Environ Fluid Mech (2016) 16:193–208
DOI 10.1007/s10652-015-9424-1

ORIGINAL ARTICLE

# Computation of the Basset force: recent advances and environmental flow applications

**Patricio A. Moreno-Casas**[1,2] · **Fabián A. Bombardelli**[1]

Received: 23 December 2014 / Accepted: 7 July 2015 / Published online: 17 July 2015


**Abstract** When numerically integrating the equation describing the motion of a particle in a carrier fluid, the computation of the Basset (history) force becomes by far the most expensive and cumbersome, as opposed to forces such as drag, virtual mass, lift, buoyancy and Magnus. The expression representing the Basset force constitutes an integro-differential term whose standard integrand is singular when the upper integration limit is enforced. These shortcomings have led some researchers to either disregard or outright neglect the contribution of the Basset force to the total force, even in those cases where it may yield to important errors in the determination of particle trajectories in the computation of sediment transport and other environmental flows. This work is devoted to review four recent contributions associated with the computation of the Basset force, and to compare their proposals to diminish the inherent problems of the term integration. All papers, except one, use variants of a window-based approach; the most recent contribution, in turn, employs a specialized quadrature to increase the accuracy of the computation. An analysis was carried out to compare CPU computation times, rates of convergence and accuracy of the approximations versus a known analytical solution. All methods provide sound solutions to the issues associated with the computation of the Basset force; further, a road map to select the best solution for each given problem is provided. Finally, we discuss the implications of the techniques for the simulation of sediment transport processes and other environmental flows.

✉ Patricio A. Moreno-Casas
patriciomoreno@miuandes.cl

Fabián A. Bombardelli
fabianbombardelli2@gmail.com

[1] Department of Civil and Environmental Engineering, University of California, Davis, One Shields Avenue, 2001 Ghausi Hall, Davis, CA 95616, USA

[2] Facultad de Ingeniería y Ciencias Aplicadas, Universidad de los Andes, Monseñor Álvaro del Portillo 12.455, Las Condes, Santiago, Chile

**Keywords** History force · Basset force · Maxey–Riley equation · Quadrature · Accuracy · Computation time

## 1 Introduction

Multi-phase flows are abundant in natural environments as well as man-made devices. They appear, for instance, in air–water flows past spillways in dams (see, for example, [1, 7]), pneumatic transport of solids in the mining industry, sediment transport in streams and rivers [5, 15], and solid particle-air flows in the atmosphere [11]. Consequently, they have become important subject matters in diverse branches of science and engineering [6, 11, 14]. Multi-phase flows are constituted by a carrier phase (water or air), in which there are either bubbles (in case of water) and/or solid particles, which form the "disperse phase". In general, either bubbles or solid material are called "particles".

In order to numerically predict the behavior of multi-phase flows, it is often required to formulate mass and momentum equations for each of the phases separately [5, 11, 14, 39]. Another common multi-phase, mathematical approach is to follow each particle in a so-called Lagrangian description. In this description, whereas the conservation of mass of the disperse phase becomes enforced automatically (because the motion of each particle is addressed explicitly [11, 14], a momentum equation needs to be provided for the particles). To that end, a version of the second Newton's law is employed, adapted to represent particles of finite size. Historically, one of such equations is the so-called Basset–Boussinesq–Oseen (BBO) expression, valid for Reynolds numbers much smaller than unity for a particle slowly accelerating in a *still fluid*. This equation includes three forces: Stokes drag, added (virtual) mass and Basset forces. However, when considering a small rigid spherical particle moving also in low Reynolds numbers in *non-uniform unsteady flow*, the equation presented by [16, 26] is customarily used. The equation of Maxey–Riley–Gatignol (MRG) is as follows:

$$m_p \frac{d\vec{u}_p}{dt} = m_f \frac{D\vec{u}_f}{Dt} - \frac{m_f}{2}\left(\frac{d\vec{u}_p}{dt} - \frac{D\vec{u}_f}{Dt}\right) - 6\pi a\rho\nu\left(\vec{u}_p - \vec{u}_f\right) + \left(m_p - m_f\right)\vec{g} - 6a^2\rho\sqrt{\pi\nu}\int_{t_0}^{t} \frac{1}{\sqrt{t-\tau}}\left(\frac{d\vec{u}_p}{d\tau} - \frac{d\vec{u}_f}{d\tau}\right)d\tau \tag{1}$$

In the above equation, $\vec{u}_p$ and $m_p$ are the velocity vector and mass of the particle, respectively; $\vec{u}_f$ and $m_f$ represent the velocity vector and mass of fluid "excluded" by the particle; $t$ is the time coordinate; $a$ is the particle radius; $\rho$ depicts the density of fluid; $\vec{g}$ represents the acceleration of gravity vector; and $\tau$ is a dummy integration variable. The expression $d/dt$ denotes the time derivative following the moving particle, whereas $D/Dt$ indicates the time derivative following a fluid parcel. The terms on the right-hand side of Eq. (1) represent the following forces: Pressure gradient of the undisturbed flow, virtual (added) mass, Stokes drag, submerged weight, and the Basset (history) force. The original MRG equation includes second-order terms known as the Faxén terms, which work as a correction for the curvature of the velocity profile [10]. The Faxén terms are neglected in Eq. (1) since they are usually smaller than the other terms in Eq. (1) [33]. It is important to mention at this point that most particles are not spherical and that, as a consequence, the above equation is an approximation to reality. Hydrodynamic effects generated by the non-

spherical nature of the particle are not taken into account in this type of models. The MGR equations were extended by [16, 37] to the case of homogeneous and inhomogeneous compressible flow. Sridhar and Katz [43] in turn used experimental observations of microscopic bubbles to infer the unsteady forces embedded in Eq. (1).

Whereas the forces related to the pressure gradient and submerged weight are obvious to understand from a physical standpoint, the forces of virtual mass and Basset are less evident. The virtual mass force is associated with the work needed from the moving particle to accelerate the fluid displaced by its body when translating to a new position; the name "added mass" refers to the fact that this force is equivalent to the effect of adding an extra mass in the accelerating particle [11]. The Basset term accounts for the temporal delay in the development of the boundary layer surrounding the particle's surface as a consequence of changes in the relative velocity [4, 11, 28].

Armenio and Fiorotto [2] found that the Basset force is important in sediment-laden flows when the particle Reynolds number ($Re_p = w_s a/\nu$, where $w_s$ represents the particle fall (limit) velocity, and $\nu$ depicts the kinematic viscosity of the fluid) is of the order or smaller than 1, for a large range of density ratios. The importance of the Bassset force thus depends on the particle size.

