A linearly scaling grid method for generalized Langevin and other integrodifferential equations

2008 J. Phys. A: Math. Theor. 41 335005

(http://iopscience.iop.org/1751-8121/41/33/335005)

J. Phys. A: Math. Theor. **41** (2008) 335005 (16pp) doi:10.1088/1751-8113/41/33/335005

A linearly scaling grid method for generalized Langevin and other integrodifferential equations

Joshua Wilkie[1] **and Yin Mei Wong**[2]

[1] Department of Chemistry, Simon Fraser University, Burnaby, British Columbia, V5A 1S6, Canada
[2] Innovative Stochastic Algorithms, Surrey, British Columbia, V3V 3N4, Canada

E-mail: wilkie@sfu.ca and wong@www.isalgorithms.ca

Received 17 April 2008, in final form 19 June 2008
Published 17 July 2008
Online at stacks.iop.org/JPhysA/41/335005

Abstract

A grid-based method for numerical solution of systems of integrodifferential dynamical equations is introduced that has computational costs which scale linearly with the grid size. Grid parameters are automatically generated from a user supplied error tolerance. We illustrate the accuracy of the method by solving a variety of representative sets of equations. The method is much faster than comparable quadratically scaling algorithms.

PACS numbers: 02.50.Ey, 02.60.Nm, 02.60.Cb, 03.65.Yz

1. Introduction

Integrodifferential dynamical equations arise in many branches of physics [1–5], engineering [6], chemistry [7–11] and mathematical finance [12]. The generalized Langevin equation (GLE) [4] for the rate of change of an observable $P(t)$, i.e.,

$$\dot{P}(t) = -\int_0^t \mathrm{d}t'\gamma(t-t')P(t') + f(t), \tag{1}$$

which is used to model transport properties in many physical systems [1–3], is of this type. Here $P(t)$ could be the momentum, for example, of a particle in a viscous medium subject to a random force $f(t)$. The key feature of the GLE is the memory kernel $\gamma(t)$ which weights the contributions of previous values of the observable $P(t)$ to its current rate of change. Unless $\gamma(t)$ is of the form $\sum_j A_j\,\mathrm{e}^{-B_j t}$, in which case equation (1) can be reformulated as a larger set of ordinary differential equations [10], such equations are typically very difficult to solve. Formally, one may write the solution of (1) as an inverse Laplace transform

$$P(t) = \frac{1}{2\pi\mathrm{i}}\int_{c-i\infty}^{c+i\infty} \mathrm{d}z\,\mathrm{e}^{tz}[z+\Gamma(z)]^{-1}(P(0)+F(z)), \tag{2}$$

where $\Gamma(z)$, $F(z)$ are the Laplace transforms of $\gamma(t)$, $f(t)$. Formula (2) is rarely useful in practice, however, due to the difficulty of choosing c.

Instead, various quadratically scaling grid based methods are employed [1, 5, 6, 13]. For example, equation (1) can be rewritten in ordinary differential form [13]

$$\dot{P}(t) = -\lambda(t,0) + f(t) \tag{3}$$

$$\dot{\lambda}(t,u) = \gamma(u)P(t) + \partial\lambda(t,u)/\partial u \tag{4}$$

at the expense of introducing a new observable $\lambda(t,u)$ with an initial condition $\lambda(0,u) = 0$ and an auxiliary variable u. If u is then discretized on a grid and $\partial/\partial u$ represented by a matrix [13, 14], then equations (3)–(4) reduce to a set of ordinary differential equations. With some minor modifications [13] this approach becomes a general and fairly useful method. However, if n is the number of points in the grid then the computational costs will scale as n^2 which can be quite expensive for many problems. Also, there are additional parameters associated with such methods (e.g. l and g in [13]) which can be tricky to estimate.

However, suppose we define instead the Fourier transformed observable

$$\Lambda(t,s) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} \mathrm{d}u\, \mathrm{e}^{-\mathrm{i}su}\lambda(t,u) \tag{5}$$

which obeys

$$\dot{\Lambda}(t,s) = \Gamma(s)P(t) + is\Lambda(t,s), \tag{6}$$

where $\Gamma(s)$ is the Fourier transform of $\gamma(u)$. This equation when placed on a grid in s will have computational costs which scale linearly with the number of grid points. Moreover,

$$\lambda(t,0) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} \mathrm{d}s\, \Lambda(t,s) \tag{7}$$

can be implemented numerically as a sum over the s grid points, and hence this calculation will also scale linearly. This observation forms the basis of the numerical method we advocate here. It avoids all of the auxiliary parameters of [13] and can be implemented so that the grid size n and extent are chosen based on a user specified tolerance.

We show that generalizations of equations (3), (6) and (7) can be used as the basis of a fast and accurate numerical method. In section 2, we explain how the method works and how the parameters of the grids can be automatically chosen. In section 3 we solve a number of example problems. Section 4 explores the connection between the requested tolerance and the actual relative accuracy of the solutions.

2. Numerical method

The procedure we will outline should apply to a wide class of equations. Indeed, the examples are so varied that the method is best illustrated by considering individual cases. However, to give some concreteness to our initial discussion we can consider equations of GLE type, e.g.,

$$\dot{X}_j(t) = a_j(\mathbf{X}(t),t) - \int_0^t \mathrm{d}t'\, \gamma_j(t-t')b_j(\mathbf{X}(t'),t') + f_j(t) \tag{8}$$

for $j = 1, \ldots, N$ where $\mathbf{X}(t) = (X_1(t), \ldots, X_N(t))$ and $f_j(t)$ are colored noises.

The first step is to transform equations (8) into ordinary differential form by defining

$$\lambda_j(t,u) = \int_0^t \mathrm{d}t'\, \gamma_j(t-t'+u)b_j(\mathbf{X}(t'),t') \tag{9}$$

in which case we may rewrite equations (8) as

$$\begin{aligned}\dot{X}_j(t) &= a_j(\mathbf{X}(t), t) - \lambda_j(t, 0) + f_j(t)\\ \dot{\lambda}_j(t, u) &= \gamma_j(u) b_j(\mathbf{X}(t), t) + \partial\lambda_j(t, u)/\partial u\end{aligned} \tag{10}$$

with $\lambda_j(0, u) = 0$ by construction. Secondly, we define a Fourier transform

$$\Lambda_j(t, s) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \mathrm{d}u\, \mathrm{e}^{-\mathrm{i}us} \lambda_j(t, u) \tag{11}$$

in terms of which equations (10) become

$$\dot{X}_j(t) = a_j(\mathbf{X}(t), t) - \lambda_j(t, 0) + f_j(t) \tag{12}$$

$$\dot{\Lambda}_j(t, s) = \Gamma_j(s) b_j(\mathbf{X}(t), t) + is\Lambda_j(t, s) \tag{13}$$

with $\Lambda_j(0, s) = 0$, and $\Gamma_j(s)$ is the Fourier transform of $\gamma_j(u)$. Equation (13) does not involve partial derivatives such as those in equations (10). Note also that $\lambda_j(t, 0)$ is to be obtained by the inverse Fourier transform

$$\lambda_j(t, 0) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \mathrm{d}s\, \Lambda_j(t, s). \tag{14}$$

