Received: 30 June 2016 | Revised: 22 June 2017 | Accepted: 16 July 2017

DOI: 10.1002/cav.1788

RESEARCH ARTICLE

WILEY

Position-based simulation of cloth wetting phenomena

Xuqiang Shao[1,2] | Wei Wu[2] | Baoyi Wang[1]

[1]School of Control and Computer Engineering, North China Electric Power University, Baoding 071003, China

[2]State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China

Correspondence
Xuqiang Shao, School of Control and Computer Engineering, North China Electric Power University, Baoding 071003, China; or State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, Beijing 100191, China.
Email: shaoxuqiang@gmail.com

Funding information
National Natural Science Foundation of China, Grant/Award Number: 61502168 and 61472020; Natural Science Foundation of Hebei Province, Grant/Award Number: F2016502069; Open Fund of the State Key Laboratory of Virtual Reality Technology and Systems, Grant/Award Number: BUAA-VR-16KF-03; National 863 Program of China, Grant/Award Number: 2015AA016403

Abstract

Because of the advantages of high stability and efficiency, position-based dynamics (PBD) is becoming increasingly popular with game developers and filmmakers. On the basis of the unified adaptive PBD, we present a nonlinear cloth wetting model to simulate visually realistic cloth wetting phenomena, which takes into account the influence of gravity on all aspects of the wetting process. Specifically, in order to model water diffusion in unsaturated cloth, we propose a novel nonlinear saturation constraint of cloth particles, which considers the influence of gravity, and then solve it using PBD to stably smooth the saturation field. The dynamic behaviors of the cloth and the water–cloth coupling during the cloth wetting process are modeled by using PBD. Moreover, a water absorption model and a water emission model are proposed to simulate the unsaturated cloth absorbing water and the oversaturated cloth draining water under the influence of gravity, respectively. The experimental results demonstrate that our novel wetting model provides an efficient way to model more stable and realistic cloth wetting phenomena using PBD.

KEYWORDS

cloth wetting, particle-based simulation, position-based dynamics, water–cloth coupling

1 | INTRODUCTION

Physically based simulations of natural phenomena have been widely used for computer graphics applications, such as commercial films and computer games. Among various natural phenomena, the wetting of porous materials, which often happens in water–solid coupling is one type of very complicated physical dynamics behavior. Recent work has made great progress in visually plausible wetting animations.[1,2] However, due to small time steps remaining a requirement for the underlying physically based solvers, the stability and efficiency of the wetting simulation still need to be improved substantially, especially for the porous thin structures like cloth.[3–5]

By formulating and solving a set of positional constraints, PBD-based solvers[6,7] omit the velocity layer and immediately work on the positions. So, overshooting problems of explicit integration schemes, which restricts the time step in force-based systems,[8,9] can be avoided. Due to the advantage of high stability and efficiency, PBD is becoming increasingly popular among game developers and filmmakers. Up to now, PBD has been successfully employed to model phenomena such as rigid objects,[10] elastic objects,[11,12] cloth,[13] incompressible fluids,[14] and fluid–solid coupling.[15] However, visually realistic simulations of cloth wetting phenomena for virtual humans are still not solved by using PBD.

In order to realize a more stable and efficient simulation of cloth wetting phenomena, we put forward a nonlinear cloth wetting model based on unified adaptive PBD, which takes fully into account the influence of gravity on all aspects of the wetting process. Specifically, to stably model the crucial water diffusion process of cloth wetting, we propose a novel

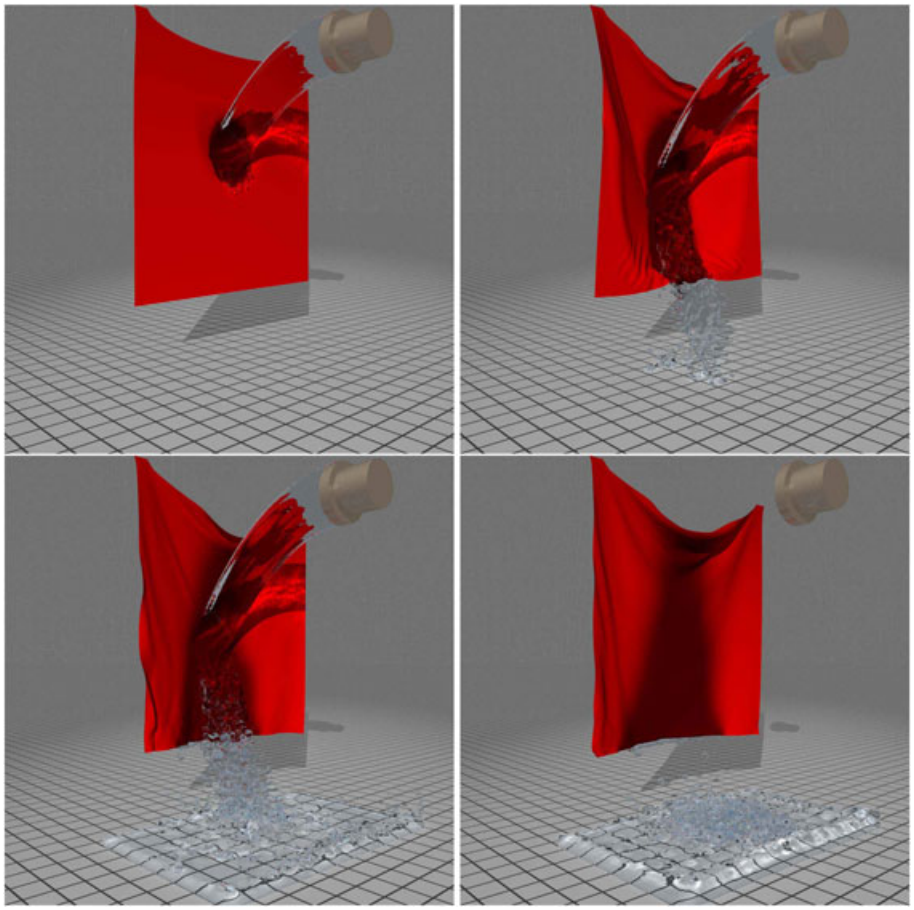


FIGURE 1 The cloth gets wet as it is hit by a water jet

nonlinear saturation constraint of cloth particles, which considers the influence of gravity, and then solve it using PBD to smooth the saturation field. For improving the visual realism, we simulate the dynamic behaviors of the cloth and the water–cloth coupling during the cloth wetting process in the unified adaptive PBD framework. In addition, a water absorption model and a water emission model are proposed to simulate the unsaturated cloth absorbing water and the oversaturated cloth losing water under the influence of gravity, respectively.

As the results shown in Figure 1, our novel wetting model provides an effective way to model realistic and stable cloth wetting phenomena in the unified PBD framework and can be easily incorporated into existing particle-based water–cloth coupling solvers.

2 | RELATED WORK

2.1 | Position-based dynamics

By omitting the velocity and acceleration layer and immediately working on the positions, PBD performs well at high stability and efficiency and is widely used to simulate various materials and phenomena in computer animation. Designing and solving a vast number of robust positional constraints between rigid bodies, Deul et al.[10] proposed a position-based approach for large-scale simulations of rigid bodies at interactive frame rates. Bender et al.[11] presented a fast and robust position-based method for the simulation of continuous deformable solids, which combines continuum mechanical material models with a position-based energy reduction. By introducing ghost points, which are additional points defined on edges, Umetani et al.[12] presented a novel PBD method to simulate complex bending and twisting of elastic rods in real time. To allow realistic cloth animation effects, Müller et al.[13] proposed a new set of constraints within the PBD framework that allow the control of strain in directions that are independent of the edge directions of the simulation mesh. By formulating and solving a set of positional constraints that enforce constant density, Macklin et al.[14] put forward a position-based method to simulate incompressible fluids, which allows large time steps suitable for real-time applications.

Macklin et al.[15] presented a unified PBD framework, which employs particles connected by constraints as the fundamental building block, to model real-time gases, liquids, deformable solids, rigid bodies, and cloth with two-way interactions. In addition, position-based facial animation synthesis[16] and skinning[17] also have been researched.

