Applied Mathematical Modelling 38 (2014) 3822–3833

Contents lists available at ScienceDirect

Applied Mathematical Modelling

journal homepage: www.elsevier.com/locate/apm

A new kernel function for SPH with applications to free surface flows ☆

X.F. Yang [a], S.L. Peng [b], M.B. Liu [a,*]

[a] Key Laboratory for Mechanics in Fluid Solid Coupling Systems (LMFS), Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China
[b] State Key Laboratory of High-Temperature Gas Dynamics (LHD), Institute of Mechanics, Chinese Academy of Sciences, Beijing 100190, China

ARTICLE INFO

Article history:
Received 7 March 2013
Received in revised form 17 December 2013
Accepted 19 December 2013
Available online 30 December 2013

Keywords:
Kernel function
Smoothed particle hydrodynamics (SPH)
Free surface flow
Meshfree particle method

ABSTRACT

Smoothed particle hydrodynamics (SPH) is a popular meshfree Lagrangian particle method, which uses a kernel function for numerical approximations. The kernel function is closely related to the computational accuracy and stability of the SPH method. In this paper, a new kernel function is proposed, which consists of two cosine functions and is referred to as double cosine kernel function. The newly proposed double cosine kernel function is sufficiently smooth, and is associated with an adjustable support domain. It also has smaller second order momentum, and therefore it can have better accuracy in terms of kernel approximation. SPH method with this double cosine kernel function is applied to simulate a dam-break flow and water entry of a horizontal circular cylinder. The obtained SPH results agree very well with the experimental results. The double cosine kernel function is also comparatively studied with two frequently used SPH kernel functions, Gaussian and cubic spline kernel functions.

1. Introduction

Smoothed Particle Hydrodynamics (SPH) [1,2] is a meshfree Lagrangian particle method, in which particles are used to represent the state of a system, and are used to approximate governing equations through using a smoothing or kernel function (or abbreviated as a kernel). As a comparatively new computational method, SPH combines the advantages of meshfree, Lagrangian and particle methods. First, particles are used to represent the state of a system and these particles can freely move according to internal particle interactions and external forces. Therefore it can naturally obtain history of fluid motion, and can easily track material interfaces, free surfaces and moving boundaries. The meshfree nature of SPH method remove the difficulties due to large deformations since SPH uses particle rather than mesh as a computational frame to approximate related governing equations. These nice features of SPH make it fairly attractive in modeling problems with deformable boundaries, moving interfaces, free surfaces and large deformations [3–6].

SPH was originally invented to solve astrophysical problems in open space [1,2] and it was later extended to simulate many other compressible flows such as shock problems [3–7]. By treating the flow as slightly compressible with an appropriate equation of state, the SPH method can be used to simulate incompressible flows successfully [8,9]. It is also possible to rigorously treat incompressible flows in SPH through solving the pressure Poisson equation [10,11] in a way similar to the moving particle semi-implicit (MPS) method [12,13]. SPH was also expended to simulate multi-phase flows with sharp interfaces [14–16]. More details of the SPH method and its recent developments can be found in a number of review papers [4–6,17].

☆ This article belongs to the Special Issue: Topical Issues on computational methods, numerical modelling & simulation in Applied Mathematical Modelling.

* Corresponding author. Tel.: +86 10 82544024.
E-mail address: liumoubin@imech.ac.cn (M.B. Liu).

0307-904X/$ - see front matter © 2014 Elsevier Inc. All rights reserved.
http://dx.doi.org/10.1016/j.apm.2013.12.001

Kernel functions are very important in SPH method as they are closely related to the computational accuracy and stability [18,19] of the SPH method, and therefore they have been studied by many researchers. According to [18] the stability of SPH is related to the second derivative of the kernel function. Monaghan [17] proposed a method on how to construct high order kernel functions by starting with normal standard kernel functions and gave a Super Gaussian kernel as an example. A disadvantage of high order kernel functions is that they can become negative in part of the domain and this may cause unphysical results. Fulk and Quinn [20] have studied SPH kernel functions in one dimensional spaces. According to their assessment, bell-shaped kernels perform better than kernels with other shapes and no kernel function is significantly better than the cubic spline. Liu et al. [21] presented a general approach to construct analytical kernel functions. They also constructed a new quadric kernel function which was applied to the one dimensional shock problem and a one dimensional TNT detonation problem. Ha [22] provided a comparative study on Liu's quartic kernel function with the cubic spline kernel function and quintic spline kernel function. He pointed out that the quartic kernel gave the best results for a benchmark lid-driven cavity problem. Jin and Ding [23] provided three criterions for determining suitable kernel functions through evaluating computational accuracy. They concluded that Gaussian and Q-spline kernel functions should be regarded as the best kernel functions among ten kernels in their study. Cabezón et al. [24] proposed a one-parameter family of interpolating kernels with compact support based on the harmonic-like functions.

2. SPH methodology

2.1. SPH approximations

In the SPH method, for a field function $f(\boldsymbol{r})$, its value, f, at point $\boldsymbol{r}_a$ can be first approximated using the following integral formula:

$$\langle f(\boldsymbol{r}_a)\rangle = \int f(\boldsymbol{r})W(\boldsymbol{r}_a - \boldsymbol{r}, h)dV, \tag{1}$$

where $W(\boldsymbol{r}_a - \boldsymbol{r}, h)$ is a kernel function, dV is a differential volume element. The parameter h in kernel W is referred to as a smoothing length which determines the actual size of the integral domain.

Eq. (1) can be further written as the averaged summation over a set of particles, each associated with a small mass elements as follows:

$$\langle f_a\rangle = \sum_b f_b W_{ab} V_b, \tag{2}$$

where $f_a = f(\boldsymbol{r}_a)$, $W_{ab} = W(\boldsymbol{r}_a - \boldsymbol{r}_b, h)$. The indexes a and b denote the labels of particles. Particle b has position $\boldsymbol{r}_b$, velocity $\boldsymbol{u}_b$, mass m_b, density ρ_b, and volume $V_b = m_b/\rho_b$.

