid adaptation strategy that couples directional resolution allocation with explicit local quality control.

The method nevertheless has several limitations. First, near-future deformation is estimated using a short-time approximation based on the current particle velocity and local velocity gradient, and prediction errors may therefore increase during abrupt collisions or complex vortex interactions. Second, the current metric primarily uses predicted free-surface importance and deformation direction as cues, and thus may not always optimally resolve fine-scale structures dominated by other

physical features such as curvature, vorticity, or pressure variation. Third, quality thresholds such as $J_{\min}$, $J_{\max}$, and $\kappa_{\max}$ are fixed throughout the experiments and may require retuning for extreme deformations or different resolution ranges.

Future work could automatically adjust the prediction horizon and quality thresholds according to the current flow state and incorporate additional features such as curvature or vorticity into the metric. In addition, comparisons with uniform high-resolution FLIP and other adaptive discretizations under not only the same cell count but also the same computational budget would enable a more direct evaluation of the resolution-allocation efficiency of the proposed method.

7 Conclusion

We presented a quality-constrained predictive anisotropic grid-warping method for FLIP liquids. The method predicts directional deformation, redistributes resolution with a determinant-normalized metric, and controls local warped-cell quality through Jacobian-based regularization and acceptance constraints. Metric-change-guided tracking reduces repeated optimization when the target metric varies slowly.

Across the tested two- and three-dimensional scenes, the method maintained comparable or greater grid displacement than conventional dynamic warping while reducing Jacobian distortion, highly distorted cells, cell-volume error, temporal variation, and post-projection divergence. These results support predictive anisotropic adaptation when it is coupled with explicit geometric quality control.

Future work will incorporate additional flow cues such as curvature, vorticity, and pressure variation and will investigate adaptive prediction horizons and quality thresholds.

Acknowledgements (not compulsory)

The authors received no specific funding for this work.

Funding

This work was supported by INHA UNIVERSITY Research Grant. This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIT) (RS-2026-25474535). This work was supported by Institute of Information & communications Technology Planning & Evaluation (IITP)

grant funded by the Korea government (MSIT) (RS-2026-25548560, Artificial Intelligence Innovation Graduate School (Inha University)).

Author contributions statement

J.H.K. conceived the study, developed the methodology, implemented the system, conducted the experiments, analyzed the results, and wrote and reviewed the manuscript.