To further improve the efficiency of the PBD method, Müller[18] proposed a multigrid-based process to speed up the convergence of PBD significantly, while keeping the power of the method to process general nonlinear constraints. Fratarcangeli et al.[19] introduced a practical partitioning technique designed for parallelizing PBD, and exploiting the ubiquitous multicore processors present in current commodity graphics processing units (GPUs).

2.2 | Wetting simulation

In computer graphics, great achievements have been realized in visually plausible wetting animations. Chu et al.[20] presented a GPU-based method to model the dispersion of ink in absorbent paper based on the lattice-Boltzmann equation. In order to realize real-time sand wetting simulation, Rungjiratananon et al.[21] presented a simple particle-based method to model the physical mechanism of wetness propagating through granular materials on GPUs. To simulate wet hair, Rungjiratananon et al.[22] introduced a Cartesian grid to capture the microscopic porosity of hair to simulate water absorption and diffusion, cohesion of wet hair strands, water flow within the hair volume, and water dripping from the wet hair strands. Lin[23] sampled several hair boundary particles on each rod segment for hair strands and simulated Smoothed Particle Hydrodynamics (SPH) water absorption inside each hair strand with a diffusion process by treating these boundary particles as porous particles with anisotropic permeability. The saturation of each boundary particle is used to derive the fluid–hair coupling force and the adhesive force between wet hair strands. To further improve stability, Lin incorporated the saturation of each boundary particle into the density computation of the fluid particles at the boundary in the follow-up studies.[24] By improving the widely used technique of employing stationary boundary particles to satisfy nonpenetration and no-slip conditions, Bayraktar et al.[25] proposed an SPH-based method of fluid simulation, where boundary conditions are designed in such a way that fluid flow through porous media, pipes, and chokes can be realistically simulated. On the basis of Darcy's law, which describes the flow of water in porous materials, several researchers have realistically simulated the wetting phenomena in grid- or particle-based fluid solvers. Andrysco et al.[26] seamlessly integrated Darcy's law into semi-Lagrangian-based solvers, which enables fluid simulations to have effects such as water flowing through soil or a towel soaking up spilled liquid on a kitchen counter. In addition, Lenaerts et al.[1] treated the pores at a macroscopic scale and reused the particle representation of the deformable objects to model the diffusion process within porous materials. Afterward, their work was extended to simulate porous flow through granular materials and to model the behavior of the resulting mixture in a unified SPH framework.[27]

As for the cloth wetting simulation discussed in this paper, Huber et al.[4,28] implemented the simulation of the saturation process by using Fick's law to compute the diffusion, and a 2D cellular automaton is applied to update the diffusion state and to simulate the process of diffusion. Chen et al.[3] presented a technique that simulates wet garments for virtual humans with realistic folds and wrinkles. In contrast to previous approaches to wet cloth, the proposed models are supported by the experimental results reported in the textile literature, with parameters varying over the course of the simulation. Patkar et al.[2] presented a simple three-stage method to simulate the mechanics of wetting of porous objects such as sponges and cloth, which is motivated by the physics of imbibitions of fluids into porous solids in the presence of gravity. Shi et al.[29] combined repulsion and absorption with a continuous collision detection technology in the bidirectional interaction between fluids and cloth. As for fluid dynamic diffusion equilibrium, surface flow inside porous media and between materials was modeled, so that a variety of phenomena can be simulated. Güdükbay et al.[5] presented a particle-based method to simulate and visualize the interaction of spring–mass-based knitwear with SPH fluids on GPUs. Two-way coupling is achieved by considering surface tension, capillary, and interparticle forces between the fluid and knitwear.

3 | UNIFIED PBD FRAMEWORK

3.1 | PBD algorithm

In contrast to general force-based animation methods,[8,9] which are unstable under larger time steps, the PBD method[6] directly manipulates particle positions by solving a system of nonlinear equality or inequality constraints such as

$$C_i(\mathbf{x}) = 0 \quad \text{or} \quad C_j(\mathbf{x}) \geq 0, \quad i,j \in [1, \ldots, m], \tag{1}$$

where $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n]^T$ is the vector of particle positions and n and m are the number of particles and constraints in the system, respectively. The strength of each constraint is defined by the stiffness parameter $\kappa^c \in [0, \ldots, 1]$.

To guarantee the positions of all particles satisfying each constraint, PBD needs to find a correction vector $\Delta\mathbf{x} = [\Delta\mathbf{x}_1, \Delta\mathbf{x}_2, \ldots, \Delta\mathbf{x}_n]^T$ such that $C(\mathbf{x} + \Delta\mathbf{x}) = 0$. This equation can be approximated by

$$C_i(\mathbf{x} + \Delta\mathbf{x}) \approx C_i(\mathbf{x}) + \nabla C_i(\mathbf{x}) \cdot \Delta\mathbf{x} = 0. \tag{2}$$

The correction vector $\Delta\mathbf{x}$ is restricted to be in the direction of $\nabla C_i(\mathbf{x})$ to enforce the conservation of linear and angular momentum[6] as follows:

$$\Delta\mathbf{x} = \lambda_i \nabla C_i(\mathbf{x}). \tag{3}$$

Substituting Equation 3 into Equation 2 and solving for λ_i and substituting it back into Equation 3 yield the final formula for $\Delta\mathbf{x}$ as follows:

$$\Delta\mathbf{x} = \frac{C_i(\mathbf{x})}{|\nabla C_i(\mathbf{x})|^2} \nabla C_i(\mathbf{x}), \tag{4}$$

where $\Delta\mathbf{x}$ is the same for all particles. If the particles have individual masses, the PBD method weights the corrections $\Delta\mathbf{x}_j$ by the inverse masses $w_j = 1/m_j$ for each particle j.[6]

Positions are updated by solving each constraint independently one after the other, and after several iterations, the velocity is computed as $\mathbf{v} = \frac{\Delta\mathbf{x}}{\Delta t}$. To consider the stiffness of the constraint, PBD multiplies the correction vector $\Delta\mathbf{x}$ by $1 - (1 - \kappa^c)^{1/n_s}$ in each iteration[6] as follows:

$$\Delta\mathbf{x} = \left(1 - (1 - \kappa^c)^{1/n_s}\right) \Delta\mathbf{x}, \tag{5}$$

where n_s is the iteration number and κ^c is the user-defined stiffness parameter.

3.2 | Water and cloth simulation

We simulate water using the position-based fluid method,[14] where the density constraint on every water particle i is defined using an equation of state as follows:

$$C_i^{\text{density}}(\mathbf{x}) = \frac{\rho_i}{\rho_0} - 1, \tag{6}$$

where ρ_0 is the rest water density and ρ_i is calculated by the standard SPH density estimator[9] as follows:

$$\rho_i = \sum_j m_j W(\mathbf{x}_i - \mathbf{x}_j, h), \tag{7}$$

where $W_{ij} = W(\mathbf{x}_i - \mathbf{x}_j, h)$ denotes a Gaussian-like kernel with finite support radius h.

Our position-based cloth solver accepts arbitrary triangle meshes as input. The only restriction imposed on the meshes is that each edge is shared by at most two triangles. Each vertex of the mesh is selected as a simulated particle of the PBD method. On the basis of the cloth simulation approach by Müller et al.,[6] we iteratively solve two equality constraints C^{stretch} and C^{bend} to simulate cloth. For each edge $(\mathbf{x}_1, \mathbf{x}_2)$, we define a stretching constraint imposed on it as follows:

$$C^{\text{stretch}}(\mathbf{x}_1, \mathbf{x}_2) = |\mathbf{x}_1 - \mathbf{x}_2| - l_0, \tag{8}$$

where l_0 is the initial length of the edge. For each pair of adjacent triangles $(\mathbf{x}_1, \mathbf{x}_3, \mathbf{x}_2)$ and $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_4)$ sharing the edge $(\mathbf{x}_1, \mathbf{x}_2)$, we generate a bending constraint function

$$C^{\text{bend}}(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{x}_4) = \text{acos}\left(\frac{(\mathbf{x}_2 - \mathbf{x}_1) \times (\mathbf{x}_3 - \mathbf{x}_1)}{|(\mathbf{x}_2 - \mathbf{x}_1) \times (\mathbf{x}_3 - \mathbf{x}_1)|} \cdot \frac{(\mathbf{x}_2 - \mathbf{x}_1) \times (\mathbf{x}_4 - \mathbf{x}_1)}{|(\mathbf{x}_2 - \mathbf{x}_1) \times (\mathbf{x}_4 - \mathbf{x}_1)|}\right) - \varphi_0, \tag{9}$$

where the scalar φ_0 is the initial dihedral angle between two triangles.