The gradient of function $f(\boldsymbol{r})$ can be obtained by differentiating the kernel in Eq. (2) as follows:

$$\langle \nabla f_a\rangle = \sum_b f_b \nabla_a W_{ab} V_b, \tag{3}$$

where $\nabla_a W_{ab}$ denotes the gradient of the kernel W taken with respect to a-coordinates.

2.2. SPH equations of motion

In this paper, the Lagrangian form of the Reynolds-averaged Navier–Stokes (RANS) equations are used as governing equations for incompressible, viscous flows:

$$\frac{d\rho}{dt} = -\rho\nabla\cdot\boldsymbol{U}, \tag{4}$$

$$\frac{d\boldsymbol{U}}{dt} = -\frac{1}{\rho}\nabla P + \frac{\mu}{\rho}\nabla^2\boldsymbol{U} + \frac{1}{\rho}\nabla\cdot(\rho\boldsymbol{R}) + \boldsymbol{g}, \tag{5}$$

where ρ is fluid density, $\boldsymbol{U}$ is the velocity vector, P is pressure, μ is the dynamic viscosity, $\boldsymbol{g}$ is the gravitational acceleration, and $\boldsymbol{R}$ is the Reynolds stress tensor whose elements are:

$$R_{ij} = -\overline{u_i' u_j'}. \tag{6}$$

The eddy viscosity assumption is used to model the Reynolds stress tensor as:

$$\boldsymbol{R} = 2\nu_t\boldsymbol{S} - \frac{2}{3}k\boldsymbol{I}, \tag{7}$$

where ν_t is the eddy viscosity, $\boldsymbol{S}$ is the mean rate-of-strain tensor, k is turbulence kinetic energy and $\boldsymbol{I}$ is unite tensor.

The Smagorinsky model of eddy viscosity is widely used:

$$v_t = (C_s \Delta l)^2 \sqrt{2 S_{ij} S_{ij}}, \tag{8}$$

where C_s is Smagorinsky constant and Δl is a mixing length which is assumed to be the initial particle spacing in SPH. Following Lo and Shao [10], a value of 0.12 is chosen for C_s. S_{ij} are the elements of $\boldsymbol{S}$ given by:

$$S_{ij} = \frac{1}{2}\left(\frac{\partial U_j}{\partial x_i} + \frac{\partial U_i}{\partial x_j}\right). \tag{9}$$

An additional equation is required to calculate turbulence kinetic energy k. For the definition of k is:

$$k = \frac{1}{2}\overline{u_i' u_i'}, \tag{10}$$

and noting Eq. (6), the relation between k and the Reynolds stress tensor can be written as:

$$k = -\frac{1}{2} R_{ii}. \tag{11}$$

Using Eq. (2) and a few mathematical techniques, the continuity Eq. (4) is expressed as:

$$\frac{d\rho_a}{dt} = \sum_b m_b \boldsymbol{U}_{ab} \cdot \nabla_a W_{ab}. \tag{12}$$

The right hand side of the momentum Eq. (5) can be expressed as:

$$\left(\frac{1}{\rho}\nabla P\right)_a = \sum_b m_b \left(\frac{p_a}{\rho_a^2} + \frac{p_b}{\rho_b^2}\right) \nabla_a W_{ab}, \tag{13}$$

$$\left(\frac{\mu}{\rho}\nabla^2 \boldsymbol{U}\right)_a = \sum_b \frac{8\mu m_b \boldsymbol{r}_{ab} \cdot \nabla_a W_{ab}}{(\rho_a + \rho_b)^2 (r_{ab}^2 + 0.01h^2)} \boldsymbol{U}_{ab}, \tag{14}$$

$$\left[\frac{1}{\rho}\nabla \cdot (\rho \boldsymbol{R})\right]_a = \sum_b m_b \left(\frac{\boldsymbol{R}_a}{\rho_a} + \frac{\boldsymbol{R}_b}{\rho_b}\right) \cdot \nabla_a W_{ab}. \tag{15}$$

In the SPH method, an artificial compressibility technique is usually used to model the incompressible flow as a slightly compressible flow. The artificial compressibility considers that every theoretically incompressible fluid is actually slightly compressible. Therefore, it is feasible to model the incompressible flow by using a quasi-incompressible equation of state [9]:

$$p(\rho) = c^2(\rho - \rho_0), \tag{16}$$

where ρ_0 is a reference density, c is a numerical speed of sound. In order to reduce the density fluctuation down to one percent c is usually taken ten times larger than the maximum fluid velocity.

2.3. Solid boundary treatment (SBT)

Many solid boundary treatment algorithms have been proposed in different SPH literatures, such as repulsive SBT algorithm [8,25], dynamic SBT algorithm [3] and some others. In this paper dynamic boundary conditions are used. Solid boundaries are replaced by three layers of virtual particles (Fig. 1) which can remain fixed in position as fixed wall boundaries, or can move as a rigid body but not as separate particles. The density and pressure of the solid boundary particles are calculated as fluid particles. For a moving body the sum of all the reaction forces on the body particles from fluid particles is applied to the whole body.

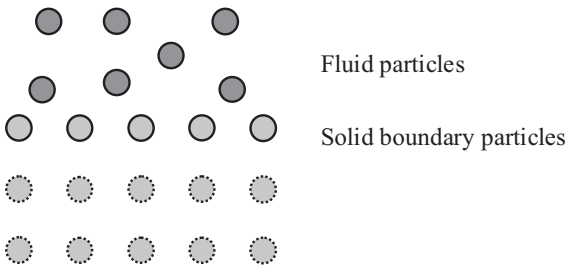

Fig. 1. Sketch of virtual particles for solid boundary.

