Journal of Computational Physics 229 (2010) 5862–5869

Contents lists available at ScienceDirect

Journal of Computational Physics

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

Implementation of high order spline interpolations for tracking test particles in discretized fields

C.C. Lalescu [a,b,*], B. Teaca [a,b], D. Carati [a]

[a] Statistical and Plasma Physics, Université Libre de Bruxelles, Campus Plaine, CP 231, B-1050 Brussels, Belgium
[b] Faculty of Physics, University of Craiova, A.I. Cuza 13, 200585 Craiova, Romania

ARTICLE INFO

Article history:
Received 21 May 2009
Received in revised form 27 October 2009
Accepted 28 October 2009
Available online 1 November 2009

Keywords:
Spline interpolation
Test particle
Numerical optimization

ABSTRACT

A systematic approach for constructing high order spline interpolation methods is proposed for fields known on regular, rectangular grids. These interpolation methods are tested in tracking trajectories of particles submitted to a force that derives from a potential known on a grid. The interplay between the time advancement scheme and the spatial interpolation is studied in detail and it is shown how the order of the trajectory solver is directly affected by the order of the spline interpolation. It is also shown how an interpolation method that preserves topological properties of physical fields can be better exploited with these higher order spline approximations.

1. Introduction

The correct description of particle transport is important in a huge number of problems encountered by physicists [1–3], including geophysicists [4] and astrophysicists [5], chemists and engineers. Particle transport may be affected by interactions between the particles themselves as well as by interactions with their environments. In the present study, only the case in which transport can be considered as fully determined by the influence of the environment, and the interactions between the particles are neglected, is considered. The approximation of non-interacting particles has an extremely wide range of applications, such as the dispersion of pollutants in the atmosphere or in rivers and oceans, the acceleration of charged particles in complex electromagnetic fields in astrophysical systems [5–7] as well as the estimation of particle loss in magnetic confinement devices. Various mathematical approaches can be considered to describe these phenomena [1]. Hydrodynamic balance equations can be solved to describe the influence of the velocity field on a concentration of passive particles. Kinetic approaches for the evolution of the particle velocity distribution function using Fokker–Planck type of equations can also be considered. However, in certain cases, it is interesting to consider the direct description of an ensemble of particle trajectories. The mathematical framework is then quite easy to establish and, considering only classical mechanics (i.e. disregarding quantum and relativistic effects), the evolution of the particles is simply described by Newton's second law [3,4]. The forces acting on the particles have of course to be known if the problem has to be solved either analytically or numerically. In practice, these forces are either known exactly and an analytical expression can be provided, or they are known in a statistical sense and stochastic processes have to be introduced such as in the Brownian motion. In certain cases however, the external forces are known from numerical simulations [6–10]. For instance, test particles in a fluid could experience a

* Corresponding author. Address: Statistical and Plasma Physics, Université Libre de Bruxelles, Campus Plaine, CP 231, B-1050 Brussels, Belgium. Tel.: +32 (0)2 650 5955; fax: +32 (0)2 650 5028.
E-mail addresses: clalescu@ulb.ac.be (C.C. Lalescu), bteaca@ulb.ac.be (B. Teaca), dcarati@ulb.ac.be (D. Carati).

0021-9991/$ - see front matter © 2009 Elsevier Inc. All rights reserved.
doi:10.1016/j.jcp.2009.10.046

friction velocity that is proportional to the difference between the particle and the fluid velocities. If the fluid velocity is described by a Navier–Stokes solver, the friction force is derived from numerical simulations. Also, if a charged particle is interacting with a complex electromagnetic field, both the electric and the magnetic fields entering the Lorentz force are sometimes obtained from numerical simulations of the Maxwell equations. In these systems, it is required to interpolate the forces at points in space along the particle's trajectory, that do not necessarily fall on the grid nodes used to compute the forces. The main purpose of the present study is to evaluate the coupling between the various time integration schemes used to follow the particles and the space interpolation schemes used to estimate the forces acting on these particles.