4 | NONLINEAR CLOTH WETTING MODEL

According to Darcy's law,[1] the wetting of porous cloth is a complex phenomenon caused by capillary forces and gravity. In order to simulate this phenomenon in the unified PBD framework at a macroscopic scale, we remodel the cloth simulation particles as porous particles to hold an amount of water. Then, a nonlinear wetting model, which takes into account the influence of gravity on water absorption, diffusion, and emission, is proposed to update the saturations of cloth particles to visually realistically simulate cloth wetting.

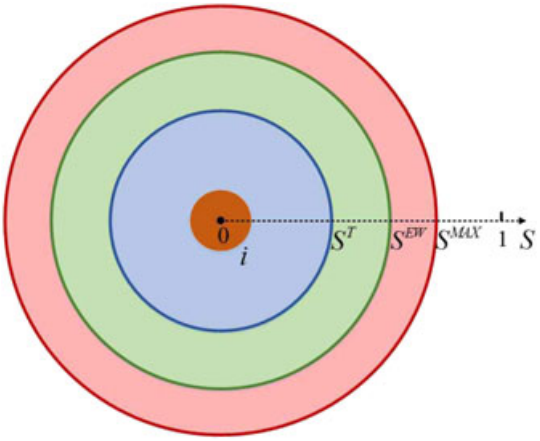


FIGURE 2 Porosity modeling of cloth particle

4.1 | Porosity modeling

Cloth is a typical type of porous media. To model the cloth wetting process, for a cloth particle i, we first define its saturation as the mass of water that the particle holds per unit volume, as follows:

$$S_i = \frac{cm^{\text{water}}}{V_i}, \tag{10}$$

where m^{water} denotes the mass of water that particle i holds, c is a user-defined coefficient to guarantee $S_i \in (0, 1)$, $V_i = \frac{1}{\sum_k W_{ik}}$ is the particle volume based on the work of Akinci et al.,[8] and k is the index of the neighboring cloth particle belonging to the same piece of cloth. In order to realistically simulate the behaviors of the wet cloth, as shown in Figure 2, we define three threshold values for the saturation of each cloth particle similar to those in the work of Rungjiratananon et al.[21] S^{MAX} is the maximum saturation for the cloth material, depending on the maximum mass of water it can hold per volume. If the saturation S_i is greater than S^{EW}, cloth particle i will emit water (see Section 4.4). $S^T = 0.3S^{\text{MAX}}$ is a threshold value used to simulate the dynamic behavior of the wet cloth (see Section 4.5).

4.2 | Water absorption

We propose a novel absorption model, which takes into account the influence of gravity, to compute the amount of water absorbed by a cloth particle. For a water particle j, the maximum mass of water that can be absorbed by its neighboring cloth particle i is calculated by

$$m_i^{\text{absorbed}} = \kappa_i^a \left(1.0 + \eta^a \cos\theta_{ij}\right) \left(S^{MAX} - S_i\right) V_i, \tag{11}$$

where the user-defined coefficient κ^a controls the rate at which the cloth material absorbs water and $\theta_{ij} \in [0, \pi]$ is the angle that the vector $\mathbf{x}_{ij} = \mathbf{x}_i - \mathbf{x}_j$ makes with the direction vector of gravity. In our absorption model, the nonlinear function $f(\theta_{ij}) = 1.0 + \eta^a \cos\theta_{ij}$ calculates the influence of gravity on the absorbent ability of the cloth, where $\cos\theta_{ij}$ denotes that a cloth particle below fluid particles absorbs more water than the one above fluid particles and $\eta^a \in [0, 1]$ is a user-defined coefficient to control the proportion of gravity influence.

Then, we calculate the actual water mass of j absorbed by i according to the relationship between m_i^{absorbed} and m_j. If $m_i^{\text{absorbed}} \geq m_j$, it indicates that the water particle j is entirely absorbed by the cloth particle i. In this case, the water particle j should be removed from the PBD framework, and the saturation and rest density of i are updated according to the actual absorbed fluid mass m_j as $S_i += \frac{cm_j}{V_i}$ and $\rho_i^0 *= \frac{m_j + m_i}{m_i}$, respectively; if $m_i^{\text{absorbed}} < m_j$, only a portion of j is absorbed by i. In our unified PBD framework supporting the adaptive particle sampling scheme,[30] j becomes a smaller water particle with mass $m_j' = m_j - m_i^{\text{absorbed}}$ and support radius $h_j' = (\frac{m_j'}{m_j})^{\frac{1}{3}} h_j$, while i updates its saturation and rest density as $S_i += \frac{cm_i^{\text{absorbed}}}{V_i}$ and $\rho_i^0 *= \frac{m_i^{\text{absorbed}} + m_i}{m_i}$, respectively.

4.3 | Water diffusion

We model the transport of absorbed water inside the cloth as a diffusion process of saturation and thus simulate the water diffusion phenomenon by smoothing the saturation field defined on cloth particles. In order to obtain stable and realistic results under larger time steps, we propose to smooth the saturation field by using PBD to solve a new nonlinear saturation constraint of cloth particles, which calculates the diffusion rate by considering not only the saturation difference but also the gravity influence.

First, we define a novel saturation field constraint for each cloth particle i based on Fick's law[28] and then discretize it using the SPH method,[9] as follows:

$$\begin{aligned} C_i(\mathbf{S}) &= \kappa^d f(\theta) \nabla^2 S_i \\ &= \sum_j \frac{\kappa_i^d + \kappa_j^d}{2} \frac{S_j - S_i}{\rho_0} \left(1.0 + \eta^d \cos\theta_{ij}\right) \nabla^2 W_{ij}, \end{aligned} \tag{12}$$

where $\mathbf{S}$ is the vector of saturation values of all particles belonging to a piece of cloth, j is the index of neighboring cloth particles of i, and κ^d defines the saturation diffusion rate of the cloth material. Because the cloth material is anisotropic, the value of κ^d for each cloth particle is different. For simplicity, we assume that the cloth material is isotropic in this paper. The nonlinear function $f(\theta)$ calculates the influence of gravity on saturation diffusion, where θ_{ij} is the angle that the vector $\mathbf{x}_{ij} = \mathbf{x}_i - \mathbf{x}_j$ makes with the gravity direction vector and η^d is a user-defined coefficient to weigh the gravity influence similar to that in Equation 12.

The aforementioned saturation field constraint equaling to 0 means that the saturation near cloth particle i is smooth. If we apply the constraint to all cloth particles, the saturation in the cloth domain is finally smoothed, and thus, the cloth wetting phenomena can be simulated. Based on PBD,[6] the change of saturation field represented by $\Delta\mathbf{S}$ is needed to satisfy Equation 2 as follows:

$$C_i(\mathbf{S} + \Delta\mathbf{S}) \approx C_i(\mathbf{S}) + \Delta\mathbf{S} \cdot \nabla_{\mathbf{S}} C_i(\mathbf{S}) = 0. \tag{13}$$

Assuming that $\Delta\mathbf{S}$ is along with the derivative of the constraint $\nabla_{\mathbf{S}} C_i(\mathbf{S})$, we get its estimation

$$\Delta\mathbf{S} = \lambda_i \nabla_{\mathbf{S}} C_i(\mathbf{S}). \tag{14}$$

Then, substituting Equation 14 into Equation 13, we have

$$\lambda_i = -\frac{C_i(\mathbf{S})}{\sum_k \left|\nabla_{S_k} C_i(\mathbf{S})\right|^2 + \varepsilon}, \tag{15}$$

where ε is a user-defined constant that avoids the divisor too small to keep the simulation stable. The derivative of the constraint function $C_i(\mathbf{S})$ with respect to particle k is given by

$$\nabla_{S_k} C_i(\mathbf{S}) = \frac{1}{\rho_0} \begin{cases} \sum_j -\frac{\kappa_i^d + \kappa_j^d}{2} \left(1.0 + \eta^d \cos\theta_{ij}\right) \nabla^2 W_{ij}, & k = i, \\ \frac{\kappa_i^d + \kappa_k^d}{2} \left(1.0 + \eta^d \cos\theta_{ik}\right) \nabla^2 W_{ik}, & k \neq i. \end{cases} \tag{16}$$

Finally, the saturation of particle i is updated as

$$S_i^* = S_i + \kappa^s \lambda_i \nabla_{S_i} C_i(\mathbf{S}), \tag{17}$$

where S_i^* is the updated saturation value and $\kappa^s \in (0, 1)$ controls the smoothness of saturation field.

