

Langevin Stabilization of Multiscale Mollified Molecular Dynamics

Jesús A. Izaguirre

*Department of Computer Science and Engineering
University of Notre Dame, Notre Dame, IN 46556, USA*

Abstract. This paper shows the possibility of using very mild stochastic damping to stabilize long time steps integrators for Newtonian molecular dynamics. More specifically, stable and accurate integrations are obtained for damping coefficients that are only a few percent of the natural decay rate of the velocity autocorrelation function. Langevin Molly (LM) is introduced in this paper. It uses the mollified impulse method for the Newtonian term and the Langevin impulse method for the Langevin term. A parallel version of LM is available in the molecular dynamics program NAMD 2.1. LM and the Langevin integrators BBK and LN are evaluated across a wide range of damping coefficients values. Using LM and mild damping, time steps of up to 16 fs are possible even in the presence of explicitly modeled flexible water. When using mild damping, LM is superior to the other methods because it is symplectic in the zero damping limit, and it uses a better integration for the Langevin damping term. The second part of this paper compares several integrators (LM, BBK, and LN) for regular Langevin dynamics, a typical application of which is the implicit modeling of solvent molecules. In this case, the method called LN is superior, with LM closely following.

1 Introduction

This paper shows the possibility of using very mild stochastic damping to stabilize long time steps integrators for Newtonian molecular dynamics. More specifically, stable and accurate integrations are obtained for damping coefficients that are only a few percent of the natural decay rate of the velocity autocorrelation function. Langevin Molly (LM) is introduced in this paper, which uses the mollified impulse method for the Newtonian term and the Langevin impulse method for the Langevin term. LM and the Langevin integrators BBK and LN are evaluated across a wide range of damping coefficients values. Using LM and mild damping, time steps of up to 16 fs are possible even in the presence of explicitly modeled flexible water. Furthermore, spectral analyses of the trajectories generated by LM and the Verlet/r-RESPA Impulse method are shown to be indistinguishable. LM is superior to the other methods for Langevin stabilization because it is symplectic in the zero damping limit, and it uses a better integration for the Langevin damping term. The second part of this paper compares several integrators (LM, BBK, and LN) for regular Langevin dynamics, a typical application of which is the implicit modeling of solvent molecules. In this case, the method called LN is superior, with LM closely following.

2 Motivation

Molecular dynamics requires the solution of Newton’s equations of motion for a classical unconstrained simulation:

$$M \frac{d^2}{dt^2} X(t) = -\nabla U(X(t)) \quad (1)$$

where M is a diagonal matrix of atomic masses, $x = X(t)$ are the atomic trajectories, and the potential field U is typically given by

$$U = U^{\text{bnd}} + U^{\text{nonbnd}}, \quad (2)$$

$$U^{\text{bnd}} = U^{\text{bond}} + U^{\text{angle}} + U^{\text{dihedral}} + U^{\text{improper}}, \quad (3)$$

$$U^{\text{nonbnd}} = U^{\text{Lennard-Jones}} + U^{\text{electrostatic}}. \quad (4)$$

The Verlet-I/r-RESPA multiple time stepping impulse method splits the force into different components whose dynamics correspond to different time scales, which are then represented as appropriately weighted impulses (with weights determined by consistency). The impulse method is

$$M \frac{d^2}{dt^2} X = - \sum_{n'=-\infty}^{\infty} \delta t \delta(t - n' \Delta t) \nabla U^{\text{fast}}(X) - \sum_{n=-\infty}^{\infty} \Delta t \delta(t - n \Delta t) \nabla U^{\text{slow}}(X) \quad (5)$$

where the partitioning of U into U^{fast} and U^{slow} is chosen so that an appropriate time step Δt for the slow part of the force is larger than a time step δt for the fast part.

Verlet-I/r-RESPA was proposed but not implemented by the authors of [1] and [2] and independently discovered by the authors of [3], who also demonstrated its usefulness. It permits an increase to 4 fs in the length of the longest time step Δt . When the method was introduced, it was predicted that there would occur resonances that might induce instability if the frequency of the slow force impulse coincides with a normal mode frequency of the system [4]. Resonance produces an oscillation in the positions whose amplitude increases with time. More surprisingly, there is also a problem for long time steps just smaller than half the period of the fastest normal mode [5, 6]. There is also empirical evidence that time steps of 5 fs or greater are not possible with this method [7].

