The Non-Newtonian Fluid Simulation Based on Predictive-Corrective Incompressible SPH

Conference Paper · September 2016

DOI: 10.1109/ICVRV.2016.20

4 authors, including:

Zhang Yalan
University of Science and Technology Beijing
27 PUBLICATIONS **72** CITATIONS
SEE PROFILE

Xiaokun Wang
University of Science and Technology Beijing
43 PUBLICATIONS **81** CITATIONS
SEE PROFILE

The Non-Newtonian Fluid Simulation Based on Predictive-Corrective Incompressible SPH

Yalan Zhang, Xiaojuan Ban, Xiaokun Wang, Xing Liu
School of Computer and Communication Engineering
University of Science and Technology Beijing
Beijing, China
yalan.zhang920503@gmail.com, banxj@ustb.edu.cn, wang1xiao2kun3@163.com, ustbdante@gmail.com

***Abstract*—*A novel non-Newtonian fluid simulation method for SPH is proposed in this article. The viscous liquid is modeled by a non-Newtonian fluid flow, and the variable viscosity under shear stress is achieved using a viscosity model known as Cross model. To avoid tensile instability and improve numerical stability, a predictive-corrective method, aimed at correcting density error, of setting up individual stiffness parameters for each particle to be added. Furthermore, to improve the overall efficiency of the proposed method, a global adaptive time-stepping method that adjust the time step automatically in accordance with individual scenarios is utilized.*

Keywords—Physically-based animation; fluid simulation; non-Newtonian fluid; Smoothed Particle Hydrodynamics; predictive-corrective method

I. INTRODUCTION

Fluids play an important role in everyday life. Examples for fluid phenomena are wind, weather and ocean waves. As simple and ordinary these phenomena may seem to be, their simulation is rather complex and difficult. Lagrangian fluid simulation is a hotspot in computer animation. Its particle-based nature makes simulations of small scale phenomena possible, while mass conservation is trivially satisfied. Currently, physics-based fluid animation re-searches have been focused on homogeneous incompressible Newtonian fluids with linear physical characteristics, and non-Newtonian fluids has rarely been involved. However, the non-Newtonian fluid is widely present in the chemical industry, oil industry, mining engineering, bio medication, food processing and many other fields, such as pit filled paste, blood, melted chocolate, mud (Fig.1), etc. Therefore, the simulation for non-Newtonian fluid not only acts as a supplement for conventional fluid animation, it is also an important and challenging research topic in the field of graphics and virtual reality.

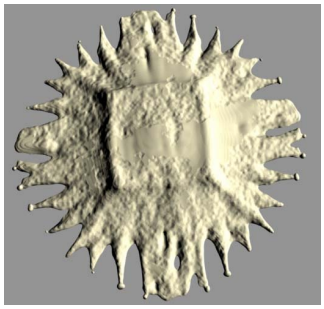

Fig. 1. *Non-Newtonian fluid (mud).*

Since the physical nature of non-Newtonian fluid is very complicated, it is pretty difficult to exactly simulate non-Newtonian fluid. A Non-Newtonian fluid has characteristic properties of both solids and fluids [1]. Taking For example, when the shear thinning non-Newtonian fluid flows with low velocity, its viscosity becomes higher, and the fluid starts to exhibit properties of solids; while as it flows with high velocity, its viscosity gets smaller, and non-Newtonian fluid exhibit liquid properties. In recent years, Smoothed particle hydrodynamics [2] (SPH) has become an important particle-based method in computer animation. However, it is rather difficult to utilize this method to simulate non-Newtonian fluids. First, the performance of fluid is directly affected by the control model, but the traditional methods of controlling are mostly designed for ideal Newtonian fluid rather than non-Newtonian fluid. Due to the non-linear relationship between viscosity and shear force, the adequate physical equations and stable approximate simulation method should be re-established for non-Newtonian fluids. Secondly, SPH method has some inherent problems common for massless methods [3]. Particles condense into a group of trends in the flow of non-Newtonian fluid because of tensile instability. In addition, due to the complicated physical equations of non-Newtonian fluids, small time step is necessary, which proves to be rather time-consuming.