Implementation of equations (12)–(13) requires discretization of the variable s, and this discretization should be done for each j independently since $\Gamma_j(s)$ will be different for each equation (13). Since $\gamma_j(-t) = \gamma_j(t)$ in most cases, $\Gamma_j(s)$ will generally be symmetric in s. In addition equation (14) requires a symmetric integration over s. Hence, a symmetric grid of the form

$$s_j(k) = \left(-n_j^{\text{grid}} + k - 1\right) s_j^{\max} \big/ n_j^{\text{grid}} \tag{15}$$

for $k = 1, 2n_j^{\text{grid}} + 1$ seems appropriate. The integral in equation (14) then becomes

$$\lambda_j(t, 0) = \frac{\Delta s_j}{\sqrt{2\pi}} \sum_{k=1}^{2n_j^{\text{grid}}+1} \Lambda_j(t, s_j(k)), \tag{16}$$

where $\Delta s_j = s_j^{\max} \big/ n_j^{\text{grid}}$. Here it is worth noting that if $\gamma_j(t)$ is real and symmetric, and if $b_j(\mathbf{X}(t), t)$ is real then $\Lambda_j(t, -s) = [\Lambda_j(t, s)]^*$ and so equations for $s < 0$ need not be included.

We now have a set of equations, (12) and (13) for a finite set of s values, for which computational costs will scale linearly with the grid sizes n_j^{grid}. Finally then, we need some systematic way of choosing the $s_j^{\max}$ and n_j^{grid} which would allow the whole procedure to be automated.

2.1. Finding $s_j^{\max}$

If $\Lambda_j(t, s)$ were known for all t and s we could find $s_j^{\max}$ by looking at the decay of $\Lambda_j(t, s)$ to zero as $s \to \infty$. Since we do not know the $\Lambda_j(t, s)$, solutions of

$$\dot{\Lambda}_j^{\text{test}}(t, s) = \Gamma_j(s) + \mathrm{i}s\Lambda_j^{\text{test}}(t, s), \tag{17}$$

for which the factors $b_j(\mathbf{X}(t), t)$ in equation (13) have been dropped, seem the best alternative. It can be readily shown that

$$\operatorname{Re} \Lambda_j^{\text{test}}(t, s) = \frac{\sin st}{s} \Gamma_j(s). \tag{18}$$

Instead of looking at all possible t values we choose instead to take $t = t_{\text{end}}$ where t_{end} is the longest time of interest in the overall dynamics. Thus, we choose $s_j^{\max}$ so that the integral of the test function over this interval will be converged. In what follows we will assume that $\Gamma_j(s)$ is symmetric in s and we will thus integrate from zero to $s_j^{\max}$. In the event that $\Gamma_j(s)$ is asymmetric, integrations forward and backward from zero would need to be performed, but otherwise the method is the same.

For each j we pick $\Delta s_j = 2\pi/t_{\text{end}}$ as a first estimate. Using a variable step-size Runge–Kutta code [15] we integrate equation (18) to $m\Delta s_j$ and $(m+1)\Delta s_j$ for $m = 1, 2, \ldots$. Let $I_j(s) = \int_0^s \left|\frac{\sin vt_{\text{end}}}{v}\Gamma_j(v)\right| \mathrm{d}v$. For each m we check whether the relative error is less than a user defined relative tolerance tol_a, i.e., whether

$$|I_j((m+1)\Delta s_j) - I_j(m\Delta s_j)|/I_j((m+1)\Delta s_j) \leqslant tol_a. \tag{19}$$

When this has occurred three consecutive steps in a row we define $s_j^{\max} = (m+1)\Delta s_j$ where m is the final value employed.

2.2. *Finding n_j^{grid}*

Having obtained $s_j^{\max}$ we can now use it to find n_j^{grid}. We make an initial guess of say $n_j^{\text{grid}} = 10$. Using the Euler–Maclaurin summation formula we then compute

$$I_j^{\text{grid}}\left(s_j^{\max}\right) = \frac{\Delta s_j}{2}\left(\left|\operatorname{Re}\Lambda_j^{\text{test}}(t_{\text{end}}, 0)\right| + \left|\operatorname{Re}\Lambda_j^{\text{test}}\left(t_{\text{end}}, s_j^{\max}\right)\right|\right) \\ + \Delta s_j \sum_{k=n_j^{\text{grid}}+2}^{2n_j^{\text{grid}}} \left|\operatorname{Re}\Lambda_j^{\text{test}}(t_{\text{end}}, s_j(k))\right|, \tag{20}$$

where $s_j(k)$ are as defined in equation (15) and $\Delta s_j = s_j^{\max}\big/n_j^{\text{grid}}$. Next we compare $\left|I_j\left(s_j^{\max}\right) - I_j^{\text{grid}}\left(s_j^{\max}\right)\right|\big/\left|I_j\left(s_j^{\max}\right)\right|$ to a user defined relative tolerance tol_b with an index *iconv* which is initially set to zero. Upon failure or success with $iconv \leqslant 2$, n_j^{grid} is increased by one; while on success *iconv* is increased by one. We terminate the process when the relative error is less than tol_b three steps in a row, i.e. when $iconv = 3$. If $iconv \neq 0$ and a step fails we set $iconv = 0$ again.

3. Examples

We now have an automated algorithm for choosing the grids which we can test for a number of exactly solvable example problems. In all examples we took $tol_a = 10^{-5}$ and $tol_b = tol_a$. Numerical solutions of ordinary differential equations were obtained via Runge–Kutta [15] while stochastic equations were solved with ANISE [16]. The grid parameters computed for each problem are given in table 1.

3.1. *Non-Markovian Brownian motion*

We consider a number of examples which obey the one-dimensional Brownian motion Langevin equation [3]

$$\dot{X}(t) = P(t)/M \tag{21}$$

$$\dot{P}(t) = -\mathrm{d}V(X(t))/\mathrm{d}X(t) - \int_0^t \mathrm{d}t'\gamma(t-t')P(t') + F(t), \tag{22}$$

Table 1. Automatically generated $s^{\max}$ and n^{grid}.