An important aspect is of course the order of the numerical time integration scheme. It should be noted however that additional constraints might have to be considered. For instance, if the particle is submitted to a force that derives from an energy potential, total energy conservation might be a more important property to be reproduced by the time integration scheme. Also, if the force contains a velocity field describing an incompressible flow or a magnetic field, it might be crucial that the interpolation scheme preserves the divergence-free condition of the field [4].

The discussion presented here is limited to force fields that are represented on a uniform, rectangular spatial grid, particularly relevant to the case of fluid dynamic or magneto-hydrodynamic pseudo spectral simulations as in [6,10,11]. The study is focused on the construction and the implementation of a hierarchy of spline interpolations. These spline interpolation methods are applied to a simple Hamiltonian system, for which the exact expression of the force is also available. The method is however applicable to any numerical simulations of physical or virtual particles in a discretized field, including first order ordinary differential equations such as those used to describe field lines instead of particle trajectories.

2. Theoretical framework

2.1. Physical problem

As a case study, the motion of a unit-mass particle is considered in a given potential V. The Hamiltonian describing this system has the simple form

$$H(\mathbf{p},\mathbf{r}) = \frac{1}{2}\mathbf{p}^2 + V(\mathbf{r}), \tag{1}$$

where $\mathbf{p}$ is the particle's momentum and $\mathbf{r}$ its position. This system evolves according to Hamilton's equations:

$$\frac{d\mathbf{r}}{dt} = \mathbf{p}, \quad \frac{d\mathbf{p}}{dt} = \mathbf{F}(\mathbf{r}), \tag{2}$$

where the force is given by $\mathbf{F} = -\nabla V(\mathbf{r})$. Since V depends explicitly only on $\mathbf{r}$, the Hamiltonian (1) is separable. The separability allows for a relatively easy way of constructing explicit, high order, symplectic solvers [12–14]. In particular, two classes of time integration schemes for solving the Hamilton equations will be considered: symplectic composition methods (CM), and the implicit Gauss–Runge–Kutta (GRK) schemes that are actually symplectic for more general non-separable systems.

From the evolution equations (2), it is obvious that the numerical simulation of the particle trajectory requires the knowledge of the force field $\mathbf{F}$ at each point $\mathbf{r}$. Typically, the force field, or any other field of interest for that matter, is known only on the nodes of a grid. The field values at the point $\mathbf{r}$ are approximated by an interpolation method and labelled $\widehat{\mathbf{F}}$. The influence of the interpolation method on the accuracy of the time integration scheme is important. For instance, considering a first order ordinary differential equation $dx/dt = f(x)$ and the Taylor expansion of its solution

$$x(t) = x(0) + tf(x)|_{x(0)} + \frac{t^2}{2}f'(x)|_{x(0)} + \cdots + \frac{t^m}{n!}f^{(m-1)}(x)|_{x(0)} + \cdots, \tag{3}$$

the expansion for any mth order solver should coincide with this Taylor expansion up to the mth order. However, if the order m derivative of the force field is ill-defined, any solver will have at most a global order m. Although this property is quite obvious, it has been often overlooked when numerical integration schemes are coupled with interpolation methods. Indeed, interpolation methods often produce force field $\widehat{\mathbf{F}}$ that are not infinitely differentiable even if the original field itself was infinitely differentiable. For instance, in the following subsection, spline approximations are considered with continuous mth order derivatives but for which the derivatives of order $m+2$ are ill-defined. Specifically, the $m+1$ order derivatives are continuous almost everywhere and bounded, whereas the $m+2$ order derivatives are not bounded at their discontinuities.

2.2. Spline approximations

In order to simplify the notations, the spline interpolation is first discussed for a scalar field $f(x)$ in a one-dimensional space. The space dimension is actually not important in this construction. In most practical cases, the field will be obtained from three-dimensional fluid, magneto-hydrodynamic or Maxwell equations solver. In that case, the information related to the force $\mathbf{F}$ is assumed to be obtained from a simulation on a rectangular, three-dimensional, uniform grid [6,8,9]. Consid-