A novel non-Newtonian fluid simulation method based on predictive-corrective incompressible SPH is presented in this paper. In order to capture the viscous behavior, the viscosity of Newtonian fluid and non-Newtonian fluid flows is measured by the Cross model. The tensile instability is removed by the predictive-corrective method that corrects density error by pressure and predictive intermediate velocity. Furthermore, an adaptive time-stepping approach that automatically estimates appropriate time steps in accordance with individual scenarios is proposed, to improve the overall efficiency of the predictive-corrective method.

II. RELATED WORKS

There are two major approaches to simulate Non-Newtonian fluid: Euler's method and Lagrange method. In Euler's method, the Simulation region is dispersed to a grid point. The Physical Property (velocity, density) of the grid point can be obtained by solving the governing equations. In

978-1-5090-5188-5/16 $31.00 © 2016 IEEE
DOI 10.1109/ICVRV.2016.20

2004, Goatskin [4] proposed a quasi linear plastic model to control the change of viscosity in the gradual transformation from a solid to a Non-Newtonian fluid. This model meets the von Mises's yield condition. They simulated the melting of a piece of wax with this model, using explicit Euler method. Losasso [5] modeled unified fluids with different viscoelasticities and densities by extending the particle level set. There are many different level sets in the same grid region. The exchanges between different level sets can be achieved by adding the pressure correction process before calculating viscosity. Batty [6] found out that the Laplace operator and Strain rate items of Non-Newtonian fluid should be computed under the velocity field with no divergence; therefore, the pressure correction process was added after the calculation of viscosity. Besides, they proposed an implied unconditionally stable simulation method based on marker-and-cell. Even though they proposed high precision free surface boundary conditions, it still cannot simulate High viscosity Newtonian fluid. Bergou [7] proposed a method based on discrete differential geometry to simulate one dimensional linear viscous fluid. This approach can simulate one dimensional phenomenon vividly, but distortion often appears when the flow layer gets thicker. Batty [8] simulated thin viscous layer with dimensionality deduce technology. They combined nonlinear surface tension and smallest-discrete-surface-area-based formulas and reserved the mass of Unit triangular mesh with the Local grid reconstruction technology. This approach can simulate the physical conditions of viscous thin layer, such as bending and draping.

Mesh free methods are superior when free surface and complex interface are to be dealt with. So that mesh free methods represented by SPH method have been more commonly applied in the field of non-Newtonian fluid simulation. Müller [9] simulated fluids of High elasticity and high plasticity by structuring a point-based animation modeling method. He combined continuum mechanics with von Mises's yield condition and calculated the displacement and the velocity using moving least squares. [10] also completed simulations with a General Newtonian Fluid model, and the viscosity of fluid was controlled by the Jump Number. The melting process of a heated non-Newtonian fluid [11], which later becomes a low viscosity fluid, was also modeled by the General Newtonian Fluid model. [12] proposed a free surface SPH method. This method can be applied to Newton fluid and viscoplastic fluid, and simulates the bending phenomena of viscous fluids. The relationship between the pressure and the viscosity of non-Newtonian fluids was described with the real polymer state equation [13]. [14] can simulate the flow of polymer with high viscosity and high pressure. In addition, the unified molding of viscous Newtonian fluid and shear thinning non-Newtonian fluid was simulated, but this method can only simulate fluid with low viscosity; whereas unstable phenomenon will appear when the time step is larger.

In addition to Euler and Lagrange methods, recently there is a novel alternative, the codimensional method [15], that is able to simulate a non-Newtonian fluid. This method enables the simulation of interaction between different dimensions of a non-Newtonian fluid, such as a filamentous remaining marks left on a pool of paint after getting brushed over, which is difficult for Euler and Lagrange methods to achieve.