3. Kernel functions for SPH method

3.1. Properties of a kernel function

The kernel function W is very important, because it determines the pattern to interpolate. There are several kernel functions used in literatures [3,19,20,26]. To choose or construct a suitable kernel, the following properties usually required:

(1) Delta function property, which means that the kernel function should tend to the Dirac delta function as the smoothing length tends to zero:

$$\lim_{h\to 0} W(\boldsymbol{r},h)=\delta(\boldsymbol{r}). \tag{17}$$

It is noted that in Eq. (1), if the kernel is replaced by the delta function, the interpolant can reproduces f exactly.

(2) Normalization condition, which means the kernel function must be normalized:

$$\int W(\boldsymbol{r},h)dV=1. \tag{18}$$

This property is also called unity condition. By satisfying this unity condition, a constant can be approximated exactly.

(3) Compact support property, which means the kernel function should have compact support:

$$W(\boldsymbol{r},h)=0, \quad |\boldsymbol{r}|\geqslant \kappa h, \tag{19}$$

where κ is usually a scalar factor which determines the area of the compact support domain.

Besides the above properties, other requirements may also required in different literatures such as monotonically decreasing condition, even function condition, smoothing condition, and non-negativity condition [21].

3.2. A new kernel function

Any function can be employed as a kernel function if it has the above properties. Considering these properties, we proposed a new kernel function as follows:

$$W(s,h)=\alpha_d\begin{cases} 4\cos\left(\frac{\pi}{\kappa}s\right)+\cos\left(\frac{2\pi}{\kappa}s\right)+3, & 0\leqslant s\leqslant\kappa \\ 0, & \kappa<s \end{cases}, \tag{20}$$

where $s=r/h$, $r=|\boldsymbol{r}|$ is the distance between two particles. α_d is the normalized coefficient in d dimensional space, and with the value of $\alpha_1=1/(6kh)$, $\alpha_2=\pi/[(3\pi^2-16)(kh)^2]$, and $\alpha_3=\pi/[(4\pi^2-30)(kh)^3]$ in one, two and three dimensional space, respectively. This new kernel function is referred to as double cosine function since it is consist of two cosine functions. The double cosine kernel function and its first two derivatives are shown in Fig. 2.

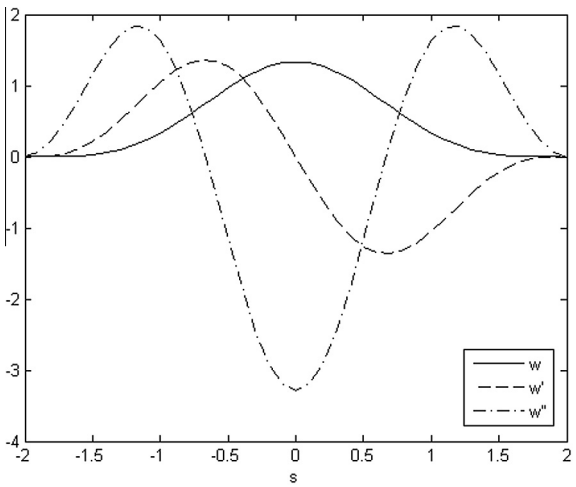

Fig. 2. Double cosine function ($\kappa=2$) and its first two derivatives.

It shall be noted that this double cosine kernel function is sufficiently smooth and can be arbitrarily differentiated. Also the support domain of the double cosine kernel function is adjustable as can be seen from Eq. (20), it is conveniently to change parameter κ to adapt the support domain.

Though there are diversified kernel functions in many different SPH literatures, the Gaussian and cubic spline kernel are more frequently used, and therefore we shall comparatively investigate the double cosine kernel function with the Gaussian and cubic spline kernel.

The Gaussian function is written as [2]:

$$G(r,h) = \alpha_d \exp[-(r/h)^2], \quad \alpha_d = 1/(\sqrt{\pi}h)^d. \tag{21}$$

The Gaussian kernel is sufficiently smooth even for high orders of derivatives, and is regarded as a golden selection by Monahan [17], sine it is very stable and accurate especially for disordered particles. It is, however, not really compact, as it never goes to zero theoretically, unless the support domain approaches infinity. Because it approaches zero very fast numerically, it is practically compact. This can also result in a large support domain with an inclusion of more particles for SPH approximation and therefore it is computationally very expensive.

Another most commonly used kernel is the cubic spline function:

$$M_4(s,h) = \alpha_d \begin{cases} (2-s)^3 - 4(1-s)^3, & 0 \leqslant s < 1 \\ (2-s)^3, & 1 \leqslant s < 2 \\ 0, & 2 \leqslant s \end{cases}, \tag{22}$$

where $s = r/h$, and $\alpha_1 = 1/(6h)$, $\alpha_2 = 15/(14\pi h^2)$, and $\alpha_3 = 1/(4\pi h^3)$ in one, two and three dimensional space, respectively. The cubic spline function works very well in many numerical simulations. However, a disadvantage is that the cubic spline kernel function is not smooth enough, as it is a piecewise function and its second derivative is a polyline.

By selecting different values of parameter κ, the shape of the double cosine function can approach the Gaussian and the spline functions (see Figs. 3 and 4). But it is important to note that the double cosine kernel function has a finite compact support domain rather an infinite support domain as in the Gaussian kernel function, and therefore it can be cost-effective compared with the Gaussian kernel function. Also, the second derivative of the double cosine kernel function is sufficiently smooth rather than a polyline as for the cubic spline kernel function, and therefore it can be numerically more stable [19].