Example	3.1.1	3.1.2	3.1.3	3.2	3.3
n_1^{grid}	3384	2963	3184	17916	820
$s_1^{\max}$	16.7552	10.1788	27.6460	3.0557	2.0944
n_2^{grid}	–	2472	3348	13251	440
$s_2^{\max}$	–	5.9062	40.2124	2.2599	2.3038
n_3^{grid}	–	–	–	16737	–
$s_3^{\max}$	–	–	–	2.8547	–

where $E[F(t)F(t')] = k_B T\gamma(t-t')$, and $E[\cdot]$ denotes an average over different realizations of the noise $F(t)$. $\gamma(t)$ is the memory function. We will consider three different models for the potential:

(a) $V(X) = (M/2)\omega_0^2X^2$ which corresponds to a particle in a Harmonic potential, and $\gamma(t) = \zeta\xi\,\mathrm{e}^{-\xi|t|}$;
(b) $V(X) = -(M/2)\omega_0^2X^2$ which corresponds to a particle in the vicinity of a parabolic barrier, and $\gamma(t) = \zeta\xi\,\mathrm{e}^{-\xi|t|}$; and
(c) $V(X) = \left(M\omega_0^2A^2/4\pi^2\right)[1-\cos(2\pi X/A)]$ which corresponds to a particle in a periodic potential of period A, and two different memory functions $\gamma(t) = \zeta\xi\,\mathrm{e}^{-\xi|t|}$ and $\gamma(t) = (\Omega^2\zeta/2\xi)\,\mathrm{e}^{-\xi|t|}[\cos(\omega_1 t) + (\xi/\omega_1)\sin(\omega_1|t|)]$ with $\omega_1 = \sqrt{\Omega^2-\xi^2}$.

Case (d) described below generalizes the above model to two spatial coordinates. In all of these models $\int_0^\infty \mathrm{d}t\,\gamma(t) = \zeta$ which can thus be regarded as the constant of friction. In order to simplify the forms of the equations we choose to work in scaled variables $x = X/A$, $p = P/(MA\omega_0)$, $\tilde{t} = \omega_0 t$, $\tilde{\zeta} = \zeta/\omega_0$, $\tilde{\xi} = \xi/\omega_0$, $\tilde{T} = 2k_BT\big/\left(M\omega_0^2A^2\right)$, $\tilde{\Omega} = \Omega/\omega_0$, $\tilde{\omega}_1 = \omega_1/\omega_0$ and $f = F\big/\left(M\omega_0^2A\right)$. However, for notational simplicity we will drop the tildes from everything which follows.

3.1.1. Cases (a) and (b). For simplicity we will first study a few examples without noise. Later we will couple the two models and add noise, but for now in scaled variables we consider

$$\dot{x} = p \tag{23}$$

$$\dot{p} = \mp x - \int_0^t \mathrm{d}t'\,\zeta\xi\,\mathrm{e}^{-\xi|t-t'|}p(t'), \tag{24}$$

where the minus sign applies to (a) while the plus sign applies to (b). Exact solutions of equation (24) can be obtained by defining $\lambda(t,0) = \int_0^t \mathrm{d}t'\,\zeta\xi\,\mathrm{e}^{-\xi|t-t'|}p(t')$ so that

$$\dot{x} = p \tag{25}$$

$$\dot{p} = \mp x - \lambda(t,0) \tag{26}$$

$$\dot{\lambda}(t,0) = \zeta\xi p - \xi\lambda(t,0) \tag{27}$$

with $\lambda(0,0) = 0$. We also chose $x(0) = 0.1$, $p(0) = 0.01$, $\zeta = 1$, and $\xi = 1$.

Similarly, to employ the grid method we define $\lambda(t,u) = \int_0^t \mathrm{d}t'\,\zeta\xi\,\mathrm{e}^{-\xi|t-t'+u|}p(t')$ which then yields the equation

$$\dot{\Lambda}(t,s) = \sqrt{\frac{2}{\pi}}\zeta\xi^2p(t)/(\xi^2+s^2) + \mathrm{i}s\Lambda(t,s), \tag{28}$$

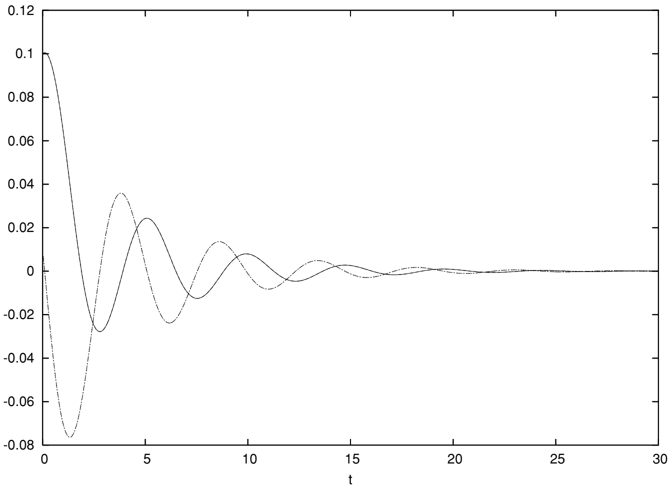


Figure 1. x and p versus t for the harmonic potential.

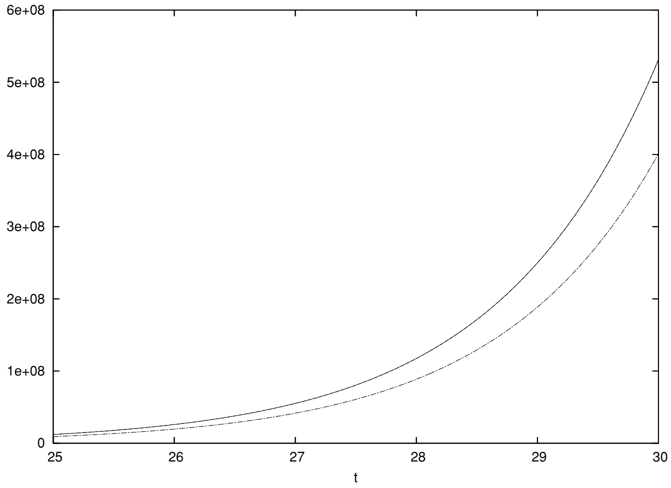


Figure 2. x and p versus t for the inverted harmonic potential.

for the Fourier transform which can be solved on a grid obtained as outlined above. We obtain $\lambda(t, 0)$ from $\Lambda(t, s)$ using equation (14). The parameters employed for (a) and (b) are those listed in table 1 in the column labeled 3.1.1.

In figure 1 we see the position (the exact is the dashed line, the grid method is solid) and momentum (the exact is the dot-dashed line, the grid method is dotted) of the particle in the harmonic potential. The particle undergoes four oscillations in the well while continuously losing amplitude and comes to rest at the bottom. It is impossible to visually distinguish the exact and grid method results.