2.1 MOLLY: The mollified impulse method

MOLLY is a family of integrators that counteracts the instabilities present in the multiple time stepping Verlet-I/r-RESPA integrator. This is accomplished by perturbing the potential at time averaged positions

$$U^{\text{slow}}(x) \rightarrow U^{\text{slow}}(\mathcal{A}(x)), \quad (6)$$

with the force defined as a gradient of this averaged potential,

$$-\nabla U^{\text{slow}}(x) \rightarrow -\mathcal{A}_x(x)^{\text{T}} \nabla U^{\text{slow}}(\mathcal{A}(x)). \quad (7)$$

This perturbation is supposed to compensate for finite Δt artifacts. Perturbing the potential rather than the force ensures that the numerical integrator is symplectic [8]. The force used by MOLLY is the gradient of the perturbed potential [9]. MOLLY can

be seen as a filter that eliminates components of the slow force impulse in the directions of the fast forces, and thus improves the stability of Verlet-I/r-RESPA. Different averaging functions give rise to MOLLY integrators with different stability and accuracy properties. For instance, an averaging based on spectral methods allows very precise filtering, but due to the diagonalizing of a Hessian, is computationally expensive. [5] suggests using an averaging function $\mathcal{A}(x)$ that takes into account the motion induced by some of the fast forces (those that are the gradient of a simpler energy, U^{fastest}). This reduced system is then integrated over a short time span (typically for an interval of order Δt) using weight functions with local support in time (e.g. B-splines). The main computational expense of these B-spline averagings is the computation of force Hessian-vector products (needed to generate the filter $\mathcal{A}_x(x)$). These averagings overcome the 5 fs barrier and their effectiveness is related to the extensiveness of the time averaging [10].

A better time averaging would eliminate the high frequency components of $-U_x^{\text{slow}}$ completely. A good starting point is the Backward Euler (BE) averaging:

$$\mathcal{A}(x) = x - \Delta t^2 M^{-1} U_x^{\text{fastest}}(\mathcal{A}(x)). \quad (8)$$

Assume that U^{fastest} can be written as $U^{\text{fastest}} = \chi(g(x))$, where $g(x)$ is a vector of independent length constraint functions. The elements of $g(x)$ are of the form

$$g^k(x) = x^T G_k x - l_k^2 \quad (9)$$

where the G_k are symmetric matrices and l_k are rest lengths. Allowing $\Delta t \rightarrow \infty$ gives the equations for $\mathcal{A}(x)$ and μ

$$\begin{cases} M(\mathcal{A}(x) - x) + g_x^T(\mathcal{A}(x))\mu = 0, \\ g(\mathcal{A}(x)) = 0, \end{cases} \quad (10)$$

which are termed *Equilibrium**; this method completely eliminates components of $-\nabla U^{\text{slow}}$ in the directions of the constraints g_x^T . [11] presents a proof of the stability of *Equilibrium** for a general linear model problem under the assumption $U^{\text{fastest}} = U^{\text{fast}}$ (all the fast forces included in the mollification). An outline of this proof is given in Appendix A. The stability condition for *Equilibrium** MOLLY on the longest time step ($\Delta t \rho (M^{-1/2} U_{xx}^{\text{slow}} M^{-1/2})^{1/2} < 2$) is less restrictive than that of the popular leapfrog integrator ($\Delta t \rho (M^{-1/2} U_{xx} M^{-1/2})^{1/2} < 2$), because it is limited by the fastest frequencies in the slow forces ($-\nabla U^{\text{slow}}$) rather than by the overall fastest frequencies (in $-\nabla U^{\text{fast}}$). This paper presents an attempt to further stabilize MOLLY by adding a mild Langevin damping term.

2.2 Scope of this study

This paper addresses two applications of Langevin dynamics:

1. Langevin stabilization; a procedure that uses a very mild damping to stabilize multiple time stepping (MTS) methods. The assumption is that this damping does not significantly affect the dynamics of the system.
2. Regular Langevin dynamics. A typical application is simulating a system with the solvent (for example, water) omitted and, instead, modeling the effects of the solvent using Langevin dynamics [12, 13]. Another application is to treat a system with stochastic boundary conditions.