3.3. Errors estimation

3.3.1. Error in kernel approximation

In early literatures, SPH has been usually referred to as a method of second order accuracy. This observation comes from Taylor series analysis of the kernel approximation of a field function (as expressed in Eq. (1)). For example, in one-dimensional space, using Taylor series expansion on f at x, and retain the first three terms, Eq. (1) can be further written as:

$$\begin{aligned} \langle f(x_a)\rangle &= \int f(x)W(x_a - x, h)dx \\ &= f(x_a)\int W(x - x_a, h)dV + f'(x_a)\int (x - x_a)W(x - x_a, h)dx + \frac{1}{2}f''(x_a)\int (x - x_a)^2 W(x - x_a, h)dx + R(h^3), \end{aligned} \tag{23}$$

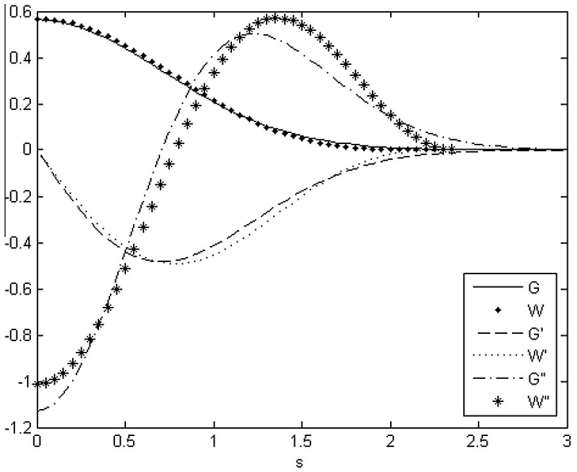

Fig. 3. Comparison of the double cosine function (denoted by W with $\kappa = 2.35$) and Gaussian function (denoted by G) and their first two derivatives.

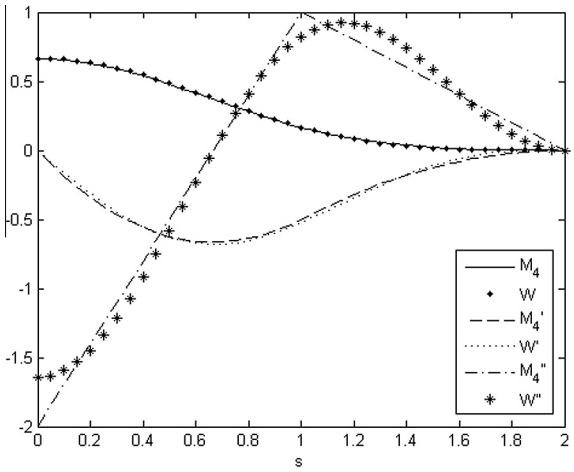

Fig. 4. Comparison of the double cosine function (denoted by W with $\kappa = 2$) and the cubic spline function (denoted by M_4) and their first two derivatives.

where R is the remainder of the Taylor series expansion. Considering the normalization condition (expressed in Eq. (18)), and even function condition (i.e. $\int xW(x,h)dx = 0$), it is easy to conclude that kernel approximation of a field function f can have second order accuracy. As f is a sufficiently smooth function, and its derivatives are associated with a finite magnitude. It can be further concluded that the accuracy of kernel approximation (expressed in Eq. (23)) is closely related to the second momentum of the kernel function:

$$\int x^2 W(x,h)dx. \tag{24}$$

Therefore, the second momentum of the kernel function can be regarded as an index to estimate the magnitude of the error of kernel approximation. A bigger value of the second momentum leads to bigger errors of the kernel approximation and vice versa.

For the double cosine, Gaussian, and cubic spline kernel functions, the second momentum can be analytically calculated. In one-dimension space, the double cosine function yields a smaller value ($0.6h^2$, when $\kappa = 2$) than the cubic spline function ($3h^2$) and the Gaussian function (h^2). Therefore as far as the kernel approximation is concerned, the double cosine kernel function can produce better results.

3.3.2. Errors in particle approximation

The computational accuracy of an SPH simulation is finally determined by the particle approximation of averaged summation over nearest neighboring particles, while the accuracy of particle approximation is not only related to the selection of kernel functions and smoothing length, but also closely related to the distribution of particles. In order to compare the performance of double cosine kernel function with Gaussian and cubic spline kernel function, SPH particle approximation results for an exponential function, $f = \exp(x)$, and a trigonometric function, $f = \sin x$, and their derivatives are analyzed.

The computational domain is $0 \leqslant x \leqslant 6$, and 101 particles are evenly distributed in the calculation. The smoothing length used is 1.2 times the initial particle spacing. The obtained SPH results by using double cosine, Gaussian and cubic spline kernel function are plotted in Figs. 5–8, in which SPH-D, SPH-G, and SPH-C denote SPH kernel approximations using double cosine, Gaussian and cubic spline kernel function, respectively.

Table 1 shows the relative errors in SPH particle approximation of $f = \exp(x)$ and $\sin x$, and their derivatives by using different kernels. The relative errors are calculated by following formulae:

$$E_f = \frac{\langle f \rangle - f}{f}, \tag{25}$$

$$E_{df/dx} = \frac{\langle df/dx \rangle - df/dx}{df/dx}. \tag{26}$$

It is seen that three kernel functions can all produce acceptable results. The relative errors obtained using the double cosine kernel for functions $f = \exp(x)$ and $\sin x$ are between those obtained using Gaussian and cubic spline kernel functions. For derivatives of $f = \exp(x)$ and $\sin x$, the relative errors obtained using double cosine kernel are also comparable with those obtained using Gaussian and cubic spline kernel functions. It should be noted that the relative errors obtained using double cosine kernel are all smaller than those obtained using cubic spline kernel.For the double cosine kernel function, it is also feasible to adjust parameter κ to obtain better particle approximation accuracy. From Figs. 9 and 10, we can see that the