Figure 2 shows the exact same quantities and line types for the parabolic barrier. Here the particle exits the barrier in the positive x direction, reaching great distances and huge

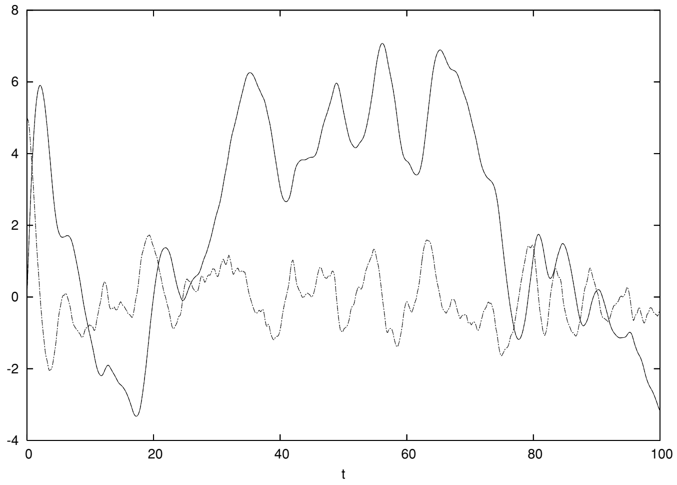


Figure 3. x and p versus t for the periodic potential, first memory function.

momentum in the same time. The figure shows results for the time interval [25, 30] where the solutions are largest. Again, it is not possible to visually distinguish the exact and grid results.

3.1.2. Case (c). Here we consider the problem of diffusion in a periodic potential [3]. In scaled variables we have

$$\dot{x} = p \tag{29}$$

$$\dot{p} = -(1/2\pi)\sin(2\pi x) - \int_0^t \mathrm{d}t'\,\gamma(t-t')p(t') + f \tag{30}$$

$$\dot{f} = g \tag{31}$$

where

$$\gamma(t) = \zeta\xi\,\mathrm{e}^{-\xi|t|} \tag{32}$$

$$g = -\xi f + \xi\sqrt{\zeta T}z(t) \tag{33}$$

for the first memory function, and

$$\gamma(t) = (\Omega^2\zeta/2\xi)\,\mathrm{e}^{-\xi|t|}[\cos(\omega_1 t) + (\xi/\omega_1)\sin(\omega_1|t|)] \tag{34}$$

$$\dot{g} = -\Omega^2 f - 2\xi g + \Omega^2\sqrt{\zeta T}z(t) \tag{35}$$

for the second. Here $E[z(t)z(t')] = 2\delta(t-t')$. In the simulations the equations for the noises f are integrated from $-t_{\mathrm{end}}$ to zero before actual dynamics of p and x begins. We also chose $x(0) = 0$, $p(0) = 5$, $\zeta = 1$, $T = 0.5066$ and $\xi = 1$. While we will show results for only a single realization of the noise the accuracy achieved is representative of a typical case.

Results for the first memory function are shown in figure 3. The grid parameters are n_1^{grid} and s_1^{max} of column 3.1.2 of table 1. Here again for x the exact result is dashed, and the grid result is solid. For p the exact is dot dashed and the grid result dotted. Initially, we see diffusion in the positive x direction with attenuation of momentum. Afterward fluctuations in

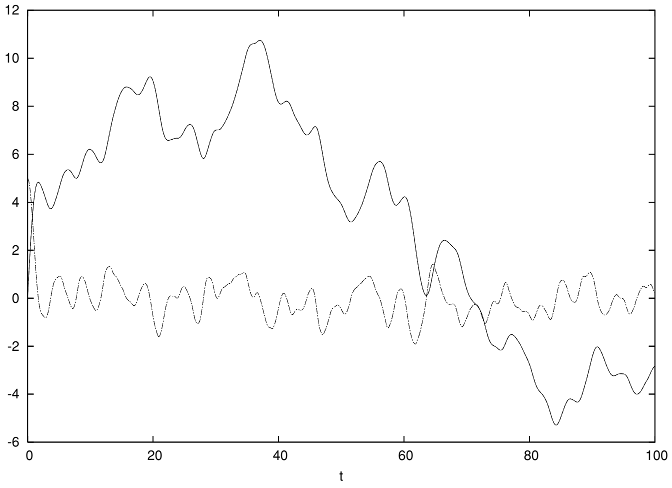


Figure 4. x and p versus t for the periodic potential, second memory function.

momentum are smaller, but we again see some diffusions as far as the seventh minimum in the positive x direction and third minimum in the negative x direction. There are no visible deviations between the grid method and exact results.

For the second memory function we set $\Omega^2 = 2.5$ and results are shown in figure 4. The grid parameters are n_2^{grid} and $s_2^{\max}$ of column 3.1.2 of table 1. The same convention is employed for line types. After a similar initial diffusion in position and decay of momentum, we see slower diffusion as far as the eleventh minima in the positive x direction, after which the particle moves all the way to the fifth well in the minus x direction. Again there are no visible discrepancies between the exact and grid results.

3.1.3. Case (d). Here we consider a set of coupled GLEs which arise in activated rate theories [11]

$$\dot{\alpha} = p_\alpha \tag{36}$$

$$\dot{p}_\alpha = -\partial V(x, y)/\partial\alpha - \lambda_\alpha(t, 0) + f_\alpha \tag{37}$$

$$\dot{f}_\alpha = -\xi_\alpha f_\alpha + \xi_\alpha\sqrt{\zeta T} z_\alpha, \tag{38}$$

where

$$\lambda_\alpha(t, u) = \int_0^t \mathrm{d}t'\, \gamma_\alpha(t - t' + u)p_\alpha(t') \tag{39}$$

with $E[f_\alpha(t)f_\alpha(t')] = T\gamma_\alpha(t - t')$, $E[z_\alpha(t)z_\alpha(t')] = 2\delta(t - t')$ and $E[z_x(t)z_y(t')] = 0$. Here $\alpha \in \{x, y\}$, $\gamma_\alpha(t) = \zeta\xi_\alpha\,\mathrm{e}^{-\xi_\alpha|t|}$ and $V(x, y) = (1/2)\big(-x^2+\omega_y^2y^2+2\mu xy\big)$. We chose $\zeta = T = 1$, $\xi_x = 1$, $\xi_y = 2$, $\omega_y = 2$, $\mu = 0.1$, $x(0) = 0$, $p_x(0) = 0.5$, $y(0) = 0.5$, $p_y(0) = 0$. Again the dynamics for the noises is integrated from $-t_{\text{end}}$ to 0 before the actual particle dynamics begins.

Exact solutions can be obtained by using

$$\dot{\lambda}_\alpha(t, 0) = -\xi_\alpha\lambda_\alpha(t, 0) + \zeta\xi_\alpha p_\alpha \tag{40}$$

and definition (39) in addition to equations (36)–(38).