ering the one-dimensional problem, the values of $f(x)$ are assumed to be known on a grid, with the constant distance Δ between the nodes. The interpolation scheme aims to provide an approximation of $f(x)$ for a given x that belongs to an interval $[x_\ell, x_{\ell+1}]$. Without loss of generality, the variable x can be translated by $-x_\ell$ and expressed in unit of Δ, so that we can assume that x lies in the interval $[0,1]$. The spline is given by the nth order polynomial:

$$s^{(n)}(x) = \sum_{k=0}^{n} a_k^{(n)} x^k. \tag{4}$$

For a Hermite spline interpolation, the values of $s^{(n)}$ and of its derivatives up the order $m \equiv (n-1)/2$ must coincide with those of the original function on the nodes $x = 0$ and $x = 1$:

$$\begin{cases} \frac{d^l s^{(n)}}{dx^l}(0) = \frac{d^l f}{dx^l}(0), \\ \frac{d^l s^{(n)}}{dx^l}(1) = \frac{d^l f}{dx^l}(1), \end{cases} \quad l = 0, \ldots, m. \tag{5}$$

By solving the system of Eq. (5) we can find the coefficients $a_k^{(n)}$ depending on the values on the nodes of the function and its derivatives: $f(0), f(1), f'(0), f'(1), \ldots$:

$$a_k^{(n)} = \sum_{l=0}^{m} \sum_{i=0}^{1} c_{kli}^{(n)} f^{(l)}(i), \tag{6}$$

where $c_{kli}^{(n)}$ are numerical coefficients fixed for a given n. This approach is used in [15], where exact derivatives of the field are used to construct a cubic spline interpolation. Computing the coefficients $a_k^{(n)}$ is mostly useful for static fields. For dynamic fields or large n, this approach is computationally prohibitive, so an alternative method is considered here. After introducing the solution (6) into (4) and expressing the derivatives using centred differences, the spline can be rewritten as:

$$s^{(n,q)}(x) = \sum_{i=-g}^{g+1} f(i) \beta_i^{(n,q)}(x), \tag{7}$$

where $q = 2g + 2$ is the number of grid nodes needed to construct the approximation and each $\beta_i^{(n,q)}(x)$ is generally a polynomial of degree n in x. The formula (7) is valid for an $x \in [0,1]$, and it is basically a weighted sum of the values of f on the neighbour grid nodes $-g, -g+1, \ldots, 0, 1, 2, \ldots, g-1, g, g+1$. The same formula is applied for other intervals, by shifting the nodes accordingly along the grid. At the grid boundaries, this formula needs to be adapted. In the tests reported hereafter, periodic boundary conditions were used and the formula is then easily adapted by using $f(i \pm N) = f(i)$ if N is the number of grid points. However, for other types of boundary conditions, adapting the formula (7) may very well become problem dependent. For $q = 4$, the following centred differences expression for the first and second order derivatives can be used (in order to emphasise the order of the expansion in terms of Δ, un-normalised units for the position x are restored *only* in the following well-known centred difference formulae (9)–(13)):

$$f'(x) = \frac{f(x+\Delta) - f(x-\Delta)}{2\Delta} + O(\Delta^2), \tag{8}$$

$$f''(x) = \frac{f(x+\Delta) - 2f(x) + f(x-\Delta)}{\Delta^2} + O(\Delta^2). \tag{9}$$

For $q = 6$, fourth order derivatives can be computed at most:

$$f'(x) = \frac{-f(x+2\Delta) + 8f(x+\Delta) - 8f(x-\Delta) + f(x-2\Delta)}{12\Delta} + O(\Delta^4), \tag{10}$$

$$f''(x) = \frac{-f(x+2\Delta) + 16f(x+\Delta) - 30f(x) + 16f(x-\Delta) - f(x-2\Delta)}{12\Delta^2} + O(\Delta^4), \tag{11}$$

$$f^{(3)}(x) = \frac{f(x+2\Delta) - 2f(x+\Delta) + 2f(x-\Delta) - f(x-2\Delta)}{2\Delta^3} + O(\Delta^2), \tag{12}$$

$$f^{(4)}(x) = \frac{f(x+2\Delta) - 4f(x+\Delta) + 6f(x) - 4f(x-\Delta) + f(x-2\Delta)}{\Delta^4} + O(\Delta^2). \tag{13}$$