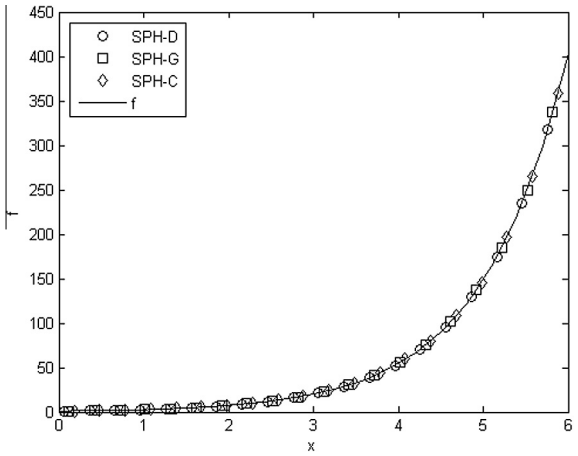

Fig. 5. SPH particle approximations of $f = \exp(x)$ by using different kernels.

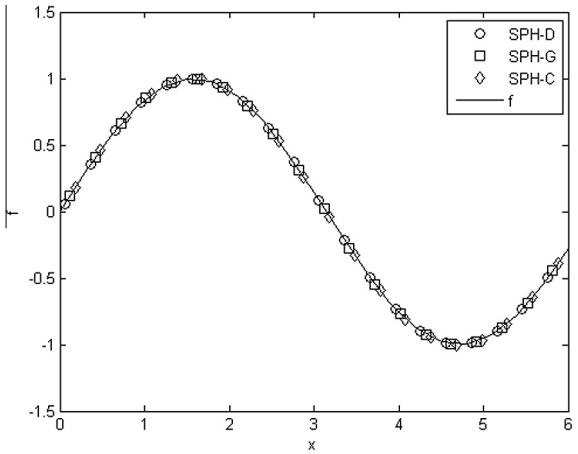

Fig. 6. SPH particle approximations of $f = \sin x$ by using different kernels.

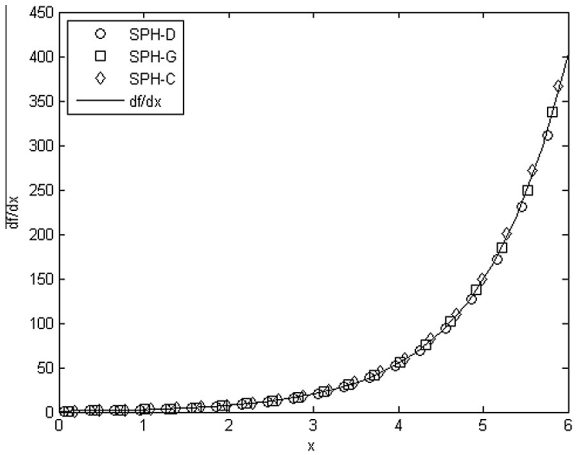

Fig. 7. SPH particle approximations of the derivative of $f = \exp(x)$ by using different kernels.

error increases when κ increases when 101 particles are used; but it decreases quickly when κ increases when enough particles (1001 particles) are employed. The relative errors in particle approximation of the derivatives of $f = \exp(x)$ and

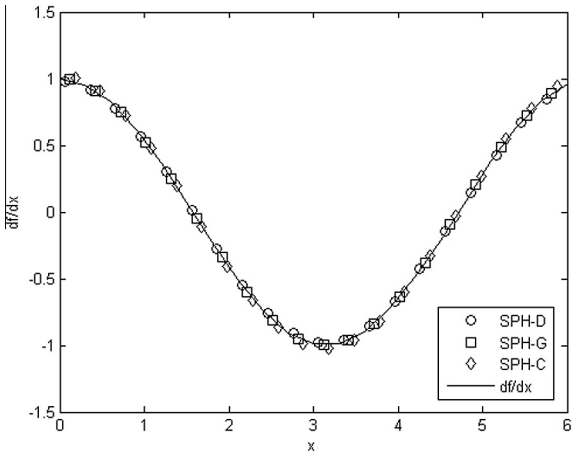

Fig. 8. SPH particle approximations of the derivative of $f = \sin x$ by using different kernels.

Table 1
Relative errors in kernel approximation.

Relative errors	E_f (%)		$E_{df/dx}$ (%)	
	$f = \exp(x)$	$f = \sin x$	$f = \exp(x)$	$f = \sin x$
Double cosine	0.154	0.011	1.76	1.91
Cubic spline	0.226	0.094	2.33	2.15
Gaussian	0.130	0.129	0.126	0.133

$f = \sin x$ are shown in Figs. 11 and 12. The relative errors reduce when the number of particles increases, but the decrease is not significant when $\kappa < 3$. Taking into account the above analysis about errors, it is reasonable to take κ in the range of $2 \leqslant \kappa \leqslant 3$. If $\kappa > 3$, much more particles will be needed, and the computational time will also increase quickly.

4. Numerical examples

In order to demonstrate the effectiveness of the newly proposed double cosine kernel function for practical engineering problems with incompressible, viscous fluids, two numerical examples are provided in this section. The first example is the dam-break flow and the second example is the water entry of a horizontal circular cylinder. Both examples involve free surface and even breaking waves, while water entry also involves fluid–solid interaction. These two numerical examples have been provided in different references to demonstrate the effectiveness of a certain numerical method, and can serve as good benchmark problems.