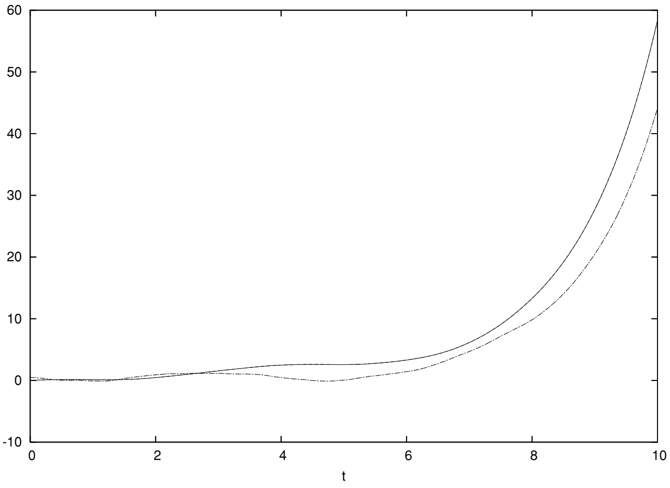


Figure 5. x and p_x versus t for the rate model.

The grid solutions for the Fourier transforms $\Lambda_\alpha(t, s)$ of equation (39), which obey

$$\dot{\Lambda}_\alpha(t, s) = \sqrt{\frac{2}{\pi}}\zeta\xi_\alpha^2 p_\alpha(t)\big/\big(\xi_\alpha^2 + s^2\big) + \mathrm{i}s\Lambda_\alpha(t, s), \tag{41}$$

give $\lambda_\alpha(t, 0)$ via equation (14). The grid parameters are given in table 1 under column 3.1.3; the first set is for $\gamma_x(t)$ and the second for $\gamma_y(t)$.

Figure 5 shows the unbounded x and p_x results for the rate theory model. We again employ the same line-type assignments. After some initial oscillation in the momentum we see departure in the positive x direction with the corresponding growth of momentum. The exact and grid results are not visibly different. In section 4 we will examine the relative errors for this case in more detail.

In figure 6 the bounded motions of y and p_y are shown. Again we employ the same line types. Here as in case 3.1 (a) we see oscillation about the minimum in both position and momentum, but without damping. Again, the accuracy is good.

3.2. Non-Markovian spin–spin-bath dynamics

The evolution of a qubit interacting with neighboring idle qubits in a isolated flawed quantum computer can be modeled as [9]

$$\dot{\sigma}_x = -B_0\sigma_y \tag{42}$$

$$\dot{\sigma}_y = -2B\sigma_z + B_0\sigma_x - 4C\lambda_y(t, 0) \tag{43}$$

$$\dot{\sigma}_z = 2B\sigma_y - 4C\lambda_z(t, 0) \tag{44}$$

$$\lambda_\alpha(t, u) = \int_0^t \mathrm{d}t'\,\gamma(t - t' + u)\sigma_\alpha(t'), \tag{45}$$

where $\alpha \in \{y, z\}$ and we will consider three different memory functions. Here σ_x, σ_y, and σ_z denote the expectations of the Pauli matrices for the central qubit. In all cases we chose $\sigma_x(0) = 1, \sigma_y(0) = \sigma_z(0) = 0, B = 0.06, C = 7.5 \times 10^{-4}, B_0 = 1$. Note also that this

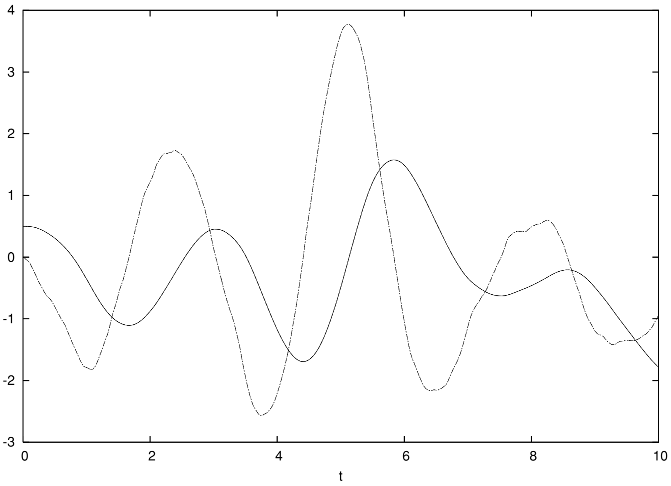


Figure 6. y and p_y versus t for the rate model.

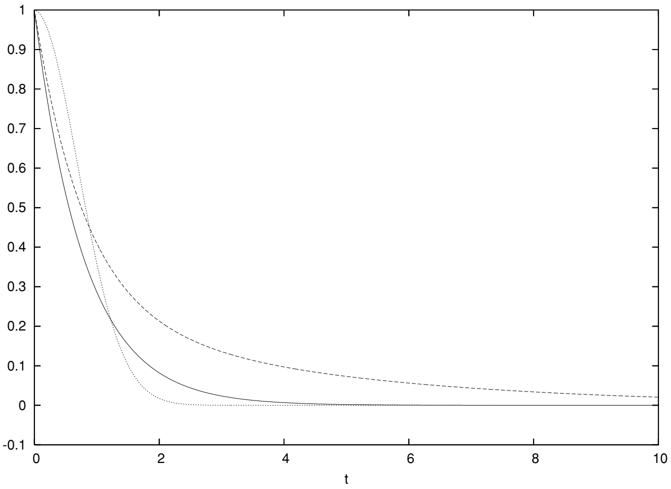


Figure 7. Memory functions versus t for the spin–spin bath.

is not a stochastic set of equations and that relaxation occurs slowly over a very long time scale.

Here we will examine dynamics for the three different memory functions shown in figure 7. One is a simple exponential (discussed in section 3.2.1, solid curve), one is a linear combination of two exponentials (discussed in section 3.2.2, dashed curve), and one is of Gaussian form (discussed in section 3.2.3, dotted curve). The three memory functions are quite similar at short time, and differ mainly in the tails, i.e. for $t > 2$. The grid parameters are given in column 3.2 of table 1 in the order that they appear below. The large $n^{\rm grid}$ values reflect the extremely long duration of the dynamics.

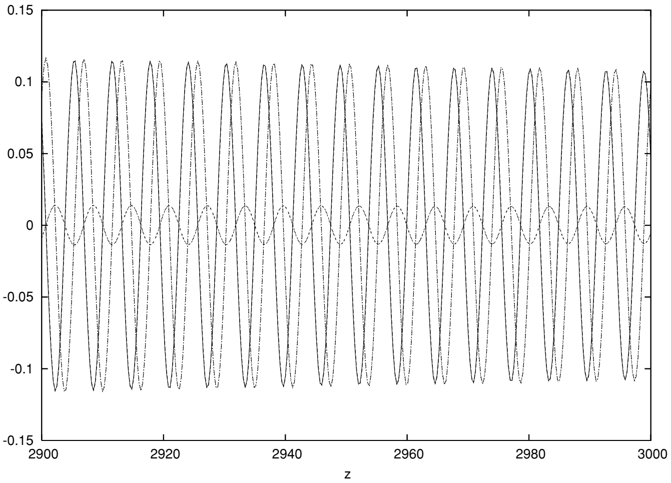


Figure 8. $\sigma_x, \sigma_y, \sigma_z$ versus t for the spin–spin-bath, first memory function.