In recent years, the standard SPH has been gradually replaced by the predictive-corrective method. Predictive - corrective algorithm was first proposed by [16] applied to the SPH method, the core idea is to predict the state of a fluid in absence of pressure, then use pressure for its conduct correction. Ihmsen et al [17] proposed an implicit SPH method that adjusts fluid's density by pressure according to the relationship between density and velocity in the next time step. [18] proposed a divergence-free velocity method, which is essentially also a predictive-corrective method that prevents volume compression and enforces a divergence-free velocity field formulation of the physical laws.

III. The non- Newtonian fluid model

A. Lagrangian Formulation

Non-Newtonian fluids are continuous medium, whose motion equations must follow the principle of conservation of mass and energy conservation principles. In the Lagrangian formulation, because the number of particles is constant and each particle has a constant mass, mass conservation is guaranteed unconditionally. Compared to a Newtonian fluid [19], a non-Newtonian fluid has elasticity and plasticity, so a strain tensor term should be added in the governing equations. The momentum equation can be written as the following:

$$\frac{d\mathbf{v}}{dt} = -\frac{1}{\rho}\nabla p + \frac{1}{\rho}\nabla \cdot \boldsymbol{\tau} + \mathbf{g} \tag{1}$$

where t denotes the time, $\mathbf{v}$ the velocity field, ρ the density, p the pressure, $\mathbf{g}$ the gravity acceleration vector and $\boldsymbol{\tau}$ the shear stress tensor.

Lagrangian formulation of equation (1) represents the acceleration of a particle moving with the fluid flow. The term $-\frac{1}{\rho}\nabla p$ is related to particle acceleration due to pressure changes in the fluid. The term $\frac{1}{\rho}\nabla \cdot \boldsymbol{\tau}$ describes the viscous acceleration due to friction forces caused by particles with different velocities, which plays a key role in non-Newtonian fluid animation.

B. Cross Model

Currently, there are many models for non-Newtonian fluid simulation, among which the most common one is the General Newtonian Fluid model [20]. This model is suitable for Newtonian fluid and non-Newtonian fluid conversion, especially for the transition of a non-Newtonian fluid with high viscosity to a liquid of low viscosity by melting [10] [11]. However, it cannot fully capture some realistic non-Newtonian fluid physical properties, such as buckling, folding and coiling.

For our simulation, the Cross model [14] was used to build a unified model for a Newtonian fluid and a shear thinning non-Newtonian fluid. The Cross model can not only capture the physical properties of a non-Newtonian fluid, but also simulate a low viscosity Newtonian fluid. In particular, for non-Newtonian fluids, the shear stress $\boldsymbol{\tau}$ is a nonlinear function of the rate-of deformation tensor $\mathbf{D} = \nabla\mathbf{v} + (\nabla\mathbf{v})^T$:

$$\boldsymbol{\tau} = \rho\upsilon(D)\mathbf{D}, \text{ with } D = \sqrt{\frac{1}{2}\cdot trace(\mathbf{D})^2} \tag{2}$$

The Cross model is one of the simplest and most frequently used model for the shear-thinning behavior, i.e., the fluid's viscosity decreases when the local shear rate D increases, thus the kinematic viscosity υ is defined as a function of D :

$$\upsilon(D) = \upsilon_\infty + \frac{\upsilon_0 - \upsilon_\infty}{1 + (KD)^n} \tag{3}$$

where K and n are positive parameters to control the viscosity of the fluid. They were first artificially set based on the properties of the fluid that is to be simulated. Assuming $K = 0$ in Equation (3), the non-Newtonian fluid model is simplified to a Newtonian fluid with constant kinematic viscosity υ_0. Respectively, υ_0 and υ_∞ are the limiting values of the viscosity at low and high shear rates. The units of viscosity are m^2 / s .

C. Smoothed Particle Hydrodynamics

In the Lagrangian setting, the basic idea is to approximate a generic field variable $A(x_i)$ by using a finite set of sampling points x_j with mass m_j and density ρ_j :

$$\langle A(\boldsymbol{x})\rangle = \sum_j \frac{m_j}{\rho_j} A_j W_{ij} \tag{4}$$

where $W_{ij} = W(\boldsymbol{x} - \boldsymbol{x}_j, h)$ is the kernel function with h the particle radius, and j iterating all the neighbors.