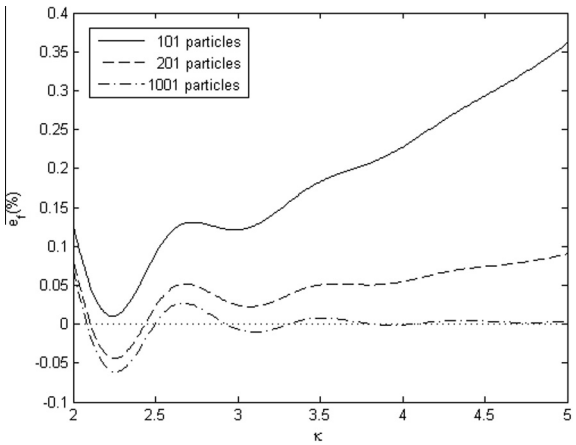

Fig. 9. Relative error of particle approximation for $\exp(x)$ vs. κ.

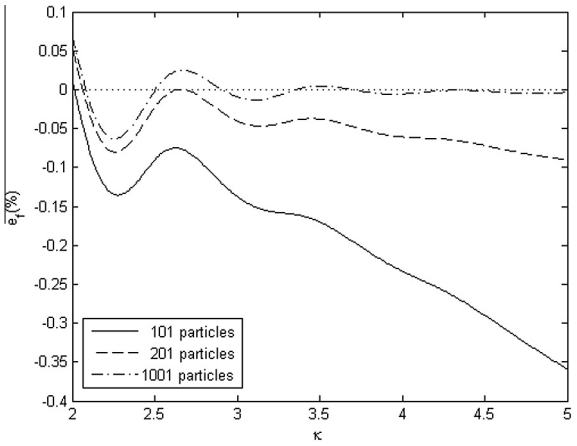

Fig. 10. Relative error of particle approximation for $\sin x$ vs. κ.

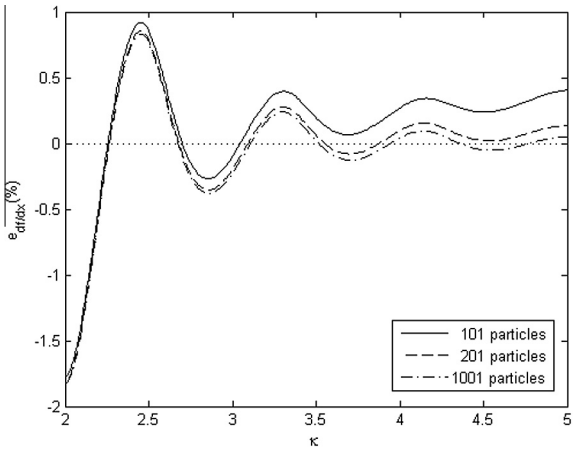

Fig. 11. Relative error of particle approximation for the derivative of $\exp(x)$ vs. κ.

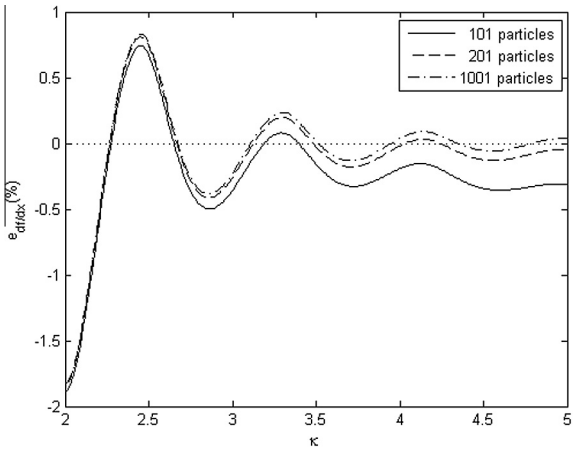

Fig. 12. Relative error of particle approximation for the derivative of $\sin x$ vs. κ.

Fig. 13. Initial configuration of the water column.

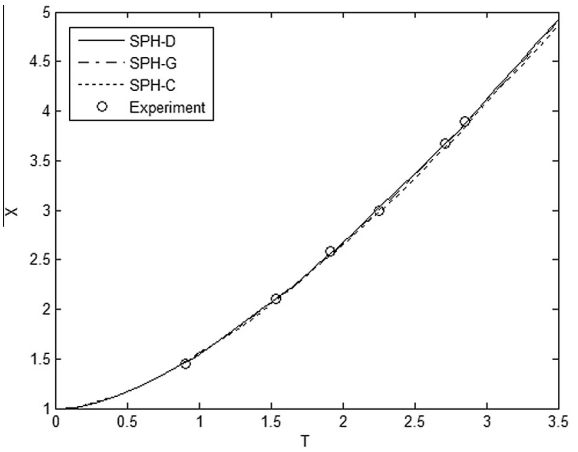

Fig. 14. The time dependent location of water front.

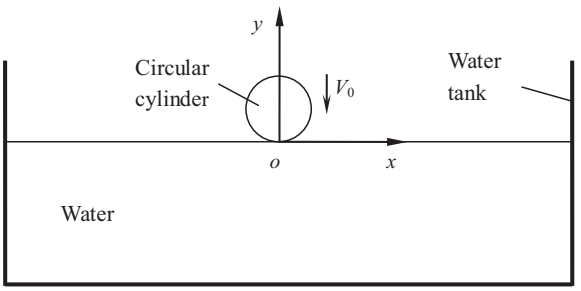

Fig. 15. Initial configuration of water entry of horizontal circular cylinder.

4.1. Dam-break flow

A 2-D dam-break flow is simulated in this section. The geometry is shown in Fig. 13. The water column is 0.2 m high (H) and 0.1 m wide (D), and the length of the dry bed is 0.9 m. At initial time, the water column is at rest, then it move under the action of gravity. The time step is 5.0×10^{-6} s.

Fig. 14 shows the water location versus time for both simulation and experimental results [27]. The non-dimensional time and location are normalized by $T = t\sqrt{2g/D}$ and $X = x/D$, respectively. It can be observed that the SPH results obtained using the double cosine kernel are very close to those obtained using Gaussian and cubic spline kernel functions, while all SPH results agree well with experimental observations.