3.2.1. Memory function $\gamma(t) = \mathrm{e}^{-\xi|t|}$. We choose $\xi = 1.25$. Exact solutions are obtained via equations (42)–(44) and

$$\dot{\lambda}_\alpha(t,0) = \sigma_\alpha - \xi\lambda_\alpha(t,0). \tag{46}$$

The Fourier transforms associated with (45) obey

$$\dot{\Lambda}_\alpha(t,s) = \sqrt{\frac{2}{\pi}}\frac{\xi}{\xi^2+s^2}\sigma_\alpha(t) + is\Lambda_\alpha(t,s) \tag{47}$$

from which we can obtain grid solutions for $\lambda_\alpha(t,0)$ via equation (14).

The dynamics of all three spin components for the first memory function are shown in figure 8. Note that only the last 100 time units are shown out of the total 3000. The σ_x exact result is plotted with dashed lines, while the grid result is solid. For σ_y the exact is dot dashed, the grid result dotted. For σ_z, with the lowest amplitude of the three, the exact is triple dashed and the grid result double dashed. All three components show persistent oscillation even at these large times, although the amplitudes of σ_x and σ_y are just a tenth of their earlier values. No significant disagreement is visible even at such long times.

3.2.2. Memory function $\gamma(t) = (3/4)\,\mathrm{e}^{-\xi_1|t|} + (1/4)\,\mathrm{e}^{-\xi_2|t|}$. Choosing $\xi_1 = 1.25, \xi_2 = 0.25$ we define

$$\lambda_{\alpha,1}(t,0) = (3/4)\int_0^t \mathrm{e}^{-\xi_1(t-t')}\sigma_\alpha(t')\,\mathrm{d}t' \tag{48}$$

$$\lambda_{\alpha,2}(t,0) = (1/4)\int_0^t \mathrm{e}^{-\xi_2(t-t')}\sigma_\alpha(t')\,\mathrm{d}t' \tag{49}$$

$$\lambda_\alpha(t,u) = \lambda_{\alpha,1}(t,0) + \lambda_{\alpha,2}(t,0), \tag{50}$$

so that exact solutions can be obtained via the above equations (42)–(44) and

$$\dot{\lambda}_{\alpha,1}(t,0) = (3/4)\sigma_\alpha - \xi_1\lambda_{\alpha,1}(t,0) \tag{51}$$

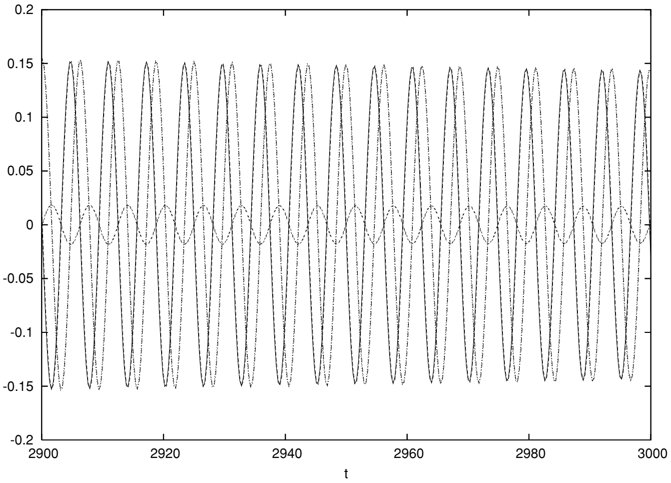


Figure 9. σ_x, σ_y, σ_z versus *t* for the spin–spin-bath, second memory function.

$$\dot{\lambda}_{\alpha,2}(t,0) = (1/4)\sigma_\alpha - \xi_2\lambda_{\alpha,2}(t,0). \tag{52}$$

To use the grid method note that the Fourier transforms of equation (50) obey

$$\dot{\Lambda}_\alpha(t,s) = \sqrt{\frac{2}{\pi}}\left(\frac{3/4\xi_1}{\xi_1^2+s^2} + \frac{1/4\xi_2}{\xi_2^2+s^2}\right)\sigma_\alpha(t) + \mathrm{i}s\Lambda_\alpha(t,s) \tag{53}$$

from solutions of which $\lambda_y(t,0)$ and $\lambda_z(t,0)$ can be obtained.

The spin components are plotted in figure 9 for the second memory function with the same line conventions. Here the memory function has a much longer tail, and this leads to less attenuation of the oscillations in σ_x and σ_y. Otherwise the dynamics resembles that of the first memory function. No disagreement is visible.

3.2.3. *Memory function* $\gamma(t) = \frac{\Delta\omega\,\mathrm{e}^{-\xi|t|}}{\sqrt{4\pi\beta}}\big(1 + 2\sum_{j=1}^{500}\mathrm{e}^{-(j\Delta\omega)^2/4\beta}\cos(j\Delta\omega t)\big)$.

Here the memory function is a good and analytically solvable approximation to $\mathrm{e}^{-\beta t^2}$ which arises in the actual physical problem [8, 13]. We define

$$\lambda_{\alpha,0}(t,u) = (1/2)\nu(0)\int_0^t \mathrm{e}^{-\xi(t-t'+u)}\sigma_\alpha(t')\,\mathrm{d}t' \tag{54}$$

$$\lambda_{\alpha,j,1}(t,u) = \nu(j)\int_0^t \mathrm{e}^{-\xi(t-t'+u)}\cos[j\Delta\omega(t-t'+u)]\sigma_\alpha(t')\,\mathrm{d}t' \tag{55}$$

$$\lambda_{\alpha,j,2}(t,u) = \nu(j)\int_0^t \mathrm{e}^{-\xi(t-t'+u)}\sin[j\Delta\omega(t-t'+u)]\sigma_\alpha(t')\,\mathrm{d}t' \tag{56}$$

$$\lambda_\alpha(t,u) = \lambda_{\alpha,0}(t,u) + \sum_{j=1}^{500}\lambda_{\alpha,j,1}(t,u) \tag{57}$$

with $\nu(j) = \frac{\Delta\omega}{\sqrt{\pi\beta}}\mathrm{e}^{-(j\Delta\omega)^2/4\beta}$, $\Delta\omega = 2\pi/300$, $\beta = 1$, $\xi = 0.025$ so that exact solutions can be obtained via (42)–(44) and