The computation of the β polynomials demands some care but it is straightforward. The $\beta_i^{(5,4)}$s are given as an example:

$$\beta_{-1}^{(5,4)}(x) = \frac{1}{2}(x-1)^3 x(2x+1), \tag{14}$$

$$\beta_0^{(5,4)}(x) = -\frac{1}{2}(x-1)(6x^4 - 9x^3 + 2x + 2), \tag{15}$$

$$\beta_1^{(5,4)}(x) = \frac{1}{2}x(6x^4 - 15x^3 + 9x^2 + x + 1), \tag{16}$$

$$\beta_2^{(5,4)}(x) = -\frac{1}{2}(x-1)x^3(2x-3). \tag{17}$$

A simple D-dimensional generalization of the spline can easily be constructed:

$$s^{(n,q)}(x_1, x_2, \ldots, x_D) = \sum_{i_1,\ldots,i_D=-g}^{g+1} \left(f(i_1, \ldots, i_D) \prod_{j=1}^{D} \beta_{i_j}^{(n,q)}(x_j) \right). \tag{18}$$

It can be shown that the cross-derivatives for this field are automatically continuous:

$$\left(\prod_{j=1}^{D} \left(\frac{\partial}{\partial x_j} \right)^{l_j} \right) s^{(n,q)} \in \mathcal{C}^{m-\max\{l_j\}}, \tag{19}$$

whenever $m \geqslant \max\{l_j\}$. Here, $\mathcal{C}^\ell$ represents the set of functions that are continuous and have continuous derivatives up to the ℓth order. The formula (18) allows for a relatively easy parallelization of the computation. In practice, if the field is split between several processors along one of the coordinates, the formula can also be split along that coordinate.

To summarise, considering a field f known only on a regular grid, the Hermite approach discussed in this section allows to construct a spline approximation, based on polynomials of order n, that uses q grid points in each direction. The formal operator that transforms a field f into its spline approximation $\hat{f}$ will be named (for later reference) $S^{(n,q)}$:

$$\hat{f} = S^{(n,q)} f. \tag{20}$$

2.3. Force representation

In the test case considered here (1) and (2), the force has been deliberately assumed to derive from a potential energy. This property is actually not required by the time integration schemes nor by the space interpolation methods we are considering. However, it allows the investigation of two very important features of various physical systems. First, it gives a direct access to the total energy of the particle. Hence, the numerical schemes can be tested not only in terms of the accuracy of the solution but also in terms of their ability to conserve the total energy. Second, and this is closely related to the previous point, the force is curl-free:

$$\nabla \times \mathbf{F}(\mathbf{r}) = -\nabla \times (\nabla V(\mathbf{r})) = 0. \tag{21}$$

The numerical schemes, and here this concerns only the space interpolation methods, can also be tested in terms of their ability to preserve this property. For instance, the operator $S^{(n,q)}$ can be applied directly to the potential and then the force can be derived from the interpolated potential:

$$\tilde{\mathbf{F}}(\mathbf{r}) = -\nabla(\widehat{V}) = -\nabla(S^{(n,q)} V). \tag{22}$$

A similar approach was first suggested in [4] where, in order to obtain a divergence-free interpolation of the magnetic field, the vector potential is interpolated with cubic splines and its derivatives are used for the magnetic field. The expression for $\widetilde{\mathbf{F}}(\mathbf{r})$ can be obtained efficiently by using the pre-computed derivatives of the β polynomials. The advantage of this approach here is that the expression for $\widetilde{\mathbf{F}}(\mathbf{r})$ is automatically curl-free. However, the components of the forcing have a smaller number of continuous derivatives than the direct approximation:

$$\widehat{\mathbf{F}}(\mathbf{r}) = -S^{(n,q)} \mathbf{F}(\mathbf{r}) = -S^{(n,q)} \nabla(V). \tag{23}$$

The choice between smoother interpolations for the force or an exactly curl-free expression is discussed in the next section.