4.2. Water entry of a horizontal circular cylinder

A neutrally buoyant cylinder case is simulated in this section. The water entry of a horizontal circular cylinder is simplified in two dimensions (Fig. 15). The results are compared with the experiments results obtained by Greenhow and Lin [28]. The diameters of both cylinders are 0.11 m, both cylinders are dropped from a height of 0.5 m, which is measured from the

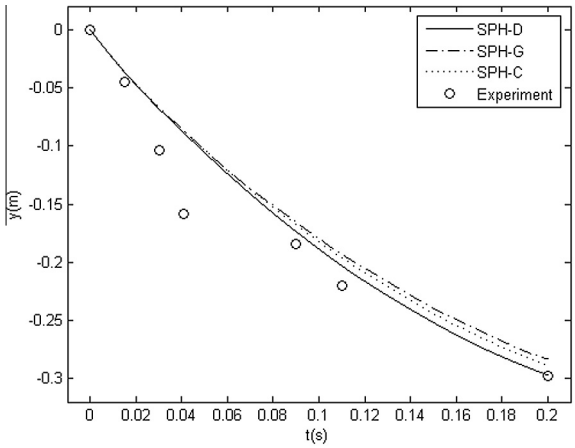

Fig. 16. Depth of penetration during water entry of a cylinder.

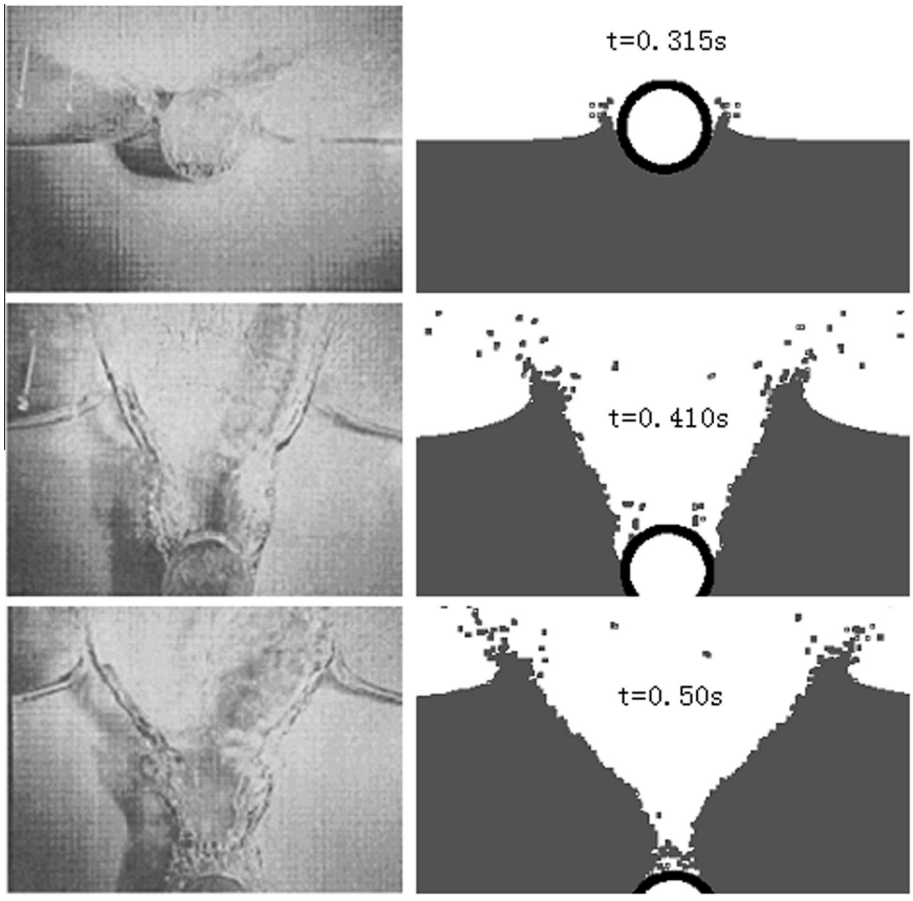

Fig. 17. Comparison of free surface profiles by the SPH (right) with the experiment [28] (left).

center of the water surface. So the initial water entry speed is $V_0 = 2.955$ m/s. The simulation started at the moment when the cylinder just touches the water surface. The time step is 5.0×10^{-6} s.

Fig. 16 shows the depth of penetration versus time during water entry of a cylinder. It is seen that all SPH results are comparable with experimental observations, while the double cosine kernel performs much better and is closer to experimental observations than Gaussian and cubic spline kernel functions.

Fig. 17 shows the free surface profile of a cylinder at several time instants. Water is splashed on both side of the cylinder and an air cavity is formed above the cylinder. However, the numerical results did not capture the cavity very well.

5. Conclusions

In this paper, a new kernel function for the smoothed particle hydrodynamics method has been proposed. This new kernel function consists of two cosine functions and therefore is referred to as double cosine kernel function. Compared with the Gaussian and cubic spline kernel functions, the double cosine kernel function has special advantages.

Firstly, unlike the frequently used cubic spline kernel function, which has a piecewise linear second derivative, the double cosine kernel function is sufficiently smooth. As the stability of SPH simulations is related to the second derivative of the kernel function, a smoother function generally results in more stable SPH formulations.Secondly, the support domain of the double cosine kernel function is adjustable through changing the scalar parameter κ. Changing different values of κ can lead to different shapes of the kernel function. Increasing κ can lead to sharper profiles while reducing κ leads to more flattened profiles with more particles and therefore more computational effort. For balancing computational accuracy and computational effort, a range of 2.0–3.0 for κ is recommended.