In the SPH approach, the derivatives of field quantities only affect the smoothing kernel. The gradient of A is simply:

$$\nabla A(\boldsymbol{x}) \approx \langle \nabla A(\boldsymbol{x})\rangle = \sum_j \frac{m_j}{\rho_j} A_j \nabla W_{ij} \tag{5}$$

By substituting equation (5) into equation (1), the density at location $\boldsymbol{x}_i$ is:

$$\rho_i = \sum_j m_j W_{ij} \tag{6}$$

There are many formulas to solve the pressure. In the SPH framework, incompressible fluids approximated by weakly compressible fluids. In other words, the pressure is usually given by the density of state equation solver. For simplicity, equation of state proposed by [16] is often used to compute the pressure:

$$p_i = k(\rho_i - \rho_0) \tag{7}$$

where k is a gas constant that depends on the temperature, and ρ_0 is the rest density. The value $\rho_0 = 1000\ kg / m^3$ turns out to be suitable for all experiments.

To compute the pressure acceleration, substitute $-\frac{1}{\rho}\nabla p$ into Equation (5) :

$$-\frac{1}{\rho_i}\nabla p_i = -\sum_j m_j \frac{p_i + p_j}{\rho_j}\nabla W_{ij} \tag{8}$$

In order to compute the shear stress $\boldsymbol{\tau}_i$ at particle i , the same SPH approximation as [21] is adopted for the deformation tensor $\mathbf{D}_i = \nabla\mathbf{v}_i + (\nabla\mathbf{v}_i)^T$ with:

$$\nabla\mathbf{v}_i = \sum_j \frac{m_j}{\rho_j}(\mathbf{v}_i - \mathbf{v}_j)\otimes\nabla W_{ij} \tag{9}$$

After updating shear stress at all particles, the viscous acceleration is approximated by:

$$\frac{1}{\rho}\nabla\cdot\boldsymbol{\tau}_i = \sum_j m_j\left(\frac{\tau_i}{\rho_i^2} + \frac{\tau_j}{\rho_j^2}\right)\nabla W_{ij} \tag{10}$$

IV. Predictive-Corrective Algorithm

In the non-Newtonian fluid model proposed above, due to the inherent numerical problems in typical mesh less methods, when a non-Newtonian fluid flows, the particles tend to condense into a group of trends that result in unreasonable breakage. In order to address this problem, a prediction - correction algorithm is proposed to correct density by pressure.

Currently, there are many different pressure solvers that minimize the density error $\rho - \rho_0$ [16] [17]. The algorithm proposed in this paper is similar to [16]; while a new solver that sets a separate stiffness parameter k_i for each fluid particle i to satisfy local incompressibility is introduced in this paper, making the density of the entire fluid system remains unchanged.

The pressure gradient is computed using the SPH formulation [17]:

$$\nabla p_i = k_i \nabla \rho_i = k_i \nabla \sum_j m_j \nabla W_{ij} \tag{10}$$

The pressure force of particle i caused by pressure acceleration is determined by:

$$\mathbf{F}_i^p = -m_i \frac{1}{\rho} \nabla p = -k_i \frac{m_i}{\rho_i} \sum_j m_j \nabla W_{ij} \tag{11}$$

$\mathbf{F}_{j \leftarrow i}^p$ is the pressure forces that act from particle i on the neighboring particles j. According to Newton's third law, its value is equal to the total pressure of all neighbor particles j that affect i, which satisfies $\mathbf{F}_i^p + \sum_j \mathbf{F}_{j \leftarrow i}^p = 0$. Then we can get:

$$\mathbf{F}_{j \leftarrow i}^p = k_i \frac{m_i}{\rho_i} m_j \nabla W_{ij} \tag{12}$$