Different variants of the MRG equation have been developed in recent decades in order to account for particle motions in rivers as bed-load, which involve the extension of the terms in the expression above to finite Reynolds numbers [4, 8, 23–25, 33, 48]. Their efforts also consisted in some cases in adding new terms known to work relatively well outside of the creeping flow regime. From the terms of Eq. (1), the quasi-steady drag force needs to be modified [22]. As the Reynolds number increases, the drag becomes non-linear. In the latter case the drag force can be rewritten in terms of the drag coefficient, $C_D$, for which there are several empirical or semi-empirical expressions available in the literature (i.e., [8, 18, 50]). The Basset force could be also modified for Reynolds numbers larger than one, although there is no agreement among researchers on how to approach the problem (see Sect. 2.1 for further discussion). The gradient of the velocity close to the bed of the stream may be significant enough causing lift of the particle; thus, a new term representing the lift force needs to be added. Another type of lift caused by the rotation of the moving particle needs to be considered through the addition of a term representing the so-called Magnus force.

The Basset term can be presented in a generalized way, as follows:

$$F_B = 6a^2\rho\sqrt{\pi\nu}\int_{t_0}^{t} K(t-\tau)\frac{d}{d\tau}(f(\tau))\,d\tau \tag{2}$$

where $K(t-\tau)$ is called the Basset kernel and it is usually represented by the standard expression $1/\sqrt{t-\tau}$. The second factor, $f(\tau)$, is the relative velocity between phases.

In most cases, an analytical solution of Eq. (1) for low and finite Reynolds numbers is not possible, requiring the assistance of numerical techniques. Whereas most terms included in Eq. (1) can be computed using well-known numerical integration procedures such as the standard Runge–Kutta method, or even the Euler method, the integro-differential Basset term poses some challenges to its computation. First, the term includes the derivative of the relative velocity between the two phases, which should be integrated since the beginning of times by definition. This translates to high memory requirements stemming from the need to store the changes in relative acceleration of each simulated particle, and to long integration times, reducing its applicability on sediment-transport or bubbly-

flow simulations when the motion of several hundred thousand particles need to be considered. Another important difficulty regarding the numerical integration of the Basset term is the singularity encountered when the upper limit is enforced in the integrand. The difficulties introduced by the Basset term have induced many authors to either completely disregard it from the particle equation of motion or to consider it negligible, which may lead to large computational errors for small particles [4, 35].

This paper critically compares major recent advances brought by four works: [4, 12, 13, 46], which involve a fractional derivative approach, window kernels and higher-order numerical integration schemes. To this end, a test integral with a known analytical solution is used to compare the approaches by the last three contributions. Differences in CPU time of computation (computational cost), rate of convergence and accuracy are analyzed. Uncertainty of the predictions is addressed via a comparison of different kernels used in the computation of the force. Finally, we provide an assessment of the implications of these results on the computations of motion of particles in environmental flows. To the best of our knowledge, this is the first time a comparison of this nature is undertaken, providing a general framework for the assessment of previous contributions and the analysis of the Basset force for further developments.

## 2 Recent discussions regarding the Basset force

The latest advances in the computation of the Basset force focus on two different aspects: (a) the validity of different kernels for the force to reflect the physics of the problem; (b) numerical integration methods able to overcome the issues associated with the term. In what follows, we discuss these issues in detail.

### 2.1 Basset kernels

Regarding the first aspect, a kernel applicable to the finite-Reynolds-number range was proposed by Mei and Adrian [27]:

$$K(t-\tau)=\left\{(t-\tau)^{1/(2c_1)}+\left[\sqrt{\frac{\pi}{\nu}}\frac{\left|\vec{u}_f-\vec{u}_p\right|^3(t-\tau)^2}{2\nu f_h}\right]^{1/c_1}\right\}^{-c_1} \tag{3}$$

$$f_h=\left[0.75+c_2\left(\frac{2a\left|\vec{u}_f-\vec{u}_p\right|}{\nu}\right)\right]^3 \tag{4}$$

In Eqs. (3) and (4), $c_1 = 2$ and $c_2 = 0.105$. Other authors made modifications to the values of the two constants originally proposed by Mei and Adrian [13, 19, 20].

There is no agreement within the scientific community on whether the standard Basset term should change in the non-linear drag range [13, 33]. The work by Mei and Adrian [27] suggests that for short-time periods the decay rate of the Basset kernel is proportional to $t^{-1/2}$, as in the standard expression; however, for long-time periods the decay rate is proportional to $t^{-2}$ (a much faster decay rate). In their work, Mei and Adrian proposed a new kernel that takes into account the above, by adding both decay expressions.

Lawrence and Mei [20] found that the long-term decay rate was valid for a solid particle falling from rest till terminal velocity, but that it was different for particles stopping or experimenting flow reversal. The works by [13, 22] use the kernel proposed by [27] (Eqs. (3) and (4)) with a modification of the value of the constants ($c_1 = 2.5$ and $c_2 = 0.2$)

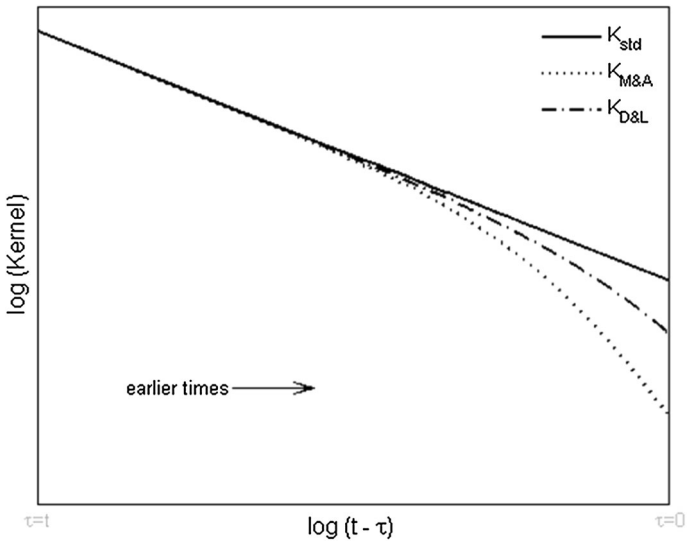


**Fig. 1** Schematic comparison of the Basset standard kernel, $\boldsymbol{K_{std}}$, versus the kernel proposed by Mei and Adrian [27], $\boldsymbol{K_{M\&A}}$, and a modified version of the Mei and Adrian's kernel using the constant values proposed by Dorgan and Loth [13], $\boldsymbol{K_{D\&L}}$, as a function of time for $\mathbf{2}\boldsymbol{a}\left|\vec{\boldsymbol{u}}_{\boldsymbol{f}} - \vec{\boldsymbol{u}}_{\boldsymbol{p}}\right|/\upsilon = \mathbf{10}$. A value of υ equal to $10^{-6}$ $m^2/s$ was used

for better approximation to experimental results of Moorman [29]. A comparison of the standard kernel with that proposed by Mei and Adrian [27], and its modified version using the values suggested by [13], is shown in Fig. 1. The clear difference in behavior is apparent. It is important to remember that the time evolution goes from right to left on Fig. 1, since the x-axis is evaluated as $t - \tau$. Further comparisons of kernels were presented by Wakaba and Balachandar [47].

