Correspondence-based Fluid Control with a Hybrid Particle-Grid Simulation Method

Kiwon Um  Seungho Baek  Junghyun Han
Korea University  Korea University  Korea University

Abstract
This paper proposes an effective fluid control framework based on a hybrid particle-grid simulation method. The proposed framework represents both fluid and target with particle primitives and implements the control method using the correspondences between the fluid and target particles. The proposed method is smoothly integrated into the underlying simulation method. The framework enables the user to produce a desired animation using the target object shapes and their transformations.

Keywords: Fluid Control, Hybrid Particle-Grid Simulation, Fluid Implicit Particle

1 Introduction
While computer graphics field has focused on realistic and efficient simulation of fluids, developing intuitive and effective methods for controlling fluids has also been an important objective for artistic purposes. An apparent approach is to resolve the overall fluid dynamics using physically driven models and then control the fluid motions using additional techniques. However, fluid behavior is generally complex and thus difficult to predict. Therefore, controlling fluids as desired is still a challenging and time-consuming task.

This paper proposes an effective target-driven fluid simulation framework based on a hybrid particle-grid simulation method and introduces an intuitive and easy-to-use method to control the fluid motions. The control method represents both the fluid and target with particle primitives and uses the correspondences between them for handling the fluid motions. This correspondence-based method unifies different control methods adopted for distinct types of fluids (e.g., water and smoke) and provides the user with a straightforward and consistent channel for controlling, thereby allowing the user to focus on building (or updating) the correspondences to obtain a desired animation.

2 Related Work
Numerous approaches have been studied for controlling fluids [1, 2]. Treuille et al. [3] specified smoke-density and velocity keyframes to control fluid shape. Shi and Yu [4] represented the smoke region and the target object using implicit functions and applied velocity constraints to the shape matching process. A path control method has been developed to move the fluid along a given path [5]. Yang et al. [6] developed a unified framework for both path and shape controls by using a signed distance field.

On the other hand, a low-resolution animation has been utilized as a guide to produce a targeted high-resolution animation. Nielsen and Bridson [7] proposed a method for constraining or guiding a high-resolution liquid simulation.
such that it becomes close to the target animation. Thurey et al. [8] proposed a scale-dependent force control that preserves the small-scale fluid details and avoids the artificial viscosity of force-based control methods. Additionally, Zhang and Liu [9] has developed particle-based control approaches.

3 Targeted Animation

The purpose of this study is to provide the user with an intuitive and consistent method of controlling the fluids. For this purpose, the proposed method represents the fluid as a set of particles. A noticeable strength of the particle primitives is that they can easily preserve the fluid mass during the simulation period. Moreover, the user can simply add or remove particles and directly change the particles’ positions and velocities.

In this study, the underlying simulation procedure is implemented by adopting a variant of fluid implicit particle (FLIP) [10], which is useful for fluid control because the preserved particles and their proper distribution provide users with an intuitive and robust control channel. The additional steps can be separated from the underlying procedure; this enables the proposed framework to release the control seamlessly. As shown in Figure 1, the proposed framework effectively produces the targeted animations for both water and smoke.

3.1 Sampling Target Shape

![Fig. 2: Morphing shapes. An initial water ball is transformed into three different shapes in order: box, bunny, and duck.](image)

The proposed framework samples the target shape using particles. For a given object that represents a target shape, a set of particles is placed inside the object, and the positions of the particles are defined in the object’s local space. If the object is transformed, the particles’ positions can be simply recomputed using the object’s transformation. The sampling rate for the target particles is determined by the particle resolution of the underlying fluid simulation. After sampling the target shape, each target particle is initialized by selecting a fluid particle that does not correspond to any target particle.

3.2 Fluid Force

Each target particle corresponds to a unique fluid particle, and the correspondence exerts the driving force. When a fluid particle \(i\) corresponds to a target particle \(j\) and their positions are \(x_i\) and \(x_j\), respectively, the attraction force of the fluid particle can be simply defined as follows:

\[
f_{ij} = k_a \frac{x_{ij}}{|x_{ij}|},
\]

where \(k_a\) is the control coefficient for the strength of the attraction force, and \(x_{ij}\) indicates the targeted direction, i.e., \(x_j - x_i\). The other fluid particles, which do not correspond to any target particles, are exerted by typical external forces (e.g., gravity and buoyancy).