By using a semi-implicit Euler scheme for position and velocity update, the velocity can be rewritten as: $\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i(t) + \Delta t \frac{\mathbf{F}_i^{adv}(t) + \mathbf{F}_i^p(t)}{m_i}$ with unknown pressure forces $\mathbf{F}_i^p$ and known non-pressure forces $\mathbf{F}_i^{adv}(t)$ such as gravity, surface tension and viscosity. Following the projection concept, intermediate (predicted) velocities are set to be $\mathbf{v}_i^*$:

$$\mathbf{v}_i^* = \mathbf{v}_i(t) + \Delta t \frac{\mathbf{F}_i^{adv}(t)}{m_i} = \mathbf{v}_i(t) + \Delta t \cdot \mathbf{g} + \Delta t \cdot \frac{1}{\rho} \nabla \cdot \boldsymbol{\tau}_i \tag{13}$$

The velocity changes $\Delta \mathbf{v}_i$ caused by pressure forces is:

$$\Delta \mathbf{v}_i = \mathbf{v}_i(t + \Delta t) - \mathbf{v}_i^* = \Delta t \frac{\mathbf{F}_i^P(t)}{m_i} = -\Delta t \frac{k_i}{\rho_i} \sum_j m_j \nabla W_{ij} \tag{14}$$

Similarly:

$$\Delta \mathbf{v}_j = \Delta t \frac{\mathbf{F}_{j \leftarrow i}^p(t)}{m_i} = \Delta t \frac{k_i}{\rho_i} m_j \nabla W_{ij} \tag{15}$$

According to [17], the intermediate density caused by intermediate velocity is:

$$\rho_i^* = \rho_i(t) + \Delta t \sum_j m_j (\mathbf{v}^*_i - \mathbf{v}^*_j) \nabla W_{ij} \tag{16}$$

We now search for pressure forces to resolve the compression, i.e., the deviation from the rest density $\Delta \rho_i$:

$$\begin{aligned}
\Delta \rho_i &= \Delta t \sum_j m_j (\Delta \mathbf{v}_i - \Delta \mathbf{v}_j) \nabla W_{ij} \\
&= \Delta t \sum_j m_j (-\Delta t \frac{k_i}{\rho_i} \sum_j m_j \nabla W_{ij} - \Delta t \frac{k_i}{\rho_i} m_j \nabla W_{ij\,j}) \nabla W_{ij} \\
&= -\frac{k_i}{\rho_i} \Delta t^2 ((\sum_j m_j \nabla W_{ij})^2 + \sum_j (m_j \nabla W_{ij})^2) \\
&= \rho_0 - \rho_i^*
\end{aligned}$$

Solving for k_i yields:

$$k_i = \frac{\rho_0 - \rho_i^*}{\Delta t^2} \alpha_i \text{, with } \alpha_i = \frac{\rho_i}{((\sum_j m_j \nabla W_{ij})^2 + \sum_j (m_j \nabla W_{ij})^2)} \tag{17}$$

Algorithm 1.simulation

for all particles i **do**
  find neighborhoods N_i
  compute time step size Δt
end for
for all particles i **do**
  update ρ_i using Equation (6)
  update α_i using Equation (18)
end for
for all particles i **do**
  update $\nabla \mathbf{v}_i$ using Equation (9)
  update $\boldsymbol{\tau}_i$ using Equation (10)
end for
for all particles i **do**
  update $\mathbf{v}_i^*$ using Equation (14)
end for
while ($\rho_{avg} - \rho_0 > \eta$)**or** $iter < 2$ **do**
  for all particles i **do**
    update ρ_i^* using Equation (17)
    update k_i using Equation (18)
    $\rho_i^* = \rho_i^* + \frac{k_i}{\alpha_i} \Delta t^2$
  end for
end while
for all particles i **do**
  $\mathbf{v}_i(t + \Delta t) = \mathbf{v}_i^* - \Delta t \sum_j m_j (\frac{k_i}{\rho_i} + \frac{k_j}{\rho_j}) \nabla W_{ij}$
  $\mathbf{x}_i(t + \Delta t) = \mathbf{x}_i(t) + \Delta t \cdot \mathbf{v}_i(t + \Delta t)$
end for

Algorithm 1 outlines the simulation using our novel method in detail. First, the particle neighborhoods N_i using compact hashing [22] is determined. Then for each particle, the density ρ_i and the factor α_i are computed. Since this factor solely depends on the current positions, it is precomputed before executing the solvers, which saves the solver from significant computational works. The third step in the simulation loop is to compute the strain tensor term of a non-Newtonian fluid, which is the main difference between a Newtonian fluid and a non-Newtonian fluid.