## 2.2 Numerical approximation of the Basset term

In the past few years, attempts to overcome the difficulties in the integration of the Basset force have been made [4, 12, 13, 46]. These proposals have been put forward with the following goals: (1) to avoid the singularity of the integration process; (2) to reduce the computational time (cost) and memory requirements of the term; and (3) to increase the accuracy of the integration. The following sections discuss these three aspects in detail by comparing the different approaches taken by [4, 12, 13, 46].

### *2.2.1 Singularity*

The Basset term may be interpreted as a member of the family of Volterra-type equations [9]. Several methods have been developed to numerically solve cases of integration in which the integrand is singular. Press et al. [38] discuss several open-formula quadratures (the integrand is not evaluated at the endpoints) to accomplish this, such as the second order Euler-MacLaurin method, and the second order Newton–Cotes scheme. The problem with these latter quadratures is that they generate solutions with low temporal accuracy ($O(h^{1/2})$) and slow convergence [4, 12, 46].

In order to circumvent the singularity problem, clever schemes need to be constructed to explicitly modify the integrand, or simply to change the upper limit of integration and separate the Basset term in two parts [10], while at the same time yielding solutions to the Basset integral with temporal accuracy larger than that obtained by the quadrature methods mentioned above.

The work by Bombardelli et al. [4] consists on the use of a semi-derivative approach based on the work by Tatom [45]. Tatom noted that the Basset integro-differential equation may be transformed via the Riemann–Liouville integral definition into a semi-derivative expression:

$$\int_{t_0}^{t} \frac{1}{\sqrt{t-\tau}} \frac{d}{d\tau}(f(\tau))d\tau = \Gamma\left(\frac{1}{2}\right) \frac{d^{-0.5}\left(\frac{df(\tau)}{d\tau}\right)}{[d(t-t_0)]^{-0.5}} \tag{5}$$

where $\Gamma(\cdot)$ is the gamma function. The second factor on the right-hand side of (5) corresponds to a semi-derivative. This semi-derivative can be expressed by using the following series expansion for an arbitrary function $g$ [36]:

$$\frac{d^q g}{[d(t-t_0)]^q} = \lim_{N\to\infty}\left\{\left(\frac{t-t_0}{N}\right)^{-q} \frac{1}{\Gamma(-q)} \sum_{k=0}^{N-1} \frac{\Gamma(k-q)}{\Gamma(k+1)} g\left(t - \frac{k(t-t_0)}{N}\right)\right\} \tag{6}$$

where $q = -0.5$ in this case, and $N$ is the number of terms considered in the semi-derivative sum. The paper by Tatom only notices the transformation given in Eq. (5), but it does not address either pure numerical results for the integral or its applications. Bombardelli et al. [4] replaced expression (6) into Eq. (5) giving the final numerical approximation of the Basset force. The beauty of this approach lies on the use of the evaluation of the semi-derivative using a series expansion, which eliminates the $t-\tau$ term from the denominator, and hence solves the singularity problem.

Van Hinsberg et al. [46] circumvented the singularity problem with the use of a trapezoidal-based method (the ordinary trapezoidal rule is not suited for singular integrals), and by approximating the derivative of the relative velocity of the moving particle with a linear interpolant $P_1(t)$, to later integrate the product $K(t-\tau)P_1(\tau)$. The integral can be evaluated using the following expression:

$$\begin{aligned}\int_{t_0}^{t} K(t-\tau)\frac{d}{d\tau}(f(\tau))d\tau &= \frac{4}{3}\frac{d}{d\tau}(f(\tau_0))\sqrt{h} + \frac{d}{d\tau}(f(\tau_N))\frac{\sqrt{h}\left(N-\frac{4}{3}\right)}{(N-1)\sqrt{N-1}+\left(N-\frac{3}{2}\right)\sqrt{N}} \\ &+ \sqrt{h}\sum_{k=1}^{N-1}\frac{d}{d\tau}(f(\tau_k))\left(\frac{k+\frac{4}{3}}{(k+1)\sqrt{k+1}+\left(k+\frac{3}{2}\right)\sqrt{k}} + \frac{k-\frac{4}{3}}{(k-1)\sqrt{k-1}+\left(k-\frac{3}{2}\right)\sqrt{k}}\right)\end{aligned} \tag{7}$$

where $h$ is the time step for the computation and $\tau_k = t - kh$, with $k = 0, 1, 2, \ldots, N$.

In turn, Daitche's methodology [12] approximates the derivative of the relative velocity with a polynomial (not the whole integral) and computes the resulting integral analytically through a simple integration by parts which brings an initial condition. In doing that, Daitche was able to drop the derivative from the original kernel and take advantage of simplified numerical schemes with which a generalized procedure for quadrature schemes of higher order was established:

$$\int_{t_0}^{t} K(t-\tau)\frac{d}{d\tau}(f(\tau))d\tau + K(t-t_0)f(t_0) = \frac{d}{dt}\int_{t_0}^{t} K(t-\tau)(f(\tau))d\tau \tag{8}$$

The second term in the left-hand side of Eq. (8) assumes that the particle has an initial velocity different from that of the carrier fluid. When considering the same initial velocity for both, fluid and particle, the latter term vanishes. Finally, the quadrature scheme is as follows:

$$\int_{t_0}^{t} K(t-\tau)G(\tau)d\tau = \sqrt{h}\sum_{j=0}^{n}\beta_j^n G(\tau_{n-j}) + O(h^m)\sqrt{t-t_0} \tag{9}$$

where $n$ is the number of intervals of the approximation of the integral, and the order of approximation of the scheme $m$ will vary with the definition of the coefficient $\beta_j^n$. The definition of the coefficients is rather lengthy, and for convenience it will not be shown in this paper. However, the format is similar to the following:

$$\beta_k^n = \frac{4}{3}\begin{cases} 1, & k=0 \\ (k-1)^{3/2}+(k+1)^{3/2}-2k^{3/2}, & 0<k<n \\ (n-1)^{3/2}-n^{3/2}+\frac{6}{4}\sqrt{n}, & k=n \end{cases} \tag{10}$$

which corresponds to a second order solution. For further details of the coefficients with error orders $O(h^3)$ and $O(h^4)$, the reader is referred to [12]. In Eqs. (9) and (10), $\tau_k = t_0 + hk$; in turn, $n = (t-t_0)/h$.