3.3 Underlying Fluid Simulation

The procedure of the underlying fluid simulation is summarized as follows: The particles’ data are transferred to the grid using a typical method, i.e., the weighted average of nearby particles’. The external forces, then, are used to update the grid velocity, and it is projected to determine the divergence-free velocity field. The divergence-free velocity field updates the particles’ velocities. The particles are then moved using the grid velocity and corrected.

In order to unify the computation of external forces, this study adopts the material point method [11]. The external forces computed on the grid are firstly interpolated to the positions of the fluid particles except the targeted particles as described in Section 3.2. The forces including both the external and the correspondence-based driving forces, then, are transferred to the grid, together with the masses, using basis functions [11] to update the grid velocity.
If the targeted particles are close enough to their target positions, the particle positions are corrected; in the current implementation, the targeted particle $i$ inside the target shape slightly moves to the position of its target particle $j$ as follows:

$$x'_i = x_i + k_c h \frac{x_{ij}}{|x_{ij}|} \quad (2)$$

where $k_c$ is the correction coefficient and $h$ is the initial particle spacing. When the position is corrected, the targeted particle is directly moved to the target position if the distance between the fluid particle and the target particle is less than a threshold, e.g., $|x_{ij}| < 0.5h$.

### 3.4 Velocity Filtering

The particles carry the simulation data. This property provides an intuitive channel for controlling the motion of the targeted fluid. For a fluid particle $i$ that corresponds to a target particle $j$, its velocity $v_i$ is decomposed into two vectors: $v^t_i$ along the targeted direction and $v^p_i$ that is perpendicular to $v^t_i$.

$$v_i = v_{ij} \cdot \frac{x_{ij}}{|x_{ij}|} \quad \text{and} \quad v^p_i = v_i - v^t_i \quad (3)$$

These two vectors can then be adjusted according to the user requirement:

$$\hat{v}^t_i = (1-k_t)v^t_i \quad \text{and} \quad \hat{v}^p_i = (1-k_p)v^p_i \quad (4)$$

where $k_t$ and $k_p$ represent the damping coefficients. A larger $k_t$ moves the fluid more steadily, and a larger $k_p$ drives the fluid more directly toward the target. The damping coefficients range from zero to one. The velocity of particle $i$ is obtained by combining $\hat{v}^t_i$ and $\hat{v}^p_i$.

### 3.5 Building and Updating Correspondences

As the simulation proceeds, the correspondences between the fluid and target particles are regularly updated based on an optimization process, similar to the work of Madill and Mould [12]. Figure 4 shows the correspondence update; the blue and red colors represent the fluid and target particles’ positions, respectively. The original correspondences (gray dotted lines) are updated if the newly built correspondences (black dotted lines) decrease the distances between the fluid and target particles.

### 4 Experimental Results and Discussion

Fig. 2 shows an animation that controls the fluid using multiple target shapes: box, bunny, and duck. The number of fluid particles for the initial water ball is 562K, and the numbers of target particles for the box, bunny, and duck are 439K, 248K, and 216K, respectively. The average computation time for control was 4089.1 ms per step, which was 10.95% of total time.

Fig. 3 demonstrates both shape and path controls. A water ball, moves along the path and passes through a water ring. In this experiment, the numbers of target particles for the ring and ball are 51K and 24K, respectively. The average computation time for control was 2280.4 ms per step, which was 7.27% of total time.

Fig. 5 shows another example for both the shape and path controls; two liquid balls are moved along the paths.
ated and collided following the paths. This experiment uses 141K target particles. The average computation time for control was 3112.8 ms per step, which was 9.27% of total time.

5 Conclusion and Future Work

This study proposed an effective target-driven fluid control simulation framework. In the proposed framework, the fluid and target were represented as particle primitives; the control mechanism was designed and implemented through the correspondences between the fluid and target particles. The proposed control method was smoothly integrated into the underlying fluid simulation method, FLIP. The experimental results demonstrated that the proposed framework appropriately produces the controlled animation using a simple user-defined scenario.

Although the proposed method effectively produces desired animation, there are still challenging areas in fluid control. For example, if a large-scale animation is desired, the user’s intervention in animation becomes difficult. The guiding method [7] can be used to overcome this difficulty. This will be a future research topic.

Acknowledgements

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (No. NRF-2016R1A2B3014319).

References