3. Numerical results

3.1. Test case

The test case that is presented here corresponds to a very simple expression for the potential:

$$V(x, y, z) = \cos(x)\cos(y) + \cos(y)\cos(z). \tag{24}$$

Obviously, since the analytic expression for the potential is known, the exact expression for the force is also known and the interpolation schemes can then be easily assessed. The interpolation schemes are built by assuming that the fields are represented in a box of length 2π, with periodic boundary conditions. The resolution chosen for the spatial grid is 6 nodes per direction, which is more than the minimum needed to capture the exact Fourier expansion for the fields without loss of information due to discretization.

Particles are placed at random initial positions chosen under the condition that their potential energy is less than 1/2. Then, their velocity is also chosen with a random direction but with an amplitude that yields to a total energy exactly equal to 1/2. In all the tests presented in the following, the number of particles is always $N = 1000$. All results are given in non-dimensional units. The main advantage of choosing an ensemble of particles with the same total energy is that the statistics are not dominated by just a few trajectories. Each particle can then be seen as "statistically equivalent" to the others.

Such an example is, of course, extremely simple. It has the major advantage of being characterised by very few free parameters and its simulation can be reproduced very easily. It must be stressed however that the coupling between the time integration schemes and the spline interpolation methods has been investigated for various other examples with increasing complexity. The same conclusions have been reached for all potentials and it has been decided to focus the present report on the simple example (24).

3.2. Limit order for solvers

The first important result of the numerical tests is the strong influence of the interpolation method on the global order of the solver. The order is estimated by computing the particle trajectories from $t = 0$ to $t = T$ with a series of decreasing time-steps τ_k. The final values of phase-space variables for the ith particle form the final state of the particle $\xi_i^{(k)}(T) = (\mathbf{r}_i^{(k)}(T), \mathbf{p}_i^{(k)}(T))$, which depends on the time step τ_k. Comparing these final states for different time-steps gives an estimate of the accuracy of the numerical scheme:

$$\varepsilon_i^{(k)} \equiv \|\xi_i^{(k+1)}(T) - \xi_i^{(k)}(T)\|. \tag{25}$$

For this test, a value of $T = 10$ for the total time is sufficiently long to obtain the global order for all solvers considered. In Figs. 1 and 2, the logarithm of the average error $\varepsilon^{(k)} = \langle \varepsilon_i^{(k)} \rangle$, computed over the ensemble of particles, is plotted against the logarithm of the time-step τ_k. The same 8th order composition method, by Kahan and Li [16], is used to for the time advancement scheme with various splines approximations for the force. The intervals containing 90% of the errors $\varepsilon_i^{(k)}$ are also shown in the figures.

The slopes obtained should be at most equal to 8. In the simulations reported in these figures, this upper limit is only observed for the analytical form of the force which corresponds to the $S^{(\infty,\infty)}$ spline. For the spline of order n, the global order of the numerical scheme appears to be approximately $(n-1)/2+2$ when using the $\widehat{\mathbf{F}}$ force approximation, and $(n-1)/2+1$ when using $\widetilde{\mathbf{F}}$. Hence, the order of the solver significantly decreases with decreasing order of the spline interpolation scheme. In particular, the linear interpolation, $S^{(1,2)}$, used extensively in the literature [6,8,9], yields a global solver order equal to 2.

To emphasize the influence of the spatial interpolation method on the global order of the particle trajectory solver, the average error $\varepsilon^{(k)}$ is computed for various time integration schemes. The CM4 solver by Blanes and Moan [17], the CM6 and CM8 solvers by Kahan and Li [16] and the implicit GRK4 and GRK6 solvers [12] are used in combination either with the analytical expression for the force $S^{(\infty,\infty)}$ in Fig. 3 or with the spline $S^{(7,6)}$ approximation in Fig. 4. This particular choice of spline allows for a maximal order of 5 for the solvers, when using $\widehat{\mathbf{F}}$. For this reason, the 4th order solvers recover the slope 4 while all the other solvers, of higher orders, are reduced to 5.