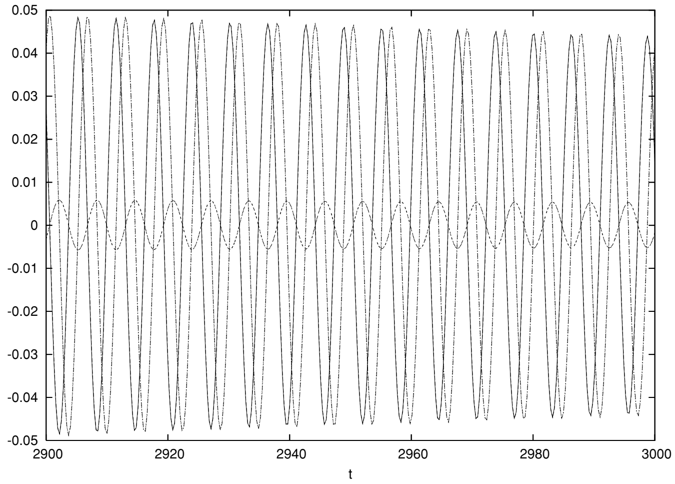


Figure 10. $\sigma_x, \sigma_y, \sigma_z$ versus t for the spin–spin-bath, third memory function.

$$\dot{\lambda}_{\alpha,0}(t,0) = (1/2)\nu(0)\sigma_\alpha - \xi\lambda_{\alpha,0}(t,0) \tag{58}$$

$$\dot{\lambda}_{\alpha,j,1}(t,0) = \nu(j)\sigma_\alpha - \xi\lambda_{\alpha,j,1}(t,0) - j\Delta\omega\lambda_{\alpha,j,2}(t,0) \tag{59}$$

$$\dot{\lambda}_{\alpha,j,2}(t,0) = -\xi\lambda_{\alpha,j,2}(t,0) + j\Delta\omega\lambda_{\alpha,j,1}(t,0). \tag{60}$$

The Fourier transforms of equation (57) obey

$$\dot{\Lambda}_\alpha(t,s) = \frac{1}{\sqrt{2\pi}}\sum_{j=-500}^{500}\nu(j)\frac{\xi}{\xi^2+(s-j\Delta\omega)^2}\sigma_\alpha(t) + \mathrm{i}s\Lambda_\alpha(t,s) \tag{61}$$

and $\lambda_y(t,0)$ and $\lambda_z(t,0)$ can be obtained in the usual way via equation (14).

This memory function is the most Markovian of the three, and consequently the attenuation seen in figure 10 is the largest. Again the accuracy is good.

3.3. Stochastic stock volatility with delay

A model [12] for the stochastic stock variance $\sigma^2(t)$ predicts

$$\mathrm{d}\sigma^2(t)/\mathrm{d}t = \epsilon V + (\alpha/\tau)[\lambda_\sigma(t,0) + (\mu - r)\tau]^2 - (\alpha+\epsilon)\sigma^2(t) \tag{62}$$

$$\lambda_\sigma(t,0) = \int_{t-\tau}^{t} \mathrm{e}^{-\xi(t-t')}\sigma(t')\,\mathrm{d}w(t') \tag{63}$$

with $E[\mathrm{d}w(t)] = 0$, $E[\mathrm{d}w(t)\,\mathrm{d}w(t')] = \mathrm{d}t\,\delta_{t,t'}$ from which it can be shown that the expected volatility $v(t) = E[\sigma^2(t)]$ obeys

$$\dot{v}(t) = \epsilon V + \alpha\tau(\mu - r)^2 + (\alpha/\tau)\lambda(t,0) - (\alpha+\epsilon)v(t) \tag{64}$$

$$\lambda(t,0) = \int_{t-\tau}^{t} \mathrm{e}^{-2\xi(t-t')}v(t')\,\mathrm{d}t'. \tag{65}$$

Note that equation (62) is nonlinear in the delay term (63). Since both of these equations are of integrodifferential form we employ a grid method for both and compare the average

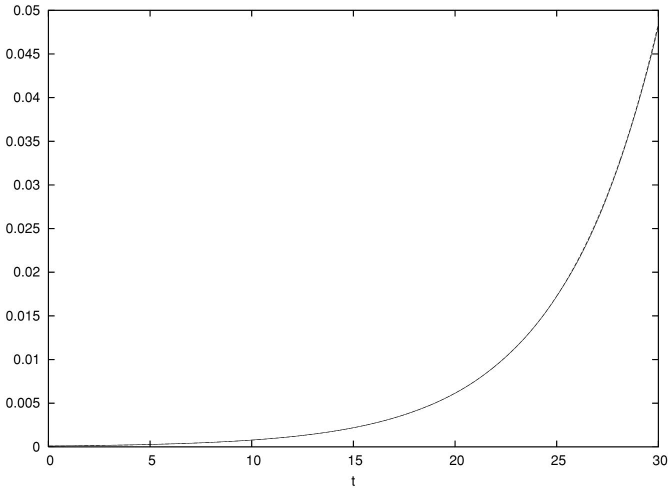


Figure 11. Mean volatility v_t versus time t.

of solutions of equation (62) for $\sigma^2(t)$ to the solution of equation (64). We used 1×10^6 trajectories to compute the average.

Grid solutions of (62) and (64) can be obtained by defining

$$\lambda_\sigma(t, u) = \int_{t-\tau+u}^{t} e^{-\xi(t-t'+u)}\sigma(t')\,dw(t') \tag{66}$$

$$\lambda(t, u) = \int_{t-\tau+u}^{t} e^{-2\xi(t-t'+u)}v(t')\,dt' \tag{67}$$

and their associated Fourier transforms which obey

$$d\Lambda_\sigma(t, s) = \sqrt{\frac{2}{\pi}}\frac{\xi}{\xi^2+s^2}\sigma(t)\,dw(t) + is\Lambda_\sigma(t, s)\,dt \tag{68}$$

$$\dot{\Lambda}(t, s) = \sqrt{\frac{2}{\pi}}\frac{2\xi}{4\xi^2+s^2}v(t) + is\Lambda(t, s). \tag{69}$$