4.4 | Water emission

When the saturation of the cloth exceeds a certain threshold and the influence of gravity overcomes the saturation difference, the excessive water will gather toward the bottom of the cloth. The gathered water tends to adhere to the cloth and not drip until the excess amount increases beyond a certain mass threshold. This phenomenon is called water emission in this paper. To realistically simulate water emission, we present a new water emission model suitable to the PBD framework based on the method of Patkar et al.[2]

First, we simulate the gathering of the excessive water toward the bottom of the cloth under the influence of gravity. Each cloth particle i maintains a water buffer, and the amount of excessive water stored in the buffer is

$EW_i^{\text{buffer}} = \max(0, (S_i - S^{\text{EW}})V_i)$. In our method, if the following two conditions are both satisfied, EW_i^{buffer} will be transferred to the water buffers of neighboring cloth particles below i:

$$\begin{cases} EW_i^{\text{buffer}} > 0, \\ \left| \sum_j V_j S_j \cos\theta_{ij} \nabla W_{ij} \right| < |\beta \mathbf{g}|, \end{cases} \tag{18}$$

where j is the index of the neighboring cloth particles above i, β is a user-defined coefficient, and $\mathbf{g}$ is the gravitational acceleration vector. The satisfaction of the second condition denotes that the influence of gravity overcomes the saturation difference of i with its neighbors above. The amount of water that is transferred from i to its neighboring particles is $\cos\theta_{ki} EW_i^{\text{buffer}} W_{ki}$, where k is the index of the neighboring cloth particles below i.

Second, we generate new water particles if the amount of excessive water in the buffer exceeds a threshold and then compute the interaction of new water particles with cloth particles. For a cloth particle i, if $EW_i^{\text{buffer}} \geq m^{\text{threshold}}$, a new water particle p with mass $m^{\text{threshold}}$ and support radius $h_p = (\frac{m_p}{m_0})^{\frac{1}{3}} h_0$ is generated to simulate water drip formation. h_0 is the initial support radius of water particles. In order to avoid instability induced by very large local density, the position of p is calculated according to Lin's approach.[24] Then, the saturation-dependent attraction force exerted on p by its neighboring cloth particles is calculated from the following equation to simulate the flows of water drip on the cloth surface:

$$\mathbf{f}_p^{\text{attract}E} = \sum_i \kappa^e \left(m_i + \eta^e S_i \right) \Psi(\rho_0) W(\mathbf{x}_{ip}, h_p), \tag{19}$$

where κ^e is a coefficient to adjust the attraction force, $\Psi(\rho_0)$ is the function in the work of Akinci et al.[8] to consider the relative contribution of neighboring cloth particles, and η^e is a user-defined coefficient to weigh the saturation influence.

4.5 | Wetting dynamics

The dynamic behavior of the cloth and the water–cloth coupling during the wetting process is called wetting dynamics in this paper. To improve the simulation realism, we propose to model two types of wetting dynamics phenomena in PBD: the sticking of wet cloth and the penetration of water into the wet cloth.

The presence of water stored in wet cloth particles induces the cloth to stick together. When this becomes too wet, the cloth separates again. To simulate this phenomenon, we present a saturation-related attractive force model for contacted wet cloth particles. For a cloth particle i, the attractive force exerted by the neighboring cloth particle j is

$$\mathbf{f}_i^{\text{attract}D} = \left(-\kappa^{ad} \left| (S_i - S^T)(S_j - S^T) \right| + \eta^{ad} \right) \cdot W(\mathbf{x}_j - \mathbf{x}_i, h), \tag{20}$$

where κ^{ad} and η^{ad} are the user-defined coefficients that make the force magnitude $\left| \mathbf{f}_i^{\text{attract}D} \right|$ greater than zero when the saturation of particle changes in $(0, S^{\text{MAX}})$. The saturation-related attractive force gets its maximal value at S^T.

If the wetting process is not under consideration, the two-way coupling of cloth and water should avoid penetration artifacts. We adopt the constraint function that is used for solving the self-collision of cloth in the work of Müller et al.[6] to simulate the two-way coupling of water and cloth. If a particle q of water moves through a triangle $(\mathbf{x}_a, \mathbf{x}_b, \mathbf{x}_c)$ of cloth mesh, we define a coupling constraint with constraint function

$$C^{\text{couple}}(\mathbf{x}_q, \mathbf{x}_a, \mathbf{x}_b, \mathbf{x}_c) = \pm(\mathbf{x}_q - \mathbf{x}_1) \cdot \frac{(\mathbf{x}_b - \mathbf{x}_q) \times (\mathbf{x}_c - \mathbf{x}_q)}{|(\mathbf{x}_b - \mathbf{x}_1) \times (\mathbf{x}_c - \mathbf{x}_1)|} - d \tag{21}$$

and stiffness κ^{couple}. d is the cloth thickness. If the water particle q penetrates the triangle $(\mathbf{x}_a, \mathbf{x}_b, \mathbf{x}_c)$ from below with respect to the triangle normal, the first term of the right-hand side of Equation 21 picks the minus. Conversely, the term picks the plus. κ^{couple} is equal to 1 in the coupling if we do not take into account the wetting phenomena.

Through actual observation of real wetting phenomena, we find that the cloth is more easily penetrated by the water when getting wetter and wetter. For a triangle $(\mathbf{x}_a, \mathbf{x}_b, \mathbf{x}_c)$ of cloth mesh, we propose to define the stiffness κ^{couple} of the coupling constraint formulated by Equation 21 according to its saturation, as follows:

$$\kappa^{\text{couple}} = 1.0 - \frac{\tilde{S}_a + \tilde{S}_b + \tilde{S}_c}{3(S^{\text{MAX}} - S^T)}, \tag{22}$$

where $\tilde{S}_i = \max(0.0, (S_i - S^T))$. According to Equation 5, by multiplying the correction vector $\Delta\mathbf{x}$ by $1 - (1 - \kappa^{\text{couple}})^{\frac{1}{n_s}}$ in each iteration of PBD, we can simulate the penetration of water into the wet cloth.

5 | IMPLEMENTATION DETAILS

The proposed cloth wetting method is entirely implemented on the GPU using CUDA for interactive frame rates. Algorithm 1 presents the pseudocode that summarizes the crucial steps of the GPU implementation pipeline. By following these steps, the cloth wetting method can be added to various particle-based water–cloth coupling solvers easily.