3.3. Energy conservation

The energy conservation can be used to introduce another measure of the solver accuracy by computing the quantity:

$$\varepsilon_e(t) = \left\langle \left| \frac{1}{2}\mathbf{p}_i^2(t) + \widehat{V}(\mathbf{r}_i(t)) - \left[\frac{1}{2}\mathbf{p}_i^2(0) + \widehat{V}(\mathbf{r}_i(0)) \right] \right| \right\rangle, \tag{26}$$

where, again, the average is performed over all the particles. For this type of diagnostics, longer runs are necessary and the particle trajectories have been followed from $t = 0$ to $t = T = 1000$ with a time-step of 2^{-3}, using the CM4 solver from Blanes

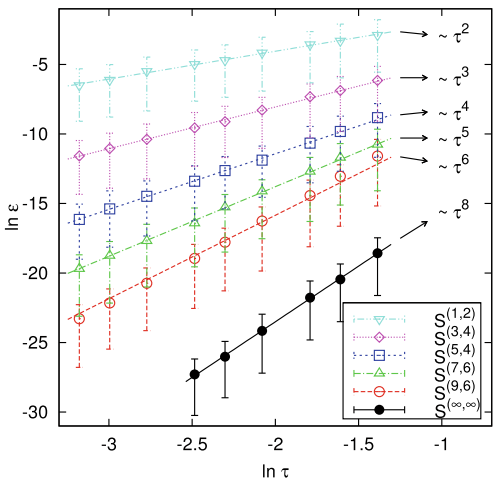

Fig. 1. Average error (ε) versus the time-step (τ), using various splines, for the $\widehat{\mathbf{F}}$ force approximation. The same 8th order composition method is used for all cases. The lines represent time-step power-laws and are shown for reference.

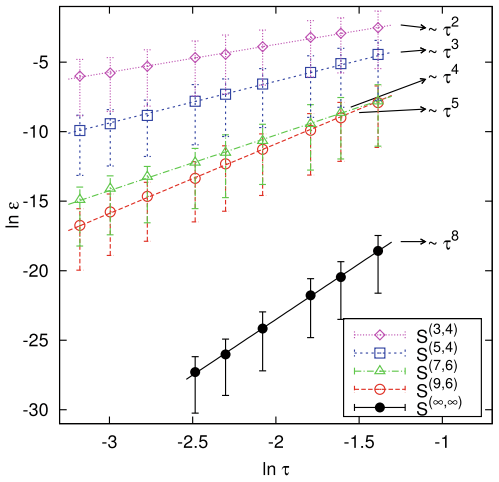

Fig. 2. Average error (ε) versus the time-step (τ), using various splines, for the $\widetilde{\mathbf{F}}$ force approximation. The same 8th order composition method is used for all cases. The lines represent time-step power-laws and are shown for reference.

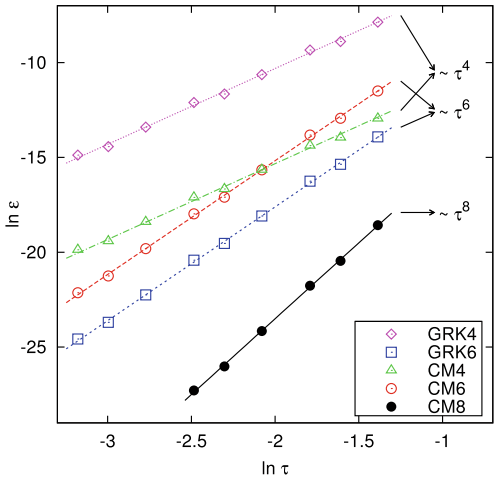

Fig. 3. Average error (ε) versus the time-step (τ), using the exact formula for the force, $S^{(\infty,\infty)}$. Results for the 4th and 6th order Gauss–Runge–Kutta (GRK) solvers and the 4th, 6th and 8th order composition methods (CM) are shown.