2.2.1 Langevin Stabilization

The use of mild Langevin damping to make MOLLY stable for longer time steps (cf. [13, 14]) is called here *Langevin stabilization*. In the limit of $\gamma = 0$, Newtonian mechanics would be recovered if it were not for the temperature control mechanism of Langevin dynamics. A prescription for estimating appropriate values of γ for *Langevin stabilization* of Newtonian dynamics is to choose γ much smaller than the decay rate of the velocity autocorrelation function. In the simulations reported here, the decay rate is estimated as 10 ps^{-1} . Thus, values of γ such as 0.2, 0.5, and 1 ps^{-1} correspond to adding 2%, 5%, and 10% error to the decay rate. A similar measurement of the decay rate of the velocity autocorrelation function appears in [15], where it is found that for every atom in a protein simulation the velocity autocorrelation function exhibits a decay rate with a relaxation time of less than 0.1 ps. The maximum values of Δt allowed with very small values of γ (e.g. 0.2, 0.5, and 1 ps^{-1}) were estimated experimentally. Numerical stability establishes a limit on how small γ can be for a given value of Δt , or equivalently, on how large Δt can be for a given value of γ . This can be intuitively grasped by recalling that in the limit $\gamma = 0$, Newtonian dynamics are recovered. Thus, any instabilities of the underlying MTS integrator will become apparent as γ approaches zero. The preceding observations are confirmed in numerical experiments for several integrators.

For some applications, notably hydrodynamics, the addition of even a mild Langevin term may cause undue damping, since the velocity autocorrelation function decays much more slowly, as $t^{-3/2}$ for 3-D liquids, cf. [16, p. 310]. However, for simulations of biological systems this long tail effect should not be important as long as one is dealing with mixing systems.

2.2.2 Regular Langevin Dynamics

Langevin dynamics can be used to implicitly model the effect of a solvent on a biomolecule. Solvent molecules are left out, and their effect on the dynamics of the solute macromolecule is incorporated by an electrostatics *potential-of-mean-force term* added to $U(x)$, by *random collisions* (the random force vector below), and by a *frictional drag on its motion through the solvent* (the friction term $-\gamma MV(t)$ below). This behavior can be reproduced by a system of stochastic differential equations given by

$$M dV(t) = -\nabla U(X(t))dt - \gamma MV(t)dt + \sqrt{2\gamma k_B T} M^{1/2} dW(t), \quad (11)$$

where $X(t)$ is the position, $V(t)$ is the velocity, γ is the collision parameter, $W(t)$ is a vector of independent standard Wiener processes (cf. [17, p. 28]), k_B is the Boltzmann constant, and T is the Langevin bath temperature. These equations correspond to a constant temperature ensemble.

In the classical solvent modeling application of the Langevin Eq. (11) the frictional term is assumed to be proportional to the velocity of the system. The constant of proportionality is given by γ/mass where γ is the collision frequency (or parameter). It is useful to define γ^{-1} as the velocity relaxation time: This is the time it takes for the motion to be damped (or the memory of the velocity of the particle to be lost). A typical value used for water as a solvent is suggested in [18] to be $\gamma = 50 \text{ ps}^{-1}$. Too large γ overdamps the low-frequency vibrational modes. For example, spectral densities appear smoother, and so does the decay of various correlations functions [13].

3 Description of Integrators for Langevin Dynamics

The integrators for the Langevin term that were tested are described here. The first was the method BBK [19]. BBK was combined with both the Verlet-I/r-RESPA method (BBK-I) and with *Equilibrium** MOLLY (BBK-M).

It is reasonable to expect an integrator for Langevin dynamics be exact for constant force. [14] proposes one such single time stepping scheme, Langevin Impulse (LI). This method was implemented and analyzed by one of the authors in [11]. Another example of such a method is the one proposed in [20]. LM is a multiple time stepping (MTS) combination of LI with *Equilibrium** MOLLY.

3.1 BBK

Different discretizations of Eq. (11) are available in the literature, but one of the most popular is the so called BBK integrator:

half a kick:

$$V^{n+\epsilon} = \left(1 - \frac{1}{2}\gamma\Delta t\right)V^n + \frac{1}{2}\Delta t M^{-1} \left(-\nabla U(X^n) + \sqrt{\frac{2\gamma k_B T}{\Delta t}} M^{1/2} Z^n\right). \quad (12)$$

drift:

$$X^{n+1} = X^n + \Delta t V^{n+\epsilon}. \quad (13)$$

half a kick:

$$V^{n+1} = \left(1 + \frac{1}{2}\gamma\Delta t\right)^{-1} \left(V^{n+\epsilon} + \frac{1}{2}\Delta t M^{-1} \left(-\nabla U(X^{n+1}) + \sqrt{\frac{2\gamma k_B T}{\Delta t}} M^{1/2} Z^{n+1}\right)\right). \quad (14)$$

Z^n represents a vector of independent Gaussian random numbers of mean zero and variance one.