Algorithm 1 Simulation loop on the GPU

while *animating* **do**
  foreach *particle i* **do**
    $\mathbf{v}_i \Leftarrow \mathbf{v}_i + \Delta t \mathbf{f}^{ext}$;
    $\mathbf{x}_i^* \Leftarrow \mathbf{x}_i + \Delta t \mathbf{v}_i$;
  foreach *particle i* **do**
    Find neighboring particles according to $\mathbf{x}_i^*$;
  foreach *water particle i* **do**
    Compute water dynamics by solving Equation 7 based on PBD;
  foreach *cloth particle i* **do**
    Compute cloth dynamics by solving Equation 8 and 9 based on PBD;
  foreach *cloth particle i* **do**
    Update particle volume and saturation;
  foreach *water particle i* **do**
    Find neighboring triangles and compute $\kappa^{coupling}$ using Equation 22; Compute water-cloth coupling by solving Equation 10 based on PBD;
  foreach *water particle i* **do**
    Compute the absorbed water using Equation 12; Update property values of *i* and neighboring cloth particles;
  foreach *cloth particle i* **do**
    Compute water diffusion by solving Equation 13 based on
  foreach *cloth particle i* **do**
    Compute the transfer of excessive water if Equation 19 is satisfied; Generate new water particles and adjust their positions;
  foreach *particle i* **do**
    Compute saturation-dependent attraction forces using Equation 20 and 21;
  Water surface construction and scene rendering;

To fully exploit the performance of the GPUs, all unchangeable physical values of particles, such as rest particle positions and initial neighboring lists of cloth particles, are stored as textures, whereas the changeable physical values, such as saturations, positions, velocities, forces, λ values, and correction vectors, are stored in GPU global memory. The physical properties of all particles are stored in the same CUDA arrays, and an additional flag value *Flag* (fluid particle: 0; cloth particle: $1, \ldots, n$; n is the number of cloth objects) stored in the fourth element of *float*4-type vector variables is used to distinguish to which object each particle belongs.

To update the physical values of each particle, the search of neighboring particles is required, which is also the most time-consuming part in each time step. To speed up this task, we launch kernels to construct a k-dimensional (K-D) tree[31,32] of all particle positions predicted by explicit time integration (line 4) on the GPU and then traverse it to find the neighboring particles for each particle. Then, based on the PBD method,[6,14] the dynamics of incompressible fluid and cloth are simulated on GPUs using CUDA according to the constraints formulated by Equations 6, 8, and 9. In SPH-based summation computation, W is the Poly6 kernel for density estimation and the Spiky kernel for gradient calculation.

The calculation process from line 11 to line 25 is the core of our wetting model. First, the saturation-dependent water–cloth coupling is simulated by solving Equations 21 and 22 based on PBD. Then, for each fluid particle, we compute the amount of water absorbed by neighboring cloth particles using Equation 11 and then update the physical properties of particles. By solving the constraint formulated by Equation 12 using PBD, the saturation diffusion among cloth particles is calculated. When the condition formulated by Equation 18 is satisfied, we compute the transfer of excessive water between cloth particles. Finally, we compute the saturation-dependent attractive forces between particles using Equations 19 and 20.

For the rendering of fluid, we adopt the GPU-based interactive rendering method of Goswami et al.[33] to define the distance field constructed from the extracted fluid surface particles. Then, the triangle meshes are extracted by using the GPU-accelerated marching cube technique[34] provided by the NVIDIA CUDA "Marching Cubes Isosurfaces" demo. The cell size for the Marching Cube method is set to $0.7r_0$.

6 | RESULTS

We have generated several cloth wetting animations to demonstrate the advantages of our novel wetting model. All testing cases have been implemented on a PC with an NVIDIA Geforce GTX 690 GPU, an Intel Xeon E5630 CPU using C++, CUDA 7.0, and OpenGL 4.0. The simulation scene is rendered off-line by the open-source Persistence of Vision Raytracer (POV-Ray) 3.7. The animations for all these cases are also available in Video S1.

The parameter values of the simulation are documented in Table 1. Our position-based cloth wetting method is stable with a time step $\Delta t = 0.015$, which is about 20 times larger than the force-based methods[1,2] in the same scene configuration.

Figure 3 is an animation scenario wherein a water cube drops on the surface of a rectangular piece of cloth whose corners are all constrained. Figure 3(a) shows the two-way coupling of water and cloth, which does not simulate the wetting phenomena. There are no penetrating artifacts existing in the coupling, especially when the cloth is under large deformations. As shown in Figure 3(b), our wetting model produces visually realistic cloth wetting simulations, which significantly improves the visual realism of the water–cloth coupling. The water is absorbed when colliding with the dry cloth and then is transported by the diffusion process. Finally, the water drips from the bottom of the overwet cloth. For cloth particles, the color transition from red to black illustrates that the saturation changes from small to large.

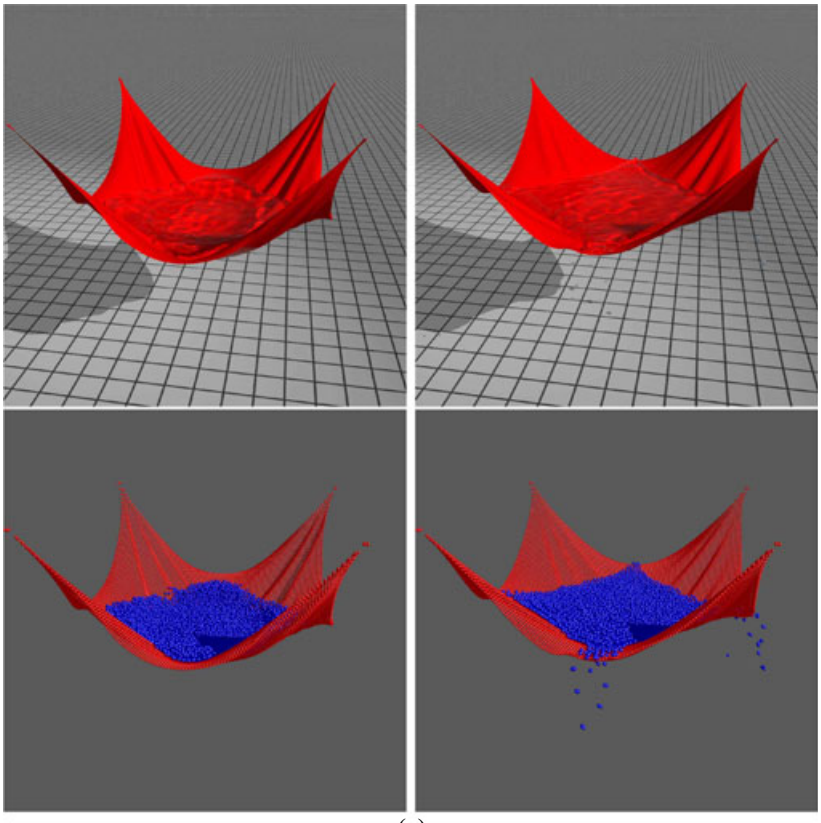
Figure 4 shows a 2D animation scenario wherein the green cloth particles absorb the water from the blue water particles under the influence of gravity. For cloth particles, the color transition from green to black illustrates the saturation changing from small to large. The results clearly demonstrate that a cloth particle below the fluid particles absorbs more water than the one above the fluid particles, because our water absorption model takes into account the gravity influence.

Figure 5 is a scenario wherein a piece of cloth held vertically from two corners becomes wet when coming into contact with the water below. Figure 5(a) shows the real cloth wetting phenomena captured by the camera. The saturations of real wet cloth are not evenly distributed, because the wetting diffusion happens only when the saturation difference plays a bigger role than the gravity. As shown in Figure 5(b), if the position-based wetting diffusion model does not consider the gravity influence, it will generate a very smooth saturation distribution. Figure 5(c) shows the wetting diffusion results simulated by the position-based wetting diffusion model taking into account the gravity influence. From the comparisons, we conclude that our wetting diffusion model considering the gravity influence can produce the wetting results close to the real data.