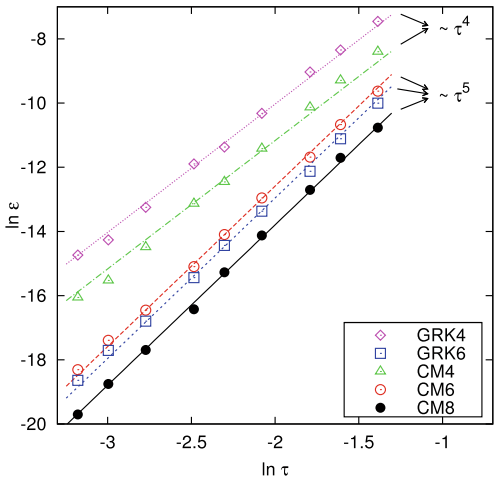

Fig. 4. Average error (ε) versus the time-step (τ), using the $S^{(7,6)}$ spline for the $\widehat{\mathbf{F}}$ force approximation. Results for the 4th and 6th order Gauss–Runge–Kutta (GRK) solvers and the 4th, 6th and 8th order composition methods (CM) are shown.

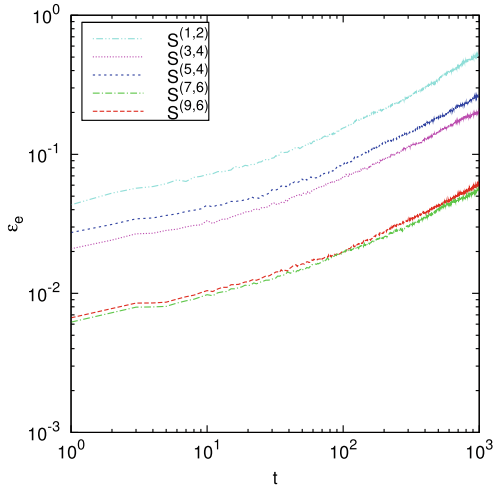

Fig. 5. Energy error ε_e versus time for various spline approximations for $\widehat{\mathbf{F}}$.

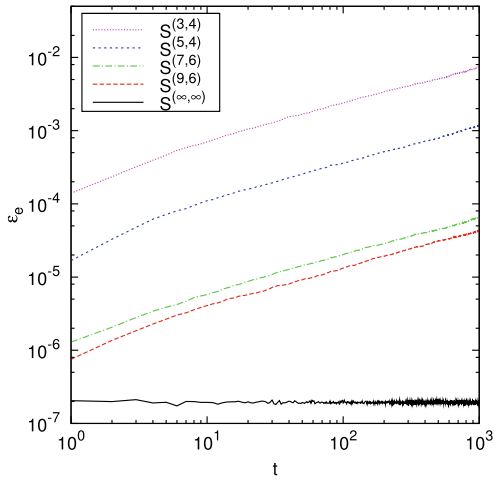

Fig. 6. Energy error ε_e versus time for various spline approximations for $\widetilde{\mathbf{F}}$.

and Moan [17]. Only the interpolation method is changed. Results are reported in Figs. 5 and 6. The energy error is constant when the exact expression for the force is used, as expected [12–14]. This indicates that the time integration scheme is not responsible for a possible lack of energy conservation. The comparison of the figures shows that energy conservation is much better when the spline approximations are used on the energy potential to produce $\widetilde{\mathbf{F}}$ than when it is applied directly to the forcing to yield $\widehat{\mathbf{F}}$. Moreover, the gain in accuracy is much more important when the spline order is increased for $\widetilde{\mathbf{F}}$ than for $\widehat{\mathbf{F}}$.

4. Conclusions

A systematic approach for constructing nth order spline interpolation methods has been proposed. These interpolation methods have been discussed for the simple case of tracking the trajectories of particles submitted to a force derived from an energy potential.

It has been shown that the effective order of the trajectory solver is directly affected by the order of the spline interpolation, as expected. However, this effect may be reduced by using finer grid resolutions for the fields. Indeed, in that case, because the discontinuity in the $m+1$ order derivative is decreased, the measure of the positions for which the $m+2$ derivative explodes and affects the order of the solver is reduced. Except in the limit of very fine grids, the order of the solver remains generally the same even if the interpolation error may be reduced in magnitude. It must also be stressed that increasing the resolution is not always an option. In particular if the fields are obtained from high resolution turbulence simulations, a further increase of the number of grid points may be impractical.