The forth and the fifth step of the algorithm are predictive-correction steps, whose propose is to minimize the density error by the pressure solver, scilicet solving $\rho - \rho_0 = f(p)$. In this step, the non-pressure forces are used to compute predicted velocities $\mathbf{v}_i^*$. The constant density solver uses this prediction and the precomputed factors α_i to determine the pressure forces for each neighborhood in order to correct the density error.

Finally, the resulting velocity changes are used to update the particle velocities.

V. Implementation and Results

A. The Non-Newtonian Fluid Model

All timings are given for an Intel 3.50 GHz CPU with 4 cores. The simulation software is parallelized with Open MP. The simulation and surface reconstruction are actualized with C++ language and multi-threading technology. We successfully tested our method in combination with the surface tension models of anisotropic kernels [23]. The density fluctuation was set to 0.01.

Figure 2 shows the Newtonian fluid and non-Newtonian fluid in particle state according to our method. As described in Section 3.2, when $K = 0$ the non-Newtonian fluid model is simplified to a Newtonian fluid with constant kinematic viscosity (Figure 2, left). When the fluid touches the container at the bottom of the scene (the container is set to be transparent for observers' convenience), the fluid particles that touch the container splash upward. In contrast, the non-Newtonian fluid particles (shown in Figure 2, right) cannot splash when they collide with container due to their viscosity.

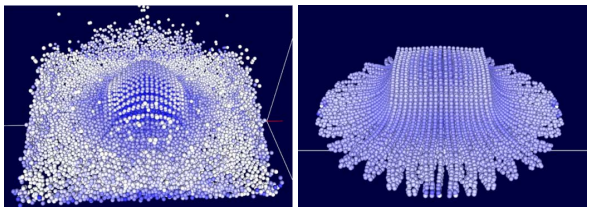

Fig. 1. Comparison between the Newtonian fluid (left) and the non-Newtonian fluid (right)

B. Predictive-corrective algorithm

Figure 3 shows a non-Newtonian fluid before and after the predictive-corrective step for comparison. The figure shows the free fall of a water column. There are obvious fault in the non-Newtonian fluid water column without the predictive-corrective step, and the density error increases as the fluid flow. After predictive-corrective step, an obvious improvement can be seen. After correction, the bottom layer distance of fluid particles is slightly longer than the top layer distance, which is a normal phenomenon resulted from the shear thinning of a non-Newtonian fluid. As stated above, the viscosity becomes smaller when velocity gets higher, which results in a longer distance between fluid layers.

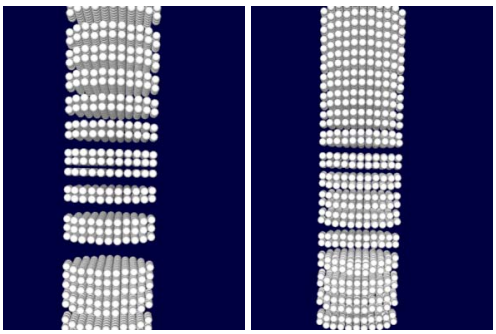

Fig. 2. The non-Newtonian fluid before (left) and after (right) predictive-corrective step

During the experiment, when the scene was small or the fluid particles moved gently, running the algorithm with predictive-corrective step was found to be time and storage space-consuming. Take the experiment shown in figure 4 as an example. Figure 4 shows a free falling non-Newtonian fluid column with 13671 particles, $\upsilon_0 = 2$, $\upsilon_\infty = 0.2$, $K = 1$, $n = 0.5$. When the time step was set to 0.2 ms, it took 53.3 ms for every time step without predictive-corrective step; while with predictive-corrective step, it took 83.4 ms. However, when the scene is large or the fluid particles collide intensely, the non-Newtonian fluid will be unstable with the larger time step. As shown in figure 4 left, when the time step was set to 0.3 ms, fluid particles scattered in all directions without the predictive-corrective step, while the algorithm still run with the predictive-corrective step (figure 4, right).