Bombardelli et al.'s methodology [4] can only be used with the standard kernel, while the approaches presented by van Hinsberg et al. [46] and Daitche [12] can numerically approximate the Basset expression by either using the standard or the modified kernel presented by Mei and Adrian [27].

### 2.2.2 *Computational cost and memory requirements of the integration*

Regarding the computational time associated with the integration of the Basset force, recent progress includes the so-called "window-based approach", or equivalently, "memory time period". This concept is associated with the idea according to which the acceleration of the particle at a given time loses correlation with previous values of acceleration as time passes by. This concept is also in agreement with the discussions in [23] where the existence of a viscous time scale beyond which there is no significant contribution from the integral is analyzed. The issue is how to express mathematically such loss of correlation. To the best of the authors' knowledge, the completely independent works by [4, 13, 17] were the first works aiming at reducing the computation time by identifying a time window in which the Basset force needs to be considered. The window method presented by Dorgan and Loth [13] was influenced by the findings of Mordant and Pintot [30], who realized that the Basset force at finite Reynolds numbers was well represented by the creeping flow kernel (Eq. (1)) up to a finite time interval or time window, while after that period it decayed exponentially and eventually became negligible. This suggests that the Basset integral may be partitioned into two integrals by a finite time usually called $t_{win}$. The integral evaluating the earlier (or older) times is defined by an approximation kernel, which is called the *tail* kernel ($K_{tail}(t-\tau)$), different from the standard Basset kernel; the integral evaluating later (or more recent) times is defined by a *window* kernel ($K_{win}(t-\tau)$), where usually $K_{win}(t-\tau) = K(t-\tau)$:

$$\int_{t_0}^{t} K(t-\tau)\frac{d}{d\tau}(f(\tau))d\tau = \int_{t_0}^{t-t_{win}} K_{tail}(t-\tau)\frac{d}{d\tau}(f(\tau))d\tau + \int_{t-t_{win}}^{t} K_{win}(t-\tau)\frac{d}{d\tau}(f(\tau))d\tau \tag{11}$$

The main idea implicit in Eq. (11) is to approximate the tail using a computationally cheap expression for $K_{tail}(t-\tau)$, which translates into a reduction in the use of computer memory and in computational time. For instance, this methodology was recently employed by Rechiman et al. [40] to compute the Basset force acting on strongly collapsing, sonoluminescent bubbles.

González et al. [17] and Bombardelli et al. [4] adopted a zero value for $K_{tail}$, assuming a complete lack of correlation of particle velocities with old ones *in the transport of solids as bed-load*. Since saltating particles move in a hopping motion, constantly colliding with the channel wall [4, 8, 31] it seems reasonable to link $t_{win}$, the memory time period (called $T_{back}$ in [4]), to a specific number of jumps for which the window kernel is computed.

Figure 2 sketches the numerical evaluation of the window and tail kernels. The approximation error of the window-based approach depends on the chosen $t_{win}$, the numerical integration method used to approximate the window kernel, and the numerical expression chosen to compute the tail kernel [13, 46].

Bombardelli et al. [4] compared their results to the second Euler–Maclaurin summation formula and to the decomposition by Brush et al. [10], using the Simpson quadrature, and found that the semi-derivative approach converged much faster than the other two, consequently reducing the computational time in about 20 %. Applying in addition the memory time period, or window-based method, reported a reduction of the simulation time to 70–90 % from the original run time.

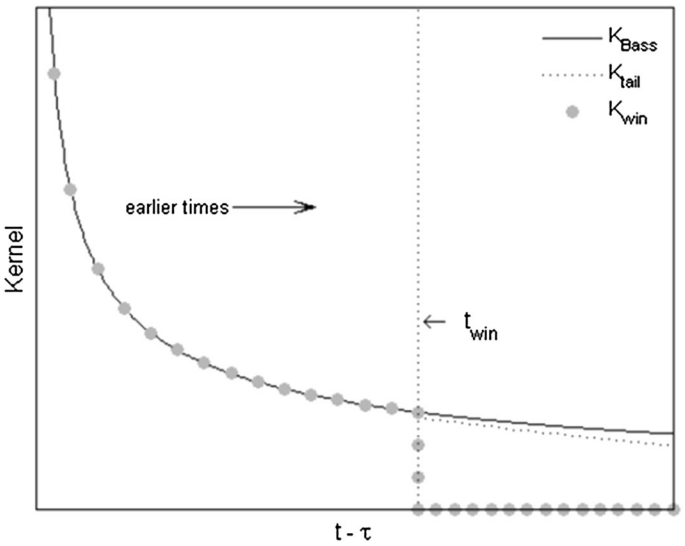


**Fig. 2** Schematic of the numerical evaluation of the standard Basset kernel ($\mathbf{K}_{Bass}$), window kernel ($\mathbf{K}_{win}$) and tail kernel ($\mathbf{K}_{tail}$) for an arbitrary $\mathbf{t}_{win}$ as a function of time ($\mathbf{t}-\boldsymbol{\tau}$). The standard Basset kernel (*solid black line*) is approximated by the window kernel (*grey circles*), from $\boldsymbol{t}$ up to $\boldsymbol{t}-\boldsymbol{t}_{win}$, and the tail kernel (*broken black line*) at times earlier than $\boldsymbol{t}-\boldsymbol{t}_{win}$. Modified from Dorgan and Loth [13]

Van Hinsberg et al. [46] applied a window-based approach through the use of less expensive functions, to approximate $K_{tail}(t-\tau)$, of exponential nature, as follows:

$$\int_{t_0}^{t-t_{win}} K_{tail}(t-\tau)\frac{d}{d\tau}(f(\tau))d\tau = \sum_{i=1}^{m} a_i F_i(t) \tag{12}$$

where the tail kernel is represented with direct ($F_{i=di}(t)$) and a recursive ($F_{i-re}(t)$) components. These components can be computed as follows:

$$F_i(t) = F_{i-di}(t) + F_{i-re}(t) \tag{13}$$

in which

$$\begin{aligned} F_{i-di}(t) = 2\sqrt{et_i}\exp\left(-\frac{t_{win}}{2t_i}\right)\left\{\frac{d}{d\tau}(f(\tau_N))\left[1-\varphi\left(-\frac{h}{2t_i}\right)\right]\right. \\ \left. + \frac{d}{d\tau}(f(\tau_{N+1}))\exp\left(-\frac{h}{2t_i}\right)\left[\varphi\left(\frac{h}{2t_i}\right)-1\right]\right\} \end{aligned} \tag{14}$$

$$F_{i-re}(t) = \exp\left(-\frac{h}{2t_i}\right)F_i(t-\Delta t) \tag{15}$$

In (13) and (14), $\varphi(z) = \frac{e^z-1}{z} = 1 + \frac{1}{2}z + \frac{1}{6}z^2 + O(z^3)$. According to [43], the use of the recursive exponential functions for the tail kernel makes the calculation of the Basset integral very efficient, reducing the computational costs by more than an order of magnitude, while the memory requirements are reduced even further. (Naturally, this is more computationally expensive than using a value of zero for the tail kernel).