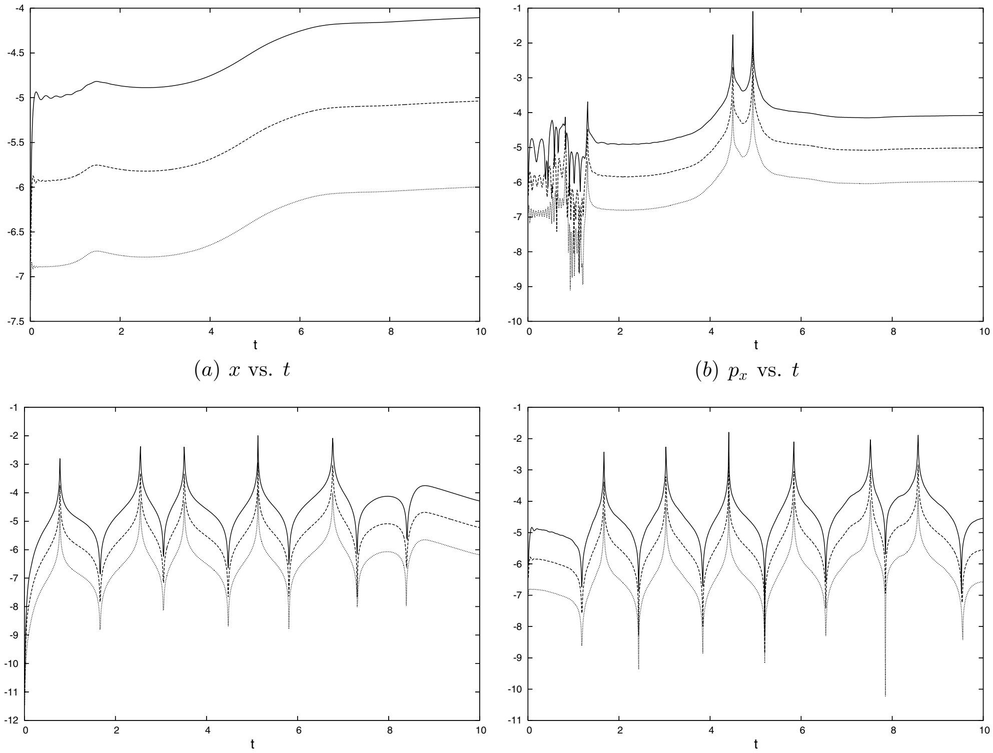
The grid parameters are given in column 3.3 of table 1 in the same order.

Following [12] where possible, we set $V = 0.00020991, r = 0.02, \tau = 1, \alpha = 0.060445, \epsilon = 0.012391, \mu = 0.000235, \xi = 0.005, \sigma_0 = 0.01, \sigma(t') = 0$ and $v(t') = 0$ for $t' < 0, v(0) = \sigma_0^2$.

As shown in figure 11, the mean volatility grows monotonically with time. The deterministic result is dashed while the stochastic average solution is solid. Clearly, the agreement is good.

4. Behavior of relative accuracy with decreasing $tol_a = tol_b$

Here we explore how the relative accuracy of solutions behaves with time and with changes in the requested tolerance. We focus on the rate problem since it should be the most difficult of the examples with two noises, two memory functions, two very different dynamical



(a) x vs. t

(b) p_x vs. t

(c) y vs. t

(d) p_y vs. t

Figure 12. Base ten logarithm of relative accuracy versus time *t* for three different tol_a.

Table 2. Automatically generated $s^{\max}$ and n^{grid}, for example 3.1.3, for different requested tolerances.

Tolerance	10^{-5}	10^{-6}	10^{-7}
n_1^{grid}	3184	14250	66497
$s_1^{\max}$	27.6460	56.5487	118.1239
n_2^{grid}	3348	11403	50099
$s_2^{\max}$	40.2124	83.5664	176.5575

diffusion distances and four equations. We thus perform calculations for this model with $tol_a = 10^{-5}$, $tol_a = 10^{-6}$ and $tol_a = 10^{-7}$. The grid parameters are shown in table 2.

The base ten logarithms of the relative accuracies are plotted in figure 12 for the four different observables x, p_x, y and p_y in (*a*), (*b*), (*c*), (*d*) respectively. Results for $tola = 10^{-5}$ are represented with solid lines, those with $tola = 10^{-6}$ with dashed, and those with $tola = 10^{-7}$ used dotted lines. In all cases the relative accuracies are initially at or exceed the requested tolerances, but as time increases an order of magnitude is generally lost. In addition, where curves cross zero a spike in the error is observed due to loss of significant figures with the decrease in magnitude of the solutions. This is a standard effect and away from such zeros one sees a return to good accuracy. In all cases, excepting near zeros, an accuracy representative of the requested tolerance is maintained.

5. Discussion

We have shown that a general linearly scaling grid method for integrodifferential equations, outlined in section 2, yields accurate solutions for a wide range of problems and memory functions. We have also shown that the accuracy of the solutions increases in a predictable way with a decrease in the user supplied tolerance tol_a. Although the grid sizes employed are sizable (e.g. typically a few thousand) the computational costs are quite reasonable. For the simple damped harmonic oscillator, for example, about 3300 grid points are used and thus about as many uncoupled equations must be solved, requiring about the same number of operations per call to the Runge–Kutta derivative function [15]. By contrast, a quadratically scaling method like [13] with the same number of grid points would require 10^7 operations per call to the derivative function, and the equations are all coupled meaning that many more such calls can be expected. It must also be emphasized that this is a fair comparison, since both grids are equivalent as a consequence of their Fourier transform relationship as long as $\Delta u \Delta s = 2\pi/(2n^{\text{grid}} + 1)$. In other words, if good results are obtained with a given n^{grid} by the quadratically scaling method, the accuracy will be precisely the same as that which could be obtained with the same n^{grid} via the linearly scaling method [13]. Finally, in this linearly scaling method all parameters are automatically generated, and so there are no difficult to estimate auxiliary parameters such as appear in other methods [1, 5, 6, 13]. Thus, this linear scaling method appears to be a significant improvement over existing techniques for obtaining high-accuracy solutions of integrodifferential dynamical equations.

Acknowledgments

JW acknowledges the support of the Natural Sciences and Engineering Research Council of Canada.

References

[1] Munakata M and Kawakatsu T 1985 *Prog. Theor. Phys.* **74** 262

[2] Frenkel G and Schwartz M 2000 *Europhys. Lett.* **50** 628

[3] Igarashi A and Munakata T 1988 *J. Phys. Soc. Japan* **57** 2439

[4] See Pechukas P 1967 *Phys. Rev.* **164** 174, and references therein

[5] Chin R C Y, Hedstrom G W and Majda G 1986 *Geophys. J. R. Astron. Soc.* **86** 137

[6] Shaw R and Whiteman J R 1997 *Comput. Methods Appl. Mech. Eng.* **150** 397

[7] Martens C C 2002 *J. Chem. Phys.* **116** 2516

[8] Wilkie J 2000 *Phys. Rev.* E **62** 8808
Wilkie J 2001 *J. Chem. Phys.* **114** 7736
Wilkie J 2001 *J. Chem. Phys.* **115** 10335

[9] Cetinbas M and Wilkie J 2007 *Phys. Lett.* A **370** 194
Cetinbas M and Wilkie J 2008 *J. Phys. A: Math. Theor.* **41** 065302

[10] Meier C and Tannor D J 1999 *J. Chem. Phys.* **111** 3365

[11] Berezhkovskii A M, Pollak E and Zitserman V Yu 1992 *J. Chem. Phys.* **97** 2422

[12] Swishchuk A 2005 *Wilmott Mag.* **19** 63

[13] Wilkie J 2003 *Phys. Rev.* E **68** 027701

[14] Colbert D T and Miller W H 1992 *J. Chem. Phys.* **96** 1982

[15] dop853.f from www.unige.ch/~hairer/software.html

[16] ANISE from www.isalgorithms.ca