References

1. J. U. Brackbill and H. M. Ruppel, "FLIP: A method for adaptively zoned, particle-in-cell calculations of fluid flows in two dimensions," *Journal of Computational Physics*, vol. 65, no. 2, pp. 314–343, 1986. doi: 10.1016/0021-9991(86)90211-1.
2. N. Foster and R. Fedkiw, "Practical animation of liquids," in *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH 2001)*, pp. 23–30, 2001. doi: 10.1145/383259.383261.
3. F. Losasso, F. Gibou, and R. Fedkiw, "Simulating water and smoke with an octree data structure," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 457–462, 2004. doi: 10.1145/1015706.1015745.
4. B. M. Klingner, B. E. Feldman, N. Chentanez, and J. F. O'Brien, "Fluid animation with dynamic meshes," *ACM Transactions on Graphics*, vol. 25, no. 3, pp. 820–825, 2006. doi: 10.1145/1179352.1141961.
5. N. Chentanez, B. E. Feldman, F. Labelle, J. F. O'Brien, and J. R. Shewchuk, "Liquid simulation on lattice-based tetrahedral meshes," in *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 219–228, 2007. doi: 10.2312/SCA/SCA07/219-228.
6. N. Chentanez and M. Müller, "Real-time Eulerian water simulation using a restricted tall cell grid," *ACM Transactions on Graphics*, vol. 30, no. 4, pp. 82:1–82:10, 2011. doi: 10.1145/2010324.1964977.
7. R. Ando, N. Thuerey, and C. Wojtan, "Highly adaptive liquid simulations on tetrahedral meshes," *ACM Transactions on Graphics*, vol. 32, no. 4, pp. 103:1–103:10, 2013. doi: 10.1145/2461912.2461982.
8. B. Zhu, W. Lu, M. Cong, B. Kim, and R. Fedkiw, "A new grid structure for domain extension," *ACM Transactions on Graphics*, vol. 32, no. 4, pp. 63:1–63:8, 2013. doi: 10.1145/2461912.2461999.
9. R. E. English, L. Qiu, Y. Yu, and R. Fedkiw, "Chimera grids for water simulation," in *Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, pp. 85–94, 2013. doi: 10.1145/2485895.2485897.
10. F. Ferstl, R. Westermann, and C. Dick, "Large-scale liquid simulation on adaptive hexahedral grids," *IEEE Transactions on Visualization and Computer Graphics*, vol. 20, no. 10, pp. 1405–1417, 2014. doi: 10.1109/TVCG.2014.2307873.
11. R. Setaluri, M. Aanjaneya, S. Bauer, and E. Sifakis, "SPGrid: A sparse paged grid structure applied to adaptive smoke simulation," *ACM Transactions on Graphics*, vol. 33, no. 6, pp. 205:1–205:12, 2014. doi: 10.1145/2661229.2661269.
12. C. Jiang, C. Schroeder, A. Selle, J. Teran, and A. Stam, "The affine particle-in-cell method," *ACM Transactions on Graphics*, vol. 34, no. 4, pp. 51:1–51:10, 2015. doi: 10.1145/2766996.
13. M. Aanjaneya, M. Gao, H. Liu, C. Batty, and E. Sifakis, "Power diagrams and sparse paged grids for high resolution adaptive liquids," *ACM Transactions on Graphics*, vol. 36, no. 4, pp. 140:1–140:12, 2017. doi: 10.1145/3072959.3073625.
14. C. Fu, Q. Guo, T. Gast, C. Jiang, and J. Teran, "A polynomial particle-in-cell method," *ACM Transactions on Graphics*, vol. 36, no. 6, pp. 222:1–222:12, 2017. doi: 10.1145/3130800.3130878.
15. H. Ibayashi, C. Wojtan, N. Thuerey, T. Igarashi, and R. Ando, "Simulating liquids on dynamically warping grids," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 6, pp. 2288–2302, 2020. doi: 10.1109/TVCG.2018.2883628.
16. X. Zhai, F. Hou, H. Qin, and A. Hao, "Fluid simulation with adaptive staggered power particles on GPUs," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 6, pp. 2234–2246, 2020. doi: 10.1109/TVCG.2018.2886322.
17. R. Ando and C. Batty, "A practical octree liquid simulator with adaptive surface resolution," *ACM Transactions on Graphics*, vol. 39, no. 4, pp. 32:1–32:17, 2020. doi: 10.1145/3386569.3392460.
18. Y. Xiao, S. Chan, S. Wang, B. Zhu, and X. Yang, "An adaptive staggered-tilted grid for incompressible flow simulation," *ACM Transactions on Graphics*, vol. 39, no. 6, pp. 171:1–171:15, 2020. doi: 10.1145/3414685.3417837.
19. T. Koike, S. Morishima, and R. Ando, "Asynchronous Eulerian liquid simulation," *Computer Graphics Forum*, vol. 39, no. 2, pp. 1–8, 2020. doi: 10.1111/cgf.13907.
20. R. Nakanishi, F. Nascimento, R. Campos, P. Pagliosa, and A. Paiva, "RBF liquids: An adaptive PIC solver using RBF-FD," *ACM Transactions on Graphics*, vol. 39, no. 6, pp. 170:1–170:13, 2020. doi: 10.1145/3414685.3417794.
21. E. A. Parreiras, M. B. Vieira, A. G. Machado, M. C. Renhe, and G. A. Giraldo, "A particle-in-cell method for anisotropic fluid simulation," *Computers & Graphics*, vol. 102, pp. 220–232, 2022. doi: 10.1016/j.cag.2021.08.010.
22. Z. Qu, M. Li, F. de Goes, and C. Jiang, "The power particle-in-cell method," *ACM Transactions on Graphics*, vol. 41, no. 4, pp. 1–13, 2022. doi: 10.1145/3528223.3530066.
23. S. Sancho, J. Tang, C. Batty, and V. C. Azevedo, "The impulse particle-in-cell method," *Computer Graphics Forum*, vol. 43, no. 2, Art. no. e15022, 2024. doi: 10.1111/cgf.15022.
24. F. Narita, N. Ochiai, T. Kanai, and R. Ando, "Quadtree tall cells for Eulerian liquid simulation," in *SIGGRAPH Conference Papers '25*, Art. no. 10, pp. 1–11, 2025. doi: 10.1145/3721238.3730652.
25. F. Narita and T. Kanai, "Adaptive optical layers: Efficient tall cell grids for liquid simulation," *Computer Graphics Forum*, vol. 45, no. 2, Art. no. e70357, 2026. doi: 10.1111/cgf.70357.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [GoogleDrivelinktodownloadtheresultvideos.txt](#)