Daitche's methodology [12] does not include a window-based method, although its implementation is straightforward. No comparisons or explicit statements on computational time and memory savings are mentioned on [12], since all the efforts are focused in obtaining highly accurate solutions (see next section).

A comparison of the computational times of methods proposed by [4, 12, 46] with second and third order approximations, is shown in Fig. 3. The different algorithms were coded in the FORTRAN language and compared over CPU time in seconds. An arbitrary integral with an known solution was selected, namely $\frac{d}{d\tau}(f(\tau)) = \cos(\tau)$, with the known analytical solution $\int_o^t \frac{\cos(\tau)}{(\sqrt{t-\tau})}d\tau = -\sqrt{2\pi}\{\cos(t)C[\sqrt{2/\pi}\sqrt{t-\tau}] + \sin(t)S[\sqrt{2/\pi}\sqrt{t-\tau}]\}_0^t$, where the Fresnel Integrals $C$ and $S$ are given by $C(\tau) = \int_o^\tau \cos[(\pi z^2)/2]dz$ and $S(\tau) = \int_o^\tau \sin[(\pi z^2)/2]dz$ [4]. The integral was evaluated at $t = 50\pi$. All methods were used to calculate the integral over the complete time domain (no window-based method was applied). Each computation was run 1000 times, and then the average results for CPU time, rate of convergence and accuracy of each method were obtained. Figure 3 shows that the method proposed by [46] is the fastest, about one half of the time that takes the Bombardelli et al.'s approach, and one and two orders of magnitude faster than Daitche 2nd and 3rd order, respectively. This difference may be of importance when simulating thousands (or even a larger number) of particles. However, the inclusion of tail kernels may render some of these comparisons moot, as discussed below.

### 2.2.3 Rate of convergence and accuracy

In the computation of the different terms present in the MRG Eq. (1), or those terms modified or added for higher Reynolds numbers, numerical techniques such as the Runge–

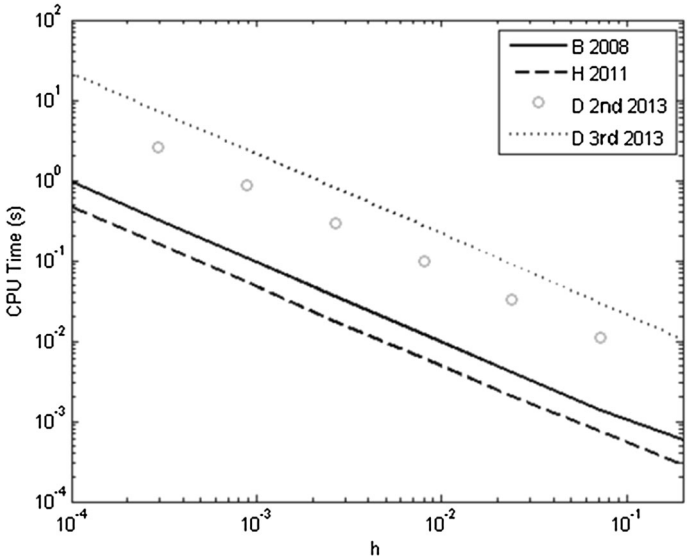


**Fig. 3** Comparison of computational time elapsed as a function of time step (**h**). The time elapsed was measured in seconds as CPU time (built in function in FORTRAN) for four different methods: Bombardelli et al. [4] (B 2008), van Hinsberg et al. [46] (H 2011), Daitche 2nd order error [12] (D 2nd 2013) and 3rd order error [12] (D 3rd 2013)

Kutta methods yield high-order solutions, and therefore it would be desirable from a purely numerical point of view that the accuracy of the computation of the Basset integral gives solutions with at least second order error [46]. The semi-derivative method proposed by [4] yields solutions with temporal accuracy $O(h)$, where $h$ is the time step. Van Hinsberg et al. [46] compared the convergence and relative error of their approach to the semi-derivative approach used by [4]. In that, a test function providing a known analytical solution for the Basset integral was used, showing that van Hinsberg et al. [46] approach produces a smaller error than the semi-derivative approach. Furthermore, when increasing the number of points ($N$) the error becomes $O(h^2)$, which is an improvement from the semi-derivative method, as expected. Daitche [12] presented a procedure to generate numerical schemes of arbitrary order [errors of $O(h^2)$ and higher] to compute the Basset integro-differential equation using quadratures.

Figure 4 depicts a comparison of order of accuracy of the four analyzed methodologies, using the same test integral from Sect. 2.2.2. It seems important to mention that the coefficients proposed by Daitche [12] for both 2nd and 3rd order approaches should be computed using quadruple precision (otherwise the 2nd and 3rd order are not achieved) and stored in double precision (to save computational time and memory).

With the information gathered from Figs. 3 and 4, it is possible to generate a plot that depicts the CPU time needed to compute the Basset term for a desired accuracy, depending on the methodology selected. This is illustrated in Fig. 5. It can be clearly seen that the method proposed by [46] is convenient to use if a second order error is sought for.