3.2 LI: Langevin impulse

This method is exact for constant forces, it is based on the idea of splitting, and it samples from a *bivariate* Gaussian distribution. As a starting point, Eq. (11) is simplified and rewritten as a system of two coupled first order differential equations:

$$dx = v dt, \quad dv = M^{-1} F(x) dt - \gamma v dt + \sqrt{2\gamma k_B T} M^{-1/2} dW(t). \quad (15)$$

The LI discretization of this equation consists of half a kick, a “fluctuation”, followed by another half a kick. The method is

half a kick:

$$v^{n+\epsilon} = v^n + \frac{1}{2}\Delta t M^{-1} F^n, \quad (16)$$

fluctuation:

$$v^{n+1-\epsilon} = e^{-\gamma\Delta t} v^{n+\epsilon} + \sqrt{2\gamma k_B T} M^{-1/2} R_1^{n+1}, \quad (17)$$

$$x^{n+1} = x^n + \Delta t \frac{1 - e^{-\gamma\Delta t}}{\gamma\Delta t} v^{n+\epsilon} + \sqrt{2\gamma k_B T} \frac{1}{\gamma} M^{-1/2} R_2^{n+1}, \quad (18)$$

half a kick:

$$F^{n+1} = F(x^{n+1}), \quad (19)$$

$$v^{n+1} = v^{n+1-\epsilon} + \frac{1}{2}\Delta t M^{-1} F^{n+1}. \quad (20)$$

where

$$R_1^{n+1} = \int_{t^n}^{t^{n+1}} e^{-\gamma(t^{n+1}-t)} dW(t), \quad R_2^{n+1} = \int_{t^n}^{t^{n+1}} (1 - e^{-\gamma(t^{n+1}-t)}) dW(t), \quad (21)$$

and where $t^{n+1} = t^n + \Delta t$. R_1^{n+1} and R_2^{n+1} are joint Gaussian random variables of zero mean and some covariance matrix C . The procedure for generating them is summarized in Appendix B.

3.3 LN: An extrapolative MTS integrator for Langevin dynamics

LN is a method introduced in [6] and [13]. It is a triple time stepping method for Langevin dynamics that uses constant extrapolation (CE) on the outer time step, mid-point extrapolation (ME) on the medium time step, and position Verlet on the innermost time step. This particular formulation is that of [21]. The motivation of LN is to avoid the instability due to the impulsive nature of Verlet-I/r-RESPA by using extrapolative methods. CE and ME used in a linear problem are unstable even for a scalar problem; however, for a linear problem the magnitude of the eigenvalues of the propagation matrix is bounded independently of the long time step Δt . To use CE and ME in a scalar problem, and be stable, a moderate amount of damping is needed. The instability of ME at small outer time steps is milder than that of CE, even though it is not bounded. These results motivate the choice of integrators for the different force components in LN: The slow forces with longest time step are integrated using CE, whereas the medium forces are integrated with ME.

Assume that the medium time step $\Delta t_m = k_1 \delta t$ and the longest time step $\Delta t = k_2 \Delta t_m$. Then, given positions X and velocities V , one long step of the method consists of

slow force evaluation:

$$F^{\text{slow}} := F^{\text{slow}}(X), \quad (22)$$

followed by k_2 steps of {

medium force evaluation:

$$F^{\text{med}} := F^{\text{med}}\left(X + \frac{1}{2}\Delta t_m V\right). \quad (23)$$

followed by k_1 steps of {

half a drift:

$$X := X + \frac{1}{2}\delta t V, \quad (24)$$

kick:

$$F^{\text{fast}} := F^{\text{fast}}(X), \quad F^{\text{rand}} := \sqrt{\frac{2\gamma k_B T}{\delta t}} M^{1/2} Z^n, \quad (25)$$

$$F := F^{\text{slow}} + F^{\text{med}} + F^{\text{fast}} + F^{\text{rand}}, \quad (26)$$

$$V := (1 + \gamma\delta t)^{-1} \left(V + \frac{1}{2}\delta t M^{-1} F \right). \quad (27)$$

half a drift:

$$X := X + \frac{1}{2}\delta t V. \quad (28)$$

}
}

4 Empirical evaluation of methods using Langevin damping

Table 1 lists 4 integrators for Langevin dynamics that have been implemented and tested for this paper. Most experiments were performed using *flexible water*; that is, vibrations of bonds and angles are included in the water model. Earlier experiments suggest that flexible water models are particularly sensitive to destabilizing artifacts in numerical integrators [7]. [9] reported the successful use of time steps of 6 fs using *Equilibrium** MOLLY. Apparently, not using all of U^{fast} in the mollification causes the remaining instability in MOLLY. This is due to an imperfect filter for excitations to the fastest modes of U^{fast} , i.e. the eigenvectors of U^{fastest} approximate crudely those of U^{fast} .