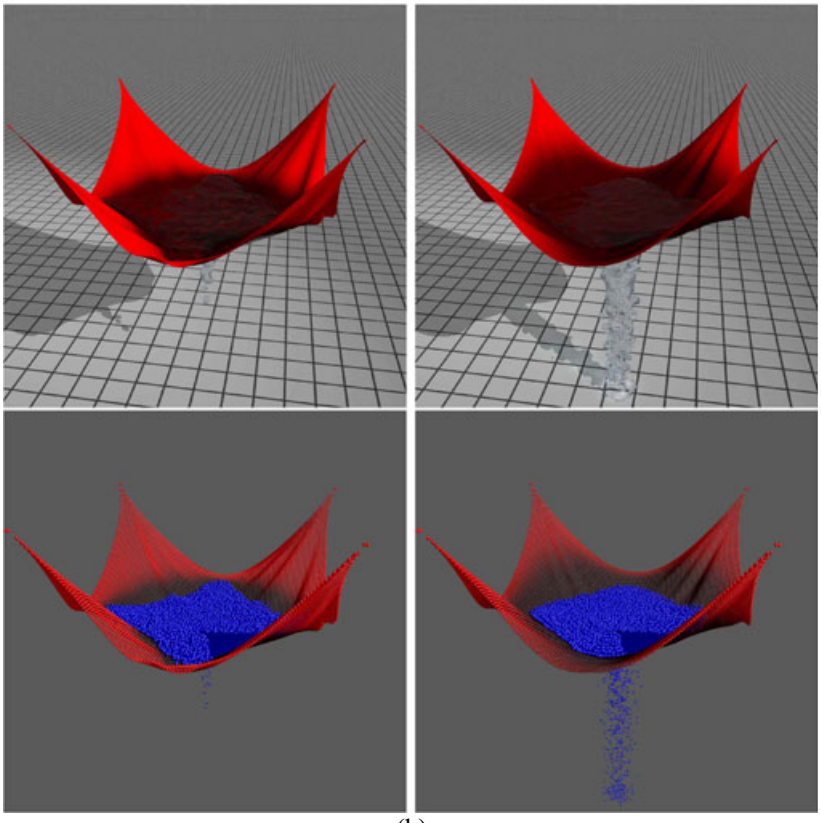
Figure 6 shows that the excessive water drips from a piece of fully saturated cloth held vertically from two corners. The excessive water stored in the cloth transfers toward the bottom of the cloth, under the influence of gravity, and forms

TABLE 1 Parameter values in the experiments

Properties	Values	Unit
Time step (Δt)	0.015	s
Initial spacing (r_0)	0.02	m
Support radius (h)	0.04	m
Initial density (ρ_0)	800 (cloth), 1000 (water)	kg/m^3
Saturation (S^*)	0.12 (S^T), 0.35 (S^{EW}), 0.4 (S^{MAX})	L
Particle mass (m)	0.0066 (cloth), 0.0082 (water)	kg
Mass threshold ($m^{\mathrm{threshold}}$)	0.002	kg
Absorption rate (κ^a)	0.1	L
Diffusion rate (κ^d)	0.5	L
Saturation smoothing rate (κ^s)	0.8	L
Attraction force coefficient (κ^*)	10.0 (κ^e), 0.3 (κ^{ad})	L
Gravity influence coefficient (η^*)	0.6 (η^a), 0.4 (η^d)	L



(a)



(b)

FIGURE 3 A water cube drops on a rectangular piece of cloth whose corners are all constrained. (a) Water–cloth coupling without wetting simulation. (b) Water–cloth coupling with wetting simulation

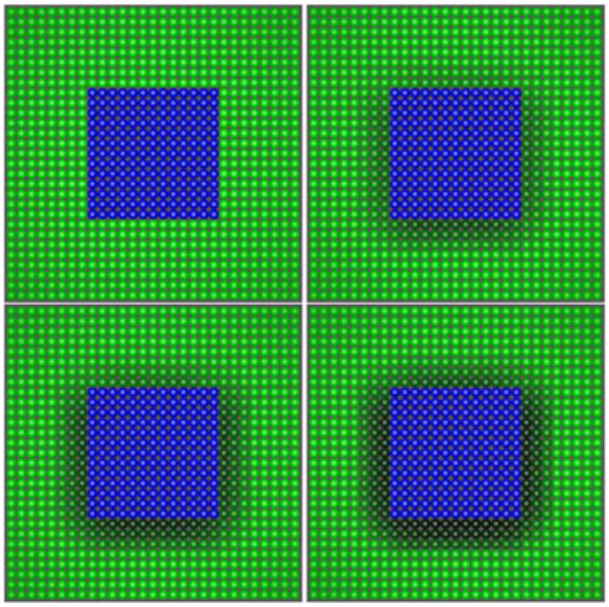


FIGURE 4 The simulation results of our water absorption model

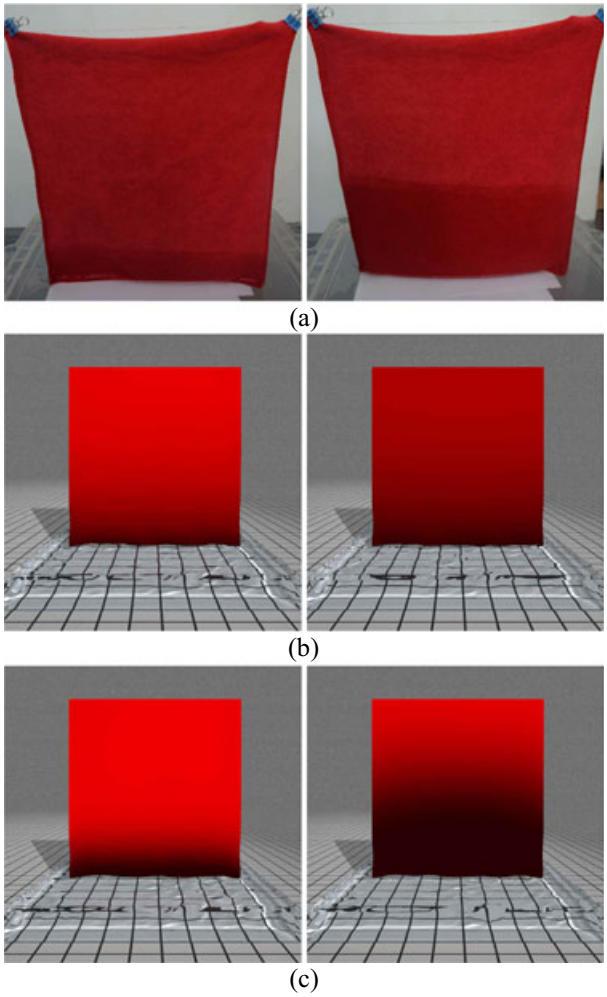


FIGURE 5 The simulation of the water diffusion process. (a) Real wetting diffusion phenomena. (b) Wetting diffusion without gravity influence. (c) Wetting diffusion with gravity influence

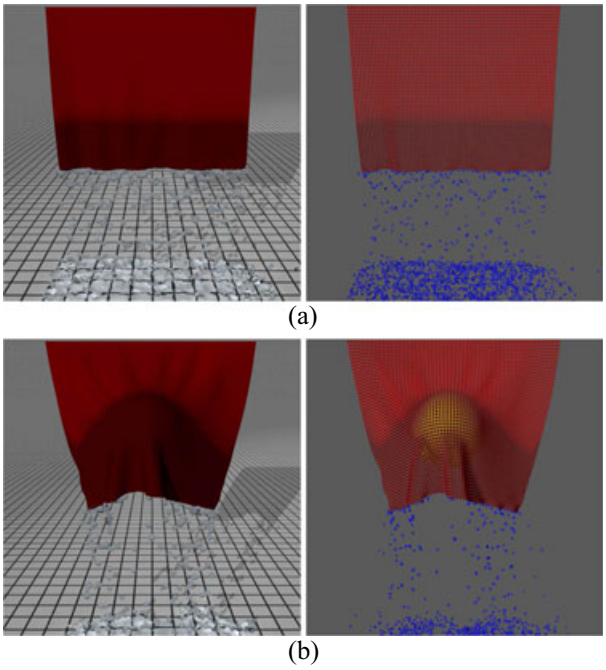


FIGURE 6 The simulation results of our water emission model. Water emission from fully saturated cloth with nearly uniform θ (a) and significantly different θ (b) in Equation 18

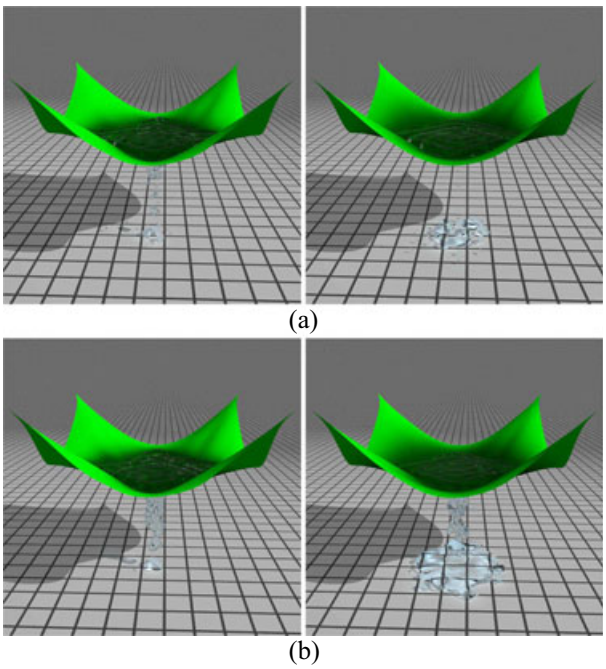


FIGURE 7 Comparison with Patkar's wetting method. (a) Patkar's wetting method. (b) Our wetting method

the water drips that adhere to the cloth edge. The water does not drip until the amount increases beyond a certain mass threshold. From the comparisons of two examples with different θ configurations, we conclude that the excessive water is transferred more quickly along the direction of gravity.