We also show how the qualitative aspects of the physics involved can be better reproduced by implementing an alternative interpolation method, suggested in [4], and using this method in conjunction with high order splines. It must also be

recognised that the iso-surfaces $\{\mathbf{r} \in \mathbb{R}^3 | V(\mathbf{r}) = a\}$ and $\{\mathbf{r} \in \mathbb{R}^3 | \widehat{V}(\mathbf{r}) = a\}$ can have different topological properties, as particles with energy a could be trapped in V but not in $\widehat{V}$ or vice versa. It is therefore important to evaluate how important these trapping effects are before choosing the interpolation method. However, since an entire hierarchy of splines interpolation methods is available, increasing the number of grid points used to reach an acceptable degree of the errors for the problem considered is always feasible.

Acknowledgments

Helpful discussions with G. Steinbrecher and I. Petrisor from the University of Craiova are gratefully acknowledged. This work has been supported by the contract of association EURATOM – Belgian state. The content of the publication is the sole responsibility of the authors and it does not necessarily represent the views of the Commission or its services. D.C. is supported by the Fonds de la Recherche Scientifique (Belgium).

References

[1] R. Balescu, Statistical Dynamics. Matter Out of Equilibrium, Imperial College Press, 1997.
[2] M. Vlad, J.-D. Reuss, F. Spineanu, J.H. Misguich, Transport scaling and trapping, J. Plasma Phys. 59 (1998) 707–718.
[3] J. Klafter, G. Zumofen, Lévy statistics in a hamiltonian system, Phys. Rev. E 49 (6) (1994) 4873–4877.
[4] F. Mackay, R. Marchand, K. Kabin, Divergence-free magnetic field interpolation and charged particle trajectory integration, J. Geophys. Res. 111 (2006) A06208.
[5] L. Vlahos, H. Isliker, F. Lepreti, Particle acceleration in an evolving network of unstable current sheets, Astrophys. J. 608 (2004) 540–553.
[6] K. Arzner, B. Knaepen, D. Carati, N. Denewet, L. Vlahos, The effect of coherent structures on stochastic acceleration in MHD turbulence, Astrophys. J. 637 (2006) 322–332.
[7] J. Cho, A. Lazarin, Particle acceleration by magnetohydrodynamic turbulence, Astrophys. J. 638 (2006) 811–826.
[8] P. Dmitruk, W.H. Matthaeus, N. Seenu, M.R. Brown, Test particle acceleration in three-dimensional magnetohydrodynamic turbulence, Astrophys. J. 597 (2003) L81–L84.
[9] P. Dmitruk, W.H. Matthaeus, N. Seenu, Test particle energization by current sheets and nonuniform fields in magnetohydrodynamic turbulence, Astrophys. J. 617 (2004) 667–679.
[10] H. Homann, J. Dreher, R. Grauer, Impact of the floating-point precision and interpolation scheme on the results of DNS of turbulence by pseudo-spectral codes, Comput. Phys. Commun. 177 (2007) 560–565.
[11] H. Homann, R. Grauer, A. Busse, W.C. Müller, Lagrangian statistics of Navier–Stokes and MHD turbulence, J. Plasma Phys. 73 (2007) 821–830.
[12] E. Hairer, C. Lubich, G. Wanner, Geometric Numerical Integration — Structure-Preserving Algorithms for Ordinary Differential Equations, second ed., Springer, 2006.
[13] R.I. McLachlan, G.R.W. Quispel, Geometric integrators for ODEs, J. Phys. A 39 (2006) 5251–5285.
[14] H. Yoshida, Construction of higher order symplectic integrators, Phys. Lett. A 150 (1990) 262–268.
[15] F. Lekien, J. Marsden, Tricubic interpolation in three dimensions, Int. J. Numer. Methods Eng. 63 (2005) 455–471.
[16] W. Kahan, R.-C. Li, Composition constants for raising the orders of unconventional schemes for ordinary differential equations, Math. Comput. 66 (1997) 1089–1099.
[17] S. Blanes, P. Moan, Practical symplectic partitioned Runge–Kutta and Runge–Kutta–Nyström methods, J. Comput. Appl. Math. 142 (2002) 313–330.