Table 1: Langevin dynamics integrators tested.

Nickname	Integrators for (Langevin, Newtonian) terms
BBK-I	(BBK, Verlet-I/r-RESPA)
BBK-M	(BBK, <i>Equilibrium*</i> MOLLY)
LM	(LI, <i>Equilibrium*</i> MOLLY)
LN	(BBK, constant extrapolation and midpoint extrapolation)

For each method the empirical results determine a stability limit, that is, the maximum Δt that for a value of the collision parameter γ gives a stable integration. The smaller the value of γ the more accurately the Langevin dynamics approximate the Newtonian dynamics. The recipe was to choose γ as a fraction of the decay rate of the velocity autocorrelation function, which was estimated from simulations of the test problem as 10 ps^{-1} . Thus, values of γ of 0.2, 0.5, and 1 ps^{-1} are equivalent to making an error of 2%, 5%, and 10%, respectively.

The criterion used to measure stability is to consider the worse of the errors in average total energy and temperature. In particular, the deviation from the “correct” average energy is computed by comparing against a trajectory with $\Delta t = 1 \text{ fs}$ for a given γ . An error of at most 3% is considered stable. This is an indirect measure of instability since Langevin dynamics tends to keep the energy bounded. A significant rise in the mean energy is indeed a sign of a failure to keep the energy bounded. Similarly, the error of the average simulated temperature with respect to the Langevin temperature was computed. In general, the error in the energy is a stricter measure of accuracy and stability than is the error in the temperature.

4.1 Results with Langevin stabilization

The results for Langevin stabilization are summarized in the first 3 numerical rows of Tables 2 and 3. Note that BBK-I and LN are not suitable methods for Langevin stabilization. On the other hand, Langevin stabilization of MOLLY is very successful: long time steps of 15 or 16 fs are possible with mild damping ($\gamma = 0.5$ or 1 ps^{-1} , *respectively*). MOLLY has been tested with both BBK (BBK-M) and an Langevin Impulse method

Table 2: **High accuracy results** for Langevin dynamics methods: maximum time step Δt for error of 0 – 0.4% in the energy.

γ \Method	LM	BBK-M	BBK-I	LN
0.2	10	8	N/A	N/A
0.5	12	11	N/A	N/A
1	15	11	5	N/A
5	15	9	7	N/A
10	15	16	7	6
20	15	16	8	10
50	20	17	8	50
100	25	6	4	

Table 3: **Medium accuracy results** for Langevin dynamics methods: maximum time step Δt for error of 4 – 5% in the energy.

γ \Method	LM	BBK-M	BBK-I	LN
0.2	14	12	N/A	N/A
0.5	16	15	N/A	N/A
1	16	16	N/A	N/A
5	17	18	N/A	7
10	17	18	11	15
20	25	25	17	21
50	35	18	16	50
100	40	7	N/A	50

(LM): LM is more accurate and able to take longer time steps Δt in all cases. The method termed LN does not produce satisfactory results for Langevin stabilization, since the underlying extrapolative nonsymplectic integrator is unstable even for linear problems, which can only be overcome with a relatively large damping γ (of 20 ps⁻¹ or greater).

4.2 Results with Langevin dynamics

The results for regular Langevin dynamics are summarized in the last 5 rows of Tables 2 and 3. LM is successful here also: with $\gamma = 50$ ps⁻¹, time steps as large as 35 fs are possible. This is superior to BBK-I or BBK-M, for which the maximum time step is 16 fs. In general, LM allows greater γ and Δt to be taken than BBK-I and BBK-M. For example, time steps of 50 fs can be taken with $\gamma = 100$ ps⁻¹ using LM. A summary of the results appear in Fig. 1. For the modeling of an implicit solvent, LM is inferior to LN. The latter allows long time steps of 50 fs for the same levels of accuracy with $\gamma = 50$ ps⁻¹. These results are consistent with [13], which reports long time steps using LN of 12 fs for flexible water, 48 fs for a biomolecule in water, and 96 fs for a biomolecule in vacuum.