Figure 1 is a scenario wherein a water jet at the velocity of 4.0 m/s hits a piece of cloth held vertically from two corners. The cloth absorbs water from the water jet and then transfers the absorbed water from the wet region to the dry region. The excessive water sheds from the fully saturated region of the cloth and forms the water drips that adhere to the cloth edge. Finally, the water drip detaches from the bottom of the cloth when its mass is beyond a certain threshold.

In Figure 7, we compare our cloth wetting method with Patkar's wetting method,[2] which selects the triangle as the basic element to simulate the water absorption, diffusion, and dripping of the cloth. Both Patkar's method and our method can simulate the visually realistic water absorption and diffusion behaviors of the cloth. However, from the comparison of water emission, our method can generate a more realistic water dripping behavior of the wet cloth. Because our method reduces the stiffness of the coupling constraint according to the saturation level, the water drips more and more when the cloth saturation increases.

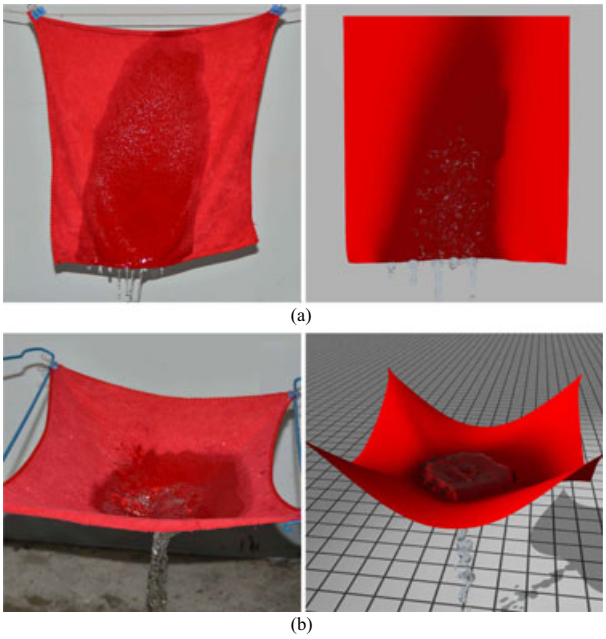


FIGURE 8 Visual comparisons of real and simulated cloth wetting phenomena. (a) A piece of wet cloth held vertically from two corners. (b) Water drops on a piece of horizontal cloth whose corners are all constrained

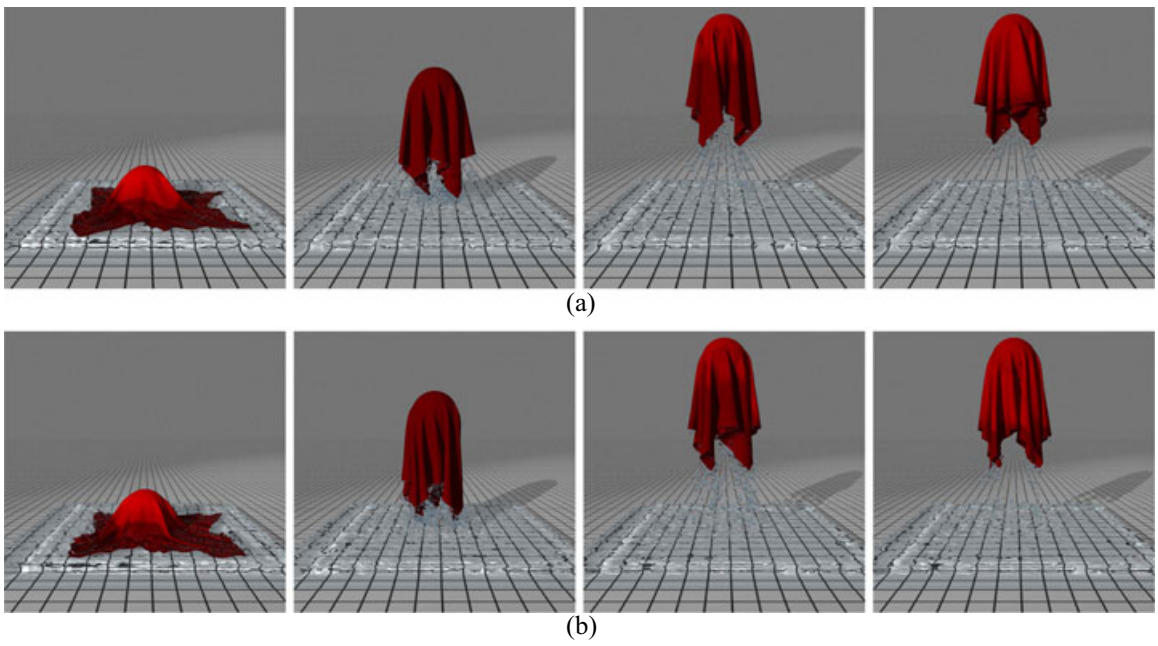


FIGURE 9 The wetting simulation of a piece of cloth wrapping a ball. (a) Wetting simulation without the sticking behaviors. (b) Wetting simulation with the sticking behaviors

TABLE 2 Time performance (in milliseconds) of our experiments

Scene	Water particles	Cloth particles	Wetting time	Total time
Figure 1	25.1k	14.4k	5.2	28.7
Figure 3(a)	13.8k	14.4k	0.0	21.6
Figure 3(b)	22.4k	14.4k	4.9	27.4
Figure 5(b)	50.0k	10.0k	5.6	42.5
Figure 5(c)	50.0k	10.0k	5.7	42.9
Figure 6(a)	11.3k	14.4k	4.5	20.2
Figure 6(b)	10.8k	14.4k	4.3	19.8
Figure 7	1.9k	2.5k	1.3	5.2
Figure 9(a)	85.2k	14.4k	15.2	67.8
Figure 9(b)	86.5k	14.4k	15.8	68.5

In Figure 8, we compare the output of some of our simulation experiments with that of simple real experiments. From the comparison results, we conclude that our method can simulate similar cloth wetting effects, including water absorption, diffusion, and emission, which indicates that our model for cloth wetting is a good approximation of the real world.

Figure 9 is a realistic animation scenario wherein a piece of dry cloth drops on a ball above the water and becomes wet when coming into contact with the water, then the water drips from the bottom of the fully saturated cloth when the cloth rises with the ball. Figure 9(a) shows the simulation results, which do not take into account the sticky behaviors of the wet cloth, whereas Figure 9(b) shows the sticky behaviors of the wet cloth simulated by our saturation-related attractive force model. From the comparison results, we conclude that our saturation-related attractive force model can simulate sticky behaviors and more wrinkles of the wet cloth.

Table 2 shows statistics for the average time cost in milliseconds of each simulation step on the GPU. Note that the rendering time is not included in the total time. The results demonstrate that our method can achieve the interactive simulation of cloth wetting effects. Even for nearly 100k particles, the frame rate of 14.6 FPS can be achieved. In addition, the time consumption of our wetting method is low when compared with the total time consumption, so the high efficiency is a remarkable advantage of our wetting method.