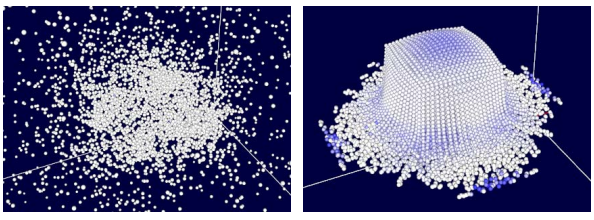

Fig. 3. Test for stability. Fluid particles scattered in all directions without predictive-corrective step (left), while the algorithm still run with predictive-corrective step (right).

C. Adaptive Time Step

To ensure the efficiency of the algorithm after adding predictive-corrective step, a global adaptive time-stepping method [23] is adopted. Experiments showed that the viscosity of the non-Newtonian fluid makes its particles more prone to be affected by one another; therefore, d adaptively integrating asynchronous time steps is not suitable. Different from [23], a global adaptive time-stepping method that sets all particles to the same time step and adjusts the time step automatically based on different scenarios.

The time step of SPH fluids must be constrained for numerical stability and convergence. The Courant-Friedrich-Levy (CFL) condition:

$$\Delta t_{CFL} \le \lambda_v \left(\frac{h}{v_{max}} \right) \tag{18}$$

ensures that the numerical propagation speed is higher than the physical propagation speed, where $v_{max} = \max_i \|\boldsymbol{v}_i\|$ is the maximum of the particle velocities, with coefficient $\lambda_v < 1$.

Besides, the time step of non-Newtonian fluid is limited by its viscosity [14]:

$$\Delta t_f \le \lambda_f \left(\frac{h}{f_{max}} \right) \tag{19}$$

The final time step needs to satisfy the following condition:

$$\Delta t \le \min\left(\Delta t_{CFL}, \Delta t_f \right) \tag{21}$$

Table 1 shows the efficiency of the algorithm before and after adding dynamic time steps .The experiment scenario have 13671 particles, with $\upsilon_0 = 2$, $\upsilon_\infty = 0.2$, $K = 1$, $n = 0.5$.

Table 1. Efficiency comparison

Time step	Animating time	Totally computing time
constant steps	2 s	556 s
globally adaptive steps	2 s	254 s

VI. Conclusion

A novel SPH-based technique for simulating a non-Newtonian fluid is presented in this paper. The technique relies on the SPH approximation of a non-Newtonian fluid, where the variable viscosity is ruled by the Cross Model. A predictive-corrective method is added to avoid tensile instability and numerical instability. Furthermore, an adaptive time-stepping method that automatically adjusts required time case by case is included to increase the algorithmic and simulation efficiency.

ACKNOWLEDGMENT

This work was supported by National Natural Science Foundation of China (No. 61272357, 61300074, 61572075).

References

[1] Ellero M, Tanner R I. SPH simulations of transient viscoelastic flows at low Reynolds number[J]. Journal of Non-Newtonian Fluid Mechanics, 2005, 132(1):61-72.

[2] Müller M, Charypar D, Gross M. Particle-based fluid simulation for interactive applications[C]//Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation. Eurographics Association, 2003: 154-159.

[3] Monaghan J J. Simulating free surface flows with SPH. [J]. Journal of Computational Physics, 1994, 110(2):399-406.

[4] Goktekin T G, Bargteil A W, O'Brien J F. A Method for Animating Viscoelastic Fluids[J]. Acm Transactions on Graphics, 2004, 23(3):461-466.