Thirdly, the double cosine kernel function has a smaller second momentum. This means that as far as kernel approximations are concerned, the double cosine kernel function has better accuracy than the cubic spline and Gaussian functions.

For SPH particle approximation, results obtained using the double cosine kernel functions are comparable to those obtained using Gaussian and cubic spline kernel functions, while for some cases, the double cosine kernel performs the best. This has also been demonstrated in the two numerical examples of dam-break and water entry.

Acknowledgment

This work has been supported by the National Natural Science Foundation of China (11302237, 11172306) and National Defense Innovation Funds of the Chinese Academy of Sciences (Y175031XML).

Appendix A. Supplementary data

Supplementary data associated with this article can be found, in the online version, at http://dx.doi.org/10.1016/j.apm.2013.12.001.

References

[1] L.B. Lucy, A numerical approach to the testing of the fission hypothesis, Astron. J. 82 (1977) 1013–1024.
[2] R.A. Gingold, J.J. Monaghan, Smoothed particle hydrodynamics: theory and application to non-spherical stars, Mon. Not. R. Astron. Soc. 181 (1977) 375–389.
[3] G.R. Liu, M.B. Liu, Smoothed Particle Hydrodynamics: A Meshfree Particle Method, World Scientific, Singapore, 2003.
[4] M.B. Liu, G.R. Liu, Smoothed particle hydrodynamics (SPH): an overview and recent developments, Arch. Comput. Methods Eng. 17 (2010) 25–76.
[5] P.W. Cleary, M. Prakash, J. Ha, N. Stokes, C. Scott, Smooth particle hydrodynamics: status and future potential, Int. J. Prog. Comput. Fluid Dyn. 7 (2007) 70–90.
[6] J.J. Monaghan, Smoothed particle hydrodynamics, Rep. Prog. Phys. 68 (2005) 1703–1759.
[7] J.J. Monaghan, R.A. Gingold, Shock simulation by the particle method SPH, J. Comput. Phys. 52 (1983) 374–389.
[8] J.J. Monaghan, Simulating free surface flows with SPH, J. Comput. Phys. 110 (1994) 399–406.
[9] J.P. Morris, P.J. Fox, Y. Zhu, Modeling low Reynolds number incompressible flows using SPH, J. Comput. Phys. 136 (1997) 214–226.
[10] Y.M.L. Edmond, S. Shao, Simulation of near-shore solitary wave mechanics by an incompressible SPH method, Appl. Ocean Res. 24 (2002) 275–286.
[11] S. Shao, E.Y.M. Lo, Incompressible SPH method for simulating Newtonian and non-Newtonian flows with a free surface, Adv. Water Resour. 26 (2003) 787–800.
[12] S. Koshizuka, Y. Oka, Moving-particle semi-implicit method for fragmentation of incompressible fluid, Nucl. Sci. Eng. 123 (1996) 421–434.
[13] S. Koshizuka, A. Nobe, Y. Oka, Numerical analysis of breaking waves using the moving particle semi-implicit method, Int. J. Numer. Meth. Fluids 26 (1998) 751–769.
[14] A. Colagrossi, M. Landrini, Numerical simulation of interfacial flows by smoothed particle hydrodynamics, J. Comput. Phys. 191 (2003) 448–475.
[15] X.Y. Hu, N.A. Adams, An incompressible multi-phase SPH method, J. Comput. Phys. 227 (2007) 264–278.
[16] N. Grenier, M. Antuono, A. Colagrossi, D. Le Touzé, B. Alessandrini, An Hamiltonian interface SPH formulation for multi-fluid and free surface flows, J. Comput. Phys. 228 (2009) 8380–8393.
[17] J.J. Monaghan, Smoothed particle hydrodynamics, Ann. Rev. Astron. Astrophys. 30 (1992) 543–574.
[18] J.W. Swegle, D.L. Hicks, S.W. Attaway, Smoothed particle hydrodynamics stability analysis, J. Comput. Phys. 116 (1995) 123–134.
[19] J.P. Morris, Analysis of Smoothed Particle Hydrodynamics With Applications, Monash University, 1996.
[20] D.A. Fulk, D.W. Quinn, An analysis of 1-D smoothed particle hydrodynamics kernels, J. Comput. Phys. 126 (1996) 165–180.
[21] M.B. Liu, G.R. Liu, K.Y. Lam, Constructing smoothing functions in smoothed particle hydrodynamics with applications, J. Comput. Appl. Math. 155 (2003) 263–284.
[22] J. Ha, A Numerical Study of the Application of Radial Basis Function and Generalised Smoothed Particle Hydrodynamics to CFD, in: 15th Australasian Fluid Mechanics Conference, The University of Sydney, Sydney, Australia, 2004.
[23] H. Jin, X. Ding, On criterions for smoothed particle hydrodynamics kernels in stable field, J. Comput. Phys. 202 (2005) 699–709.
[24] R.M. Cabezón, D. García-Senz, A. Relaño, A one-parameter family of interpolating kernels for smoothed particle hydrodynamics studies, J. Comput. Phys. 227 (2008) 8523–8540.
[25] J.J. Monaghan, A. Kos, Solitary waves on a Cretan beach, J. Waterw. Port Coastal Ocean Eng. 125 (1999) 250–259.
[26] T. Belytschko, Y. Krongauz, D. Organ, M. Fleming, P. Krysl, Meshless methods: an overview and recent developments, Comput. Methods Appl. Mech. Eng. 139 (1996) 3–47.
[27] C. Jin, K. Xu, A unified moving grid gas-kinetic method in Eulerian space for viscous flow computation, J. Comput. Phys. 222 (2007) 155–175.
[28] M. Greenhow, W.M. Lin, Nonlinnear Free Surface Effects: Experiments and Theory, Department of Ocean Engineering, MIT, 1983.