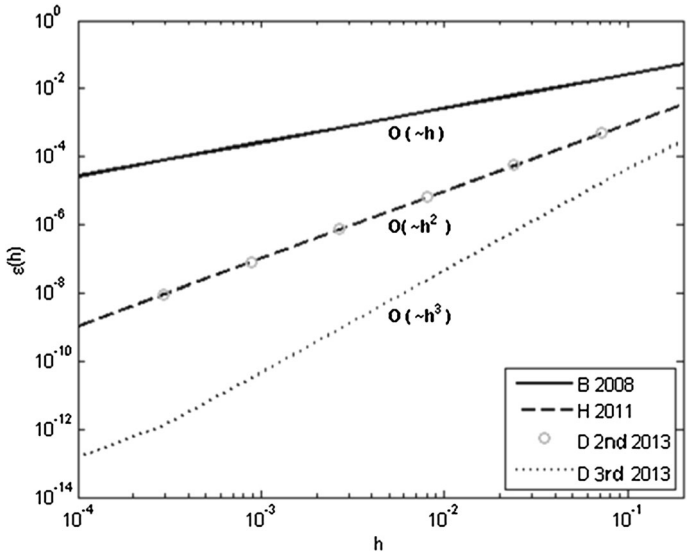


**Fig. 4** Comparison of the order of accuracy for four different methods: Bombardelli et al. [4] (B 2008), van Hinsberg et al. [46] (H 2011), Daitche 2nd order error [12] (D 2nd 2013) and 3rd order error [12] (D 3rd 2013). h indicates the time step

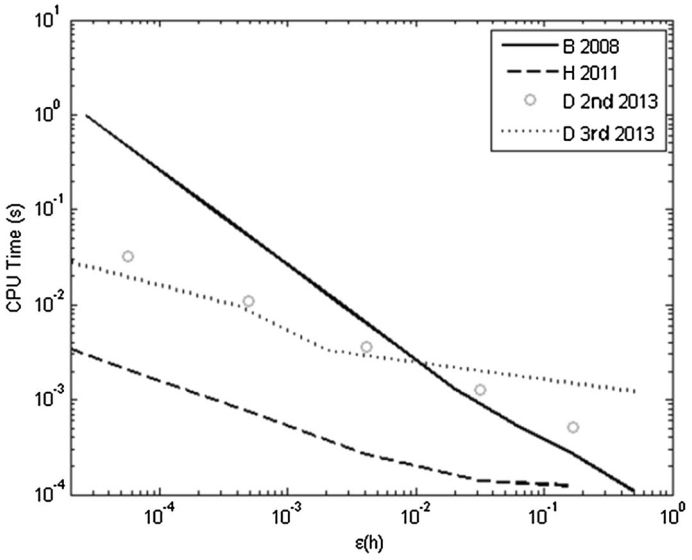


**Fig. 5** Accuracy versus CPU time of computation for four different methods: Bombardelli et al. [4] (B 2008), van Hinsberg et al. [46] (H 2011), Daitche 2nd order error [12] (D 2nd 2013) and 3rd order error [12] (D 3rd 2013)

## 3 Discussion: implications for the simulation of sediment transport and other environmental flows at the mesoscale

The methods compared in this paper offer a clear menu of techniques to compute the Basset force in many applications (in general) and in environmental flows (in particular). There are nonetheless certain restrictions in those computations, as follows. First, it is not clear whether the discussed kernels are valid in all applications, leaving it as an open question in each problem. Therefore, it should be demonstrated in each case that the kernel used represents the experimental data, to discern between having an approximate solution or a completely erroneous one, and thus reduce the uncertainty in the computations. This implies that, although some very accurate techniques have been compared herein, often the lack of knowledge about the nature of the kernel makes the very high accuracy unnecessary; it has been shown herein that in general more accurate methods require more computational time for some ranges of errors. Second, the integration of different functions for the tail kernel can allow further reductions of computational time as compared with the window kernel, so the time comparisons presented above become affected by this new computation.

Regarding the sediment-transport problem, the Basset force has been historically neglected from the analyses [21, 42, 49]. However, it has been recently found that the Basset force becomes important when the particles are relatively small [4, 35] as follows. In Figs. 6 and 7 herein, we compare numerical results obtained with our code [4, 17, 18] against data collected by [32, 34], for the cases of gravels and sands moving in saltation, respectively. Results indicate that whereas the Basset force is negligible for the range of sizes of *gravels* moving as bed-load [33], the force becomes very important for *sands* [35]. In the latter case, the length of a single particle jump can be under-predicted by about 40 % if the force is omitted, and the jump height can be under-predicted by about 15 % [4]. In agreement with our results, Mordant and Pinton [30] had noticed that the Basset force needs to be simulated in order to correctly describe the particle acceleration when the particle Reynolds number is smaller than 4000. These findings are also in accord with those of Armenio and Fiorotto [2] mentioned above.

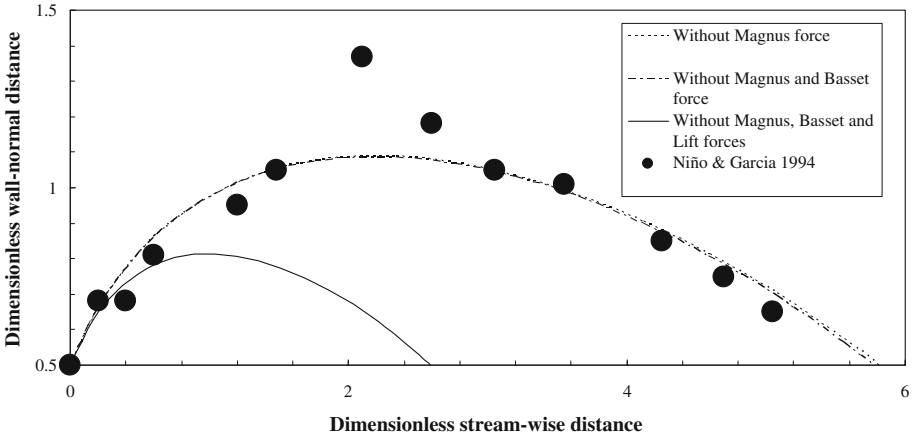


**Fig. 6** Comparison of the numerical simulation of a sediment particle in saltation mode (in 2D) with experimental data obtained by [32] for gravels (single jump). Distances are made non-dimensional by using the particle diameter. Single jump case. $d_p = 30$ mm; $u_* = 0.22$ m/s