[5] Losasso F, Shinar T, Selle A, et al. Multiple interacting liquids.[J]. Acm Transactions on Graphics, 2006, 25(3):812-819.

[6] Batty C, Bridson R. Accurate Viscous Free Surfaces for Buckling, Coiling, and Rotating Liquids.[C]// Proceedings of the 2008 Eurographics/ACM SIGGRAPH Symposium on Computer Animation, SCA 2008, Dublin, Ireland, 2008. 2008:219-228.

[7] Bergou M, Audoly B, Vouga E, et al. Discrete viscous threads[J]. Acm Transactions on Graphics, 2010, 29(4):157-166.

[8] Batty C, Uribe A, Audoly B, et al. Discrete Viscous Sheets[J]. Acm Transactions on Graphics, 2012, 31(4):13-15.

[9] Müller M, Keiser R, Nealen A, et al. Point based animation of elastic, plastic and melting objects[C]// ACM Siggraph/eurographics Symposium on Computer Animation. Eurographics Association, 2004:141-151.

[10] Afonso P, Fabiano P, Thomas L, Geovan T. Particle-based viscoplastic fluid/solid simulation[J]. Computer-Aided Design, 2009, 41(4): 306-314.

[11] Paiva A, Petronetto F, Lewiner T, et al. Particle-based non-Newtonian fluid animation for melting objects[C]//2006 19th Brazilian Symposium on Computer Graphics and Image Processing. IEEE, 2006: 78-85.

[12] Rafiee A, Manzari M T, Hosseini M. An incompressible SPH method for simulation of unsteady viscoelastic free-surface flows[J]. International Journal of Non-Linear Mechanics, 2007, 42(10): 1210-1223.

[13] Fan X J, Tanner R I, Zheng R. Smoothed particle hydrodynamics simulation of non-Newtonian moulding flow[J]. Journal of Non-Newtonian Fluid Mechanics, 2010, 165(5): 219-226.

[14] Andrade L F D S, Sandim M, Petronetto F, et al. SPH Fluids for Viscous Jet Buckling[C]// Graphics, Patterns and Images. IEEE, 2014:65-72.

[15] Zhu B, Lee M, Quigley E, et al. Codimensional non-Newtonian fluids[J]. Acm Transactions on Graphics, 2015, 34(4):1-9.

[16] Solenthaler B, Pajarola R. Predictive-corrective incompressible SPH[J]. Acm Transactions on Graphics, 2009, 28(3):341-352.

[17] Ihmsen M, Cornelis J, Solenthaler B, et al. Implicit Incompressible SPH[J]. IEEE Transactions on Visualization & Computer Graphics, 2014, 20(3):426-435.

[18] Bender J, Koschier D. Divergence-free smoothed particle hydrodynamics[C]//Proceedings of the 14th ACM SIGGRAPH/Eurographics Symposium on Computer Animation. ACM, 2015: 147-155.

[19] Becker M, Teschner M. Weakly compressible SPH for free surface flows.[C]// ACM Siggraph/eurographics Symposium on Computer Animation, SCA 2007, San Diego, California, Usa, August. 2007:209-217.

[20] de Souza Mendes P R, Dutra E S S, Siffert J R R, et al. Gas displacement of viscoplastic liquids in capillary tubes[J]. Journal of Non-Newtonian Fluid Mechanics, 2007, 145(1): 30-40.

[21] Morris J P, Fox P J, Zhu Y. Modeling low Reynolds number incompressible flows using SPH[J]. Journal of computational physics, 1997, 136(1): 214-226.

[22] Ihmsen M, Akinci N, Becker M, et al. A Parallel SPH Implementation on Multi‐Core CPUs[C]// Computer Graphics Forum. Blackwell Publishing Ltd, 2011:99‐112.

[23] Ban X, Wang X, He L, et al. Adaptively stepped SPH for fluid animation based on asynchronous time integration[J]. Neural Computing and Applications, 2016: 1-10.