7 | CONCLUSION AND FUTURE WORK

To stably and realistically simulate the cloth wetting phenomena, we have presented a nonlinear cloth wetting model based on unified adaptive PBD, which takes into account the influence of gravity on all aspects of the wetting process, including water absorption, water diffusion, and water emission. In addition, the dynamic behaviors of the cloth and the water–cloth coupling during the cloth wetting process is also elaborately modeled. Various animation results demonstrate that stable and realistic cloth wetting phenomena have been realized in the unified PBD framework. Our cloth wetting method is inherently mass conserving and computationally lightweight and can be easily incorporated into existing particle-based solvers.

In future work, we would like to extend the cloth wetting model to handle more complicated cloth wetting phenomena, which will take into account the air influences and material properties of the cloth.

ACKNOWLEDGEMENTS

This work is supported by the National Natural Science Foundation of China under Grants 61502168 and 61472020, the Natural Science Foundation of Hebei Province under Grant F2016502069, the Open Fund of the State Key Laboratory of Virtual Reality Technology and Systems under Grant BUAA-VR-16KF-03, and the National 863 Program of China under Grant 2015AA016403. We also thank the anonymous reviewers for their constructive comments.

ORCID

Xuqiang Shao http://orcid.org/0000-0002-0385-4269

REFERENCES

1. Lenaerts T, Adams B, Dutré P. Porous flow in particle based fluid simulations. ACM Trans Graph. 2008;27(3):49:1–49:8.
2. Patkar S, Chaudhuri P. Wetting of porous solids. IEEE Trans Vis Comput Graph. 2013;19(9):1592–1604.
3. Chen Y, Magnenat-Thalmann N, Allen BF. Physical simulation of wet clothing for virtual humans. Vis Comput. 2012;28(6):765–774.
4. Huber M, Pabst S, Strasser W. Wet cloth simulation: A fast simulation framework for liquid diffusion in porous textiles. Proceedings of Computer Graphics International Workshop on VFX, Computer Animation, and Stereo Movies; Ottawa, Ontario, Canada. 2011. p. 1–6.
5. Güdükbay U, Bayraktar S, Koca C, Özgüc B. Particle based simulation of the interaction between fluid and knitwear. SIViP. 2014;8(3):415–422.
6. Müller M, Heidelberger B, Hennix M, Ratcliff J. Position based dynamics. J Vis Commun Image Represent. 2007;18(2):109–118.
7. Bender J, Müller M, Otaduy MA, Teschner M, Macklin M. A survey on position-based simulation methods in computer graphics. Comput Graph Forum. 2014;33(6):228–251.
8. Akinci N, Ihmsen M, Akinci G, Solenthaler B, Teschner M. Versatile rigid-fluid coupling for incompressible SPH. ACM Trans Graph. 2012;31(4):62:1–62:8.
9. Müller M, Charypar D, Gross M. Particle based fluid simulation for interactive applications. Proceedings of the ACM SIGGRAPH/Eurographics Symposium on Computer Animation; Aire-la-Ville, Switzerland. 2003. p. 154–159.
10. Deul C, Charrier P, Bender J. Position-based rigid-body dynamics. Comput Anim Virtual Worlds. 2016;27(2):103–112.
11. Bender J, Koschier D, Charrier P, Weber D. Position-based simulation of continuous materials. Comput Graph. 2014;44:1–10.
12. Umetani N, Schmidt R, Stam J. Position-based elastic rods. Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Computer Animation; Copenhagen, Denmark. 2014. p. 21–30.
13. Müller M, Chentanez N, Kim TY, Macklin M. Strain based dynamics. Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Computer Animation; Copenhagen, Denmark. 2014. p. 149–157.
14. Macklin M, Müller M. Position based fluids. ACM Trans Graph. 2013;32(32):104:1–104:12.
15. Macklin M, Müller M, Chentanez N, Kim TY. Unified particle physics for real-time applications. ACM Trans Graph. 2014;33(4):153:1–153:12.
16. Fratarcangeli M. Position-based facial animation synthesis. Comput Anim Virtual Worlds. 2012;23(3-4):457–466.
17. Rumman NA, Fratarcangeli M. Position-based skinning for soft articulated characters. Comput Graph Forum. 2015;34(6):240–250.
18. Müller M. Hierarchical position based dynamics. Proceedings of the 5th Workshop on Virtual Reality Interactions and Physical Simulation. Grenoble, France. 2008. p. 1–8.
19. Fratarcangeli M, Pellacini F. Scalable partitioning for parallel position based dynamics. Comput Graph Forum. 2015;34(2):405–413.
20. Chu NS-H, Tai CL. MoXi: Real-time ink dispersion in absorbent paper. ACM Trans Graph. 2005;24(3):504–511.
21. Rungjiratananon W, Szego Z, Kanamori Y, Nishita T. Real-time animation of sand-water interaction. Comput Graph Forum. 2008;27(7):1887–1893.
22. Rungjiratananon W, Kanamori Y, Nishita T. Wetting effects in hair simulation. Comput Graph Forum. 2012;31(7):1993–2002.
23. Lin WC. Coupling hair with smoothed particle hydrodynamics fluids. Proceedings of the 11th Workshop on Virtual Reality Interaction and Physical Simulation; Bremen, Germany. 2014.
24. Lin WC. Boundary handling and porous flow for fluid-hair interactions. Comput Graph. 2015;52:33–42.
25. Bayraktar S, Güdükbay U, Özgüc B. Particle-based simulation and visualization of fluid flows through porous media. J Visualization. 2010;13:327–336.
26. Andrysco N, Beneš B, Brisbin M. Permeable and absorbent materials in fluid simulations. Proceedings of the Eurographics/ACM SIGGRAPH Symposium on Computer Animation; Dublin, Ireland. 2008.
27. Lenaerts T, Dutré P. Mixing fluids and granular materials. Comput Graph Forum. 2009;28(2):213–218.
28. Huber M, Eberhardt B, Weiskopf D. Boundary handling at cloth-fluid contact. Comput Graph Forum. 2015;34(1):14–25.
29. Shi X, Xiao S. Fluid absorption and diffusion in and between porous materials. Proceedings of the 14th ACM SIGGRAPH International Conference on Virtual Reality Continuum and Its Applications in Industry; Kobe, Japan. 2015. p. 23–26.
30. Adams B, Pauly M, Keiser R, Guibas LJ. Adaptively sampled particle fluids. ACM Trans Graph. 2007;26(3):No. 48.
31. Zhou K, Hou Q, Wang R, Guo B. Real-time KD-tree construction on graphics hardware. ACM Trans Graph. 2008;27(5):No. 126.
32. Shao X, Zhou Z, Magnenat-Thalmann N, Wu W. Stable and fast fluid-solid coupling for incompressible SPH. Comput Graph Forum. 2015;34(1):191–204.
33. Goswami P, Schlegel P, Solenthaler B. Interactive SPH simulation and rendering on the GPU. Proceedings of Eurographics/ACM SIGGRAPH Symposium on Computer Animation; Madrid, Spain. 2010. p. 55–64.
34. Lorensen WE, Cline HE. Marching cubes: A high resolution 3D surface construction algorithm. Proceedings of the 14th Annual Conference on Computer Graphics and Interactive Techniques; Anaheim, California, USA. 1987. p. 163–169.

Xuqiang Shao, born in 1982. He received the PhD degree from the State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, China. Now he is a lecturer of the North China Electric Power University. His main research interests include physically based computer animation and North China Electric Power University.

Wei Wu, born in 1961. Professor of State Key Laboratory of Virtual Reality Technology and Systems, Beihang University, China. Senior membership of China Computer Federation. His main research interests include wireless sensor network, distributed virtual reality and visualization technology.

Baoyi Wang, born in 1964. Professor of the North China Electric Power University. He is the membership of China Computer Federation. His main research interests include information security technology and visualization technology.

SUPPORTING INFORMATION

Additional Supporting Information may be found online in the supporting information tab for this article.

How to cite this article: Shao X, Wu W, Wang B. Position-based simulation of cloth wetting phenomena. *Comput Anim Virtual Worlds*. 2017;e1788. https://doi.org/10.1002/cav.1788