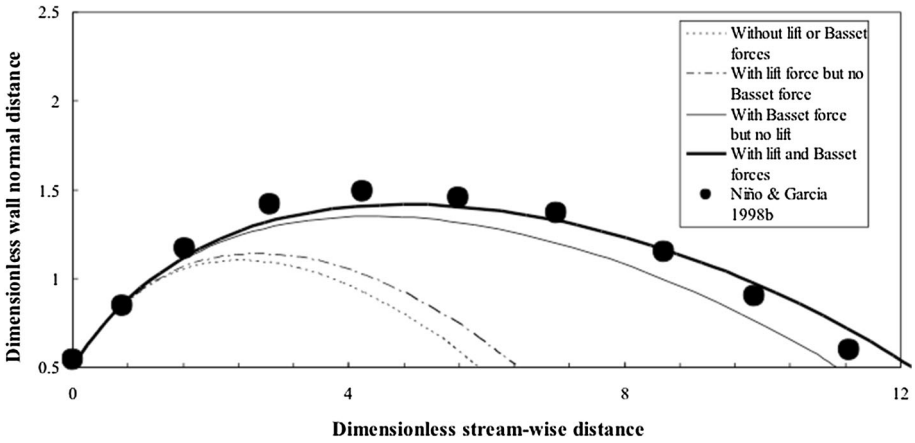


**Fig. 7** Comparison of the numerical simulation of a sediment particle in saltation mode (in 2D) with experimental data obtained by [34] for sands (single jump). Distances are made non-dimensional by using the particle diameter. Single jump case. $d_p = 0.56$ mm; $u_* = 0.025$ m/s

By looking at Fig. 7, it is immediate to conclude that the cumulative error when simulating thousands or millions of particle jumps at the mesoscale could be enormous. This is important when parameters developed from micro scales studies with expressions based on the MRG Equation (Eq. (1)) are used to mimic mesoscale processes.

Take for instance the simulation of solid particle motion to study the formation, movement, and modification of bed-forms. Consider the case of the simulation of bed-load motion of sands in a river reach comprised by several meters. Assume particles have 1 mm in diameter. According to Fig. 7, the average jump could be taken as approximately 10 times the diameter, or 1 cm in this case. For 10,000 jumps, whereas the particle should have achieved 100 m (approximately), the error of not considering the Basset force would be of approximately 50 m (following Fig. 7)!! This has direct implications on the prediction of the mass flow rates of sediment as bed-load, which are crucial in establishing patterns of deposition in harbors for instance. (This has no importance in gravels, as shown in Fig. 6). Errors appearing by neglecting the Basset force can have influence on the transverse direction as well. Non-Fickian (i.e., which do not follow Fick's law) diffusive processes in the transverse direction occur as a result of the combination of particle collisions with the wall and of inter-particle collisions [31]. Erroneous calculations of the particle position due to not incorporating the Basset force affect the collisions and, therefore, the transverse diffusion of sands.

If we also follow Fig. 3, the computation of 10,000 jumps with the second-order Daitche's method [12] will take 8 times as much effort as Bombardelli et al.'s first-order method, [4] and about 16 times as much effort as van Hinsberg et al.'s second-order method [46]. Usually, *hundreds of thousands of particles* need to be simulated to address natural flows. Therefore, if the number of jumps is multiplied by the number of particles, we can obtain an idea of the computational effort needed with all methods and the cumulative error associated with neglecting the Basset force. (Naturally, a straightforward calculation depends on many factors, among which we can mention the compiler and computer used, and whether parallel computing has been employed).

This analysis developed regarding solid particles applies to bubbly flows as well, making the computation of bubble motion feasible under certain conditions. Several Eulerian–Lagrangian models are currently applied to study the problem of bubble plumes and columns (see, for instance, [3, 44]), in which the disperse phase (i.e., the bubbles) are followed with an equation similar to (1), whereas the water flow is modeled as a continuum. In these studies, the scale of the flow is of the order of 1 or 2 m at most, with smaller distances in the transverse and span-wise directions. The Basset force needs to be considered under certain circumstances (see [41]). We believe that the findings of this paper will allow researchers to undertake such computations with more accuracy when the Basset force needs to be included.

## 4 Summary and conclusions

In this paper, we provided an overview of, and compared four *recent* contributions in the computation of the Basset force, out of several contributions in the last two decades. The focus was given to the papers [4, 12, 13, 46]. *For the first time we presented what we believe is a unified analysis of all relevant aspects of the problem*. All methods discussed efficiently overcome the singularity of the term.

For the accuracy in the computation of the standard term, the intrinsic difference between all three methods is the order of accuracy of each approximation. Whereas Bombardelli et al.'s approach [4] leads to a first-order solution in the computation, the method by van Hinsberg et al. [46] leads to a second-order solution, and the Daitche approach [12] sets grounds to obtain higher order solutions (second-order and higher). The selection of the level of accuracy required for the computation of the Basset force should depend on the integration accuracy of the other terms, but this paper clearly offers a menu to decide upon. Usually, the fourth-order Runge–Kutta method is used to integrate the other forces in Eq. (1), so higher-order solutions would be desirable, to have a consistent, global high-accurate numerical solution. However, the frequent lack of exact knowledge of the validity of the Basset force kernel to use can often make this high accuracy "illusory".

Overall, all methods give better time accuracy than known quadratures ($O(h^{1/2})$), and in that regard they are all efficient contributions in their own ways. The second-order method by van Hinsberg et al. [46] seems to provide in principle a good balance between accuracy and computational cost. The issue is that in spite of the approach and the accuracy of the numerical scheme employed, the reduction in computational time coming from the tail term can have a large importance in comparison with the window term. Therefore, savings in this aspect will highly depend on the application and the use of a tail kernel.

We believe that the presentation on the subject followed in this paper offers a good framework for the understanding of past and future developments on the Basset force.

## References

1. Amador A, Sanchez-Juny M, Dolz J (2006) Characterization of the nonaerated flow region in a stepped spillway by PIV. J Fluids Eng 128(6):1266–1273
2. Armenio V, Fiorotto V (2001) The importance of the forces acting on particles in turbulent flows. Phys Fluids 13(8):2437–2440
3. Ali BA, Pushpavanam S (2011) Analysis of unsteady gas-liquid flows in a rectangular tank: comparison of Euler-Eulerian and Euler-Lagrangian simulations. Int J Multiph Flow 37:268–277

4. Bombardelli FA, González AE, Niño YI (2008) Computation of the particle Basset force with a fractional-derivative approach. J Hydraul Eng ASCE 134(10):1513–1520
5. Bombardelli FA, Jha S (2009) Hierarchical modeling of the dilute transport of suspended sediment in open channels. Environ Fluid Mech 9(2):207–235. doi:10.1007/s10652-008-9091-6
6. Bombardelli FA, Chanson H (2009) Progress in the observation and modeling of turbulent multi-phase flows. Environ Fluid Mech 9(2):121–123
7. Bombardelli FA, Meireles I, Matos J (2011) Laboratory measurements and multi-block numerical simulations of the mean flow and turbulence in the non-aerated skimming flow region of steep stepped spillways. Environ Fluid Mech 11:263–288
8. Bombardelli FA, Moreno PA (2012) Exchange at the bed sediments-water column interface. In: Gualtieri C, Mihailovic DT (eds) Fluid mechanics of environmental interfaces, Chapter 8, 2nd edn. Taylor & Francis Group, London, pp 221–253
9. Brunner H, Tang T (1989) Polynomial spline collocation methods for the nonlinear Basset equation. Comput Math Appl 18(5):449–457
10. Brush JS, Ho HW, Yen BC (1964) Acceleration motion of sphere in a viscous fluid. J Hydraul Div 90:149–160
11. Crowe C, Sommerfeld M, Tsuji M (2011) Multiphase flows with droplets and particles. CRC Press, Boca Raton
12. Daitche A (2013) Advection of inertial particles in the presence of the history force: higher order numerical schemes. J Comput Phys 254:93–106
13. Dorgan A, Loth E (2007) Efficient calculation of the history force at finite Reynolds numbers. Int J Multiph Flow 33:833–848
14. Drew DA, Passman SL (1999) Theory of multicomponent fluids. Springer, Berlin
15. García MH (2008) Sedimentation engineering: processes, measurements, modeling, and practice. ASCE Manuals and Reports on Engineering Practice N° 110, Reston
16. Gatignol R (1983) The Faxén formulae for a rigid particle in an unsteady non-uniform Stokes flow. J Méc Théor 1:143–160
17. González AE, Bombardelli FA, Nino YI (2006) Improving the prediction capability of numerical models for particle motion in water bodies. In: Proceedings of the 7th International Conference on HydroScience and Engineering ICHE 2006, Philadelphia, USA
18. González AE (2008) Coupled numerical modeling of sediment transport near the bed using a two-phase flow approach. Ph.D. Thesis, University of California, Davis, p 188
19. Kim I, Elghobashi S, Sirignana WB (1998) On the equation for spherical-particle motion: effect of Reynolds and acceleration numbers. J Fluid Mech 367:221–253
20. Lawrence CJ, Mei R (1995) Long-time behavior of the drag on a body in impulsive motion. J Fluid Mech 283:307–327
21. Lee HY, Hsu IS (1994) Investigation of saltating particle motions. J Hydraul Eng ASCE 120:831–845
22. Loth E, Dorgan A (2009) An equation of motion for particles of finite Reynolds number and size. Environ Fluid Mech 9(2):187–206
23. Lovalenti PM, Brady JF (1993) The hydrodynamic force on a rigid particle undergoing arbitrary time-dependent motion at small Reynolds number. J Fluid Mech 256:561–605
24. Lukerchenko N, Chara Z, Vlasak P (2006) 2D Numerical model of particle-bed collision in fluid-particle flows over bed. J Hydraul Res IAHR 44(1):70–78
25. Lukerchenko N, Piatsevich S, Chara Z, Vlasak P (2009) 3D numerical model of spherical particle saltation in a channel with a rough fixed bed. J Hydrol Hydromech 57(2):100–112
26. Maxey MR, Riley JJ (1983) Equation of motion for a small rigid sphere in a nonuniform flow. Phys Fluids 26:883–889
27. Mei R, Adrian RJ (1992) Flow past a sphere with an oscillation in the free-stream velocity and unsteady drag at finite Reynolds number. J Fluid Mech 237:323–341
28. Michaelides EE (2006) Particles, bubbles and drops: their motion, heat and mass transfer. World Scientific Publishing Co, Hackensack
29. Moorman RW (1955) Motion of a spherical particle in the accelerated portion of free-fall. Doctor of Philosophy Dissertation, University of Iowa
30. Mordant N, Pintot JF (2000) Velocity measurement of a settling sphere. Eur Phys J B 18:343–353
31. Moreno PA, Bombardelli FA (2012) 3D numerical simulation of particle-particle collisions in saltation mode near stream beds. Acta Geophys 60:1661–1688
32. Niño Y, García M, Ayala L (1994) Gravel saltation. Experiments. Water Resour Res 30(6):1907–1914
33. Niño Y, García M (1994) Gravel saltation. 2. Modeling. Water Resour Res 30(6):1915–1924
34. Niño Y, García M (1998) Experiments on saltation of sand in water. J Hydraul Eng 124(10):1014–1025

35. Niño Y, García M (1998) Using Lagrangian particle saltation observations for bed-load sediment transport modelling. Hydrol Process 12:1197–1218
36. OldhamY Spanier M (1974) The fractional calculus. Academic, New York
37. Parmar M, Haselbacher A, Balachandar S (2011) Generalized Basset-Boussinesq-Oseen equation for unsteady forces on a sphere in a compressible flow. Phys Rev Lett 106:084501
38. Press WH, Teukolsky SA, Vetterling WT, Flannery BP (1992) Numerical recipes. Cambridge University Press, New York
39. Prosperetti A, Tryggvason G (2007) Computational methods for multiphase flow. Cambridge University Press, Cambridge
40. Rechiman LM, Dellavalle D, Bonetto FJ (2013) Path suppression of strongly collapsing bubbles at finite and low Reynolds numbers. Phys Rev E 87:063004
41. Sangani AS, Zhang DZ, Prosperetti A (1991) The added mass, Basset, and viscous drag coefficients in nondilute bubbly liquids undergoing small-amplitude oscillatory motion. Phys Fluids 3(12):2955–2970
42. Schmeeckle MW, Nelson JM (2003) Direct numerical simulation of bed-load transport using a local, dynamic boundary condition. Sedimentology 50:279–301
43. Sridhar G, Katz J (1995) Drag and lift forces on microscopic bubbles entrained by a vortex. Phys Fluids 7(2):389–399
44. Sungkorn R, Derksen JJ, Khinast JG (2012) Euler-Lagrange modeling of a gas-liquid stirred reactor with consideration of bubble breakage and coalescence. AIChE J 58(5):1356–1370
45. Tatom FB (1988) The Basset term as a semiderivative. Appl Sci Res 45:283–285
46. van Hinsberg MAT, Boonkkamp JHMT, Clercx HJH (2011) An efficient, second order method for the approximation of the Basset history force. J Comput Phys 230(4):1465–1478
47. Wakaba L, Balachandar S (2005) History force on a sphere in a weak shear flow. Int J Multiph Flow 31:996–1014
48. Wiberg P, Smith JD (1985) A theoretical model for saltating grains in water. J Geophys Res 90:7341–7354
49. Wood IR, Jenkins BS (1973) A numerical study of the suspension of a non-buoyant particle in a turbulent stream. In: Proceedings of IAHR internatinal symposium on river mechanics, vol 1, pp 431–442
50. Yen BC (1992) Sediment fall velocity in oscillating flow. Water resources environmental engineering resources. Report 11. Dept. of Civil Eng, University of Virginia