4.3 Implementation and Performance

All the methods described here are implemented in a sequential testing problem called SAMD (<http://www.cse.nd.edu/~izaguirr/research>). An implementation of *Equilibrium* MOLLY into the parallel molecular dynamics program NAMD 2.1 was tested using a protein and water with 17527 atoms, both by itself and with Langevin damping (BBK-M) [22]. Timings are shown in Table 4, whereas speedups are shown in Table 5.

Parallelism is achieved by doing a spatial decomposition of the system into boxes called “patches” with the constraint that each H group is kept in a single patch. An

Figure 1: Maximum achievable Δt for three Langevin integrators. The criterion was that the error in the average energy was no more than 3% with respect to the average obtained with leapfrog and $\Delta t = 1$ fs.

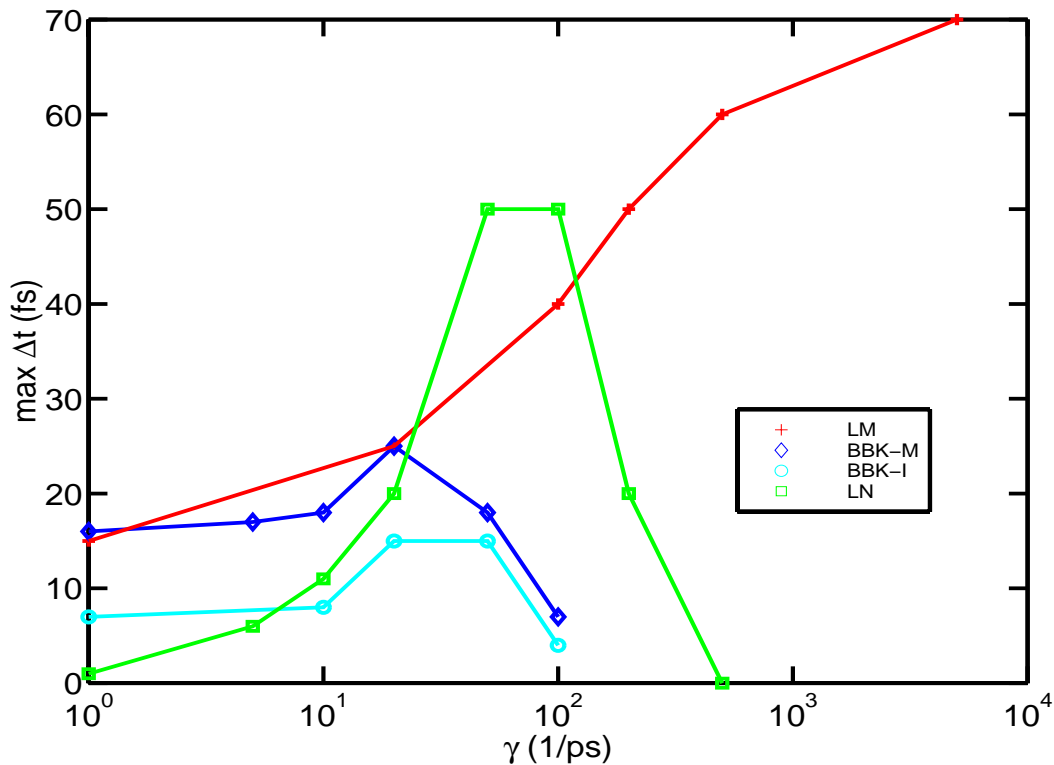


Table 4: **CPU times** for NAMD 2.1 on a protein and water system with 17527 atoms for 1008 fs. The machine was an HP K-box.

Method	$\Delta t, \delta t$	p=1	2	4
Verlet/r-RESPA	4,1	23783.8	14052.9	7425.3
<i>Equilibrium</i>	6,1	17929.4	10437.2	5644.0
BBK-M	16,1	8684.3	4949.3	2704.4
BBK-M	16,2	7305.2	4232.3	2266.7

Table 5: **Speedups** for NAMD 2.1 on a protein and water system with 17527 atoms for 1008 fs. The machine was an HP K-box.

Method	$\Delta t, \delta t$	p=2	4	Method speedup
Verlet/r-RESPA	4,1	1.7	3.2	1
<i>Equilibrium</i>	6,1	1.7	3.2	1.3
BBK-M	16,1	1.8	3.2	2.8
BBK-M	16,2	1.7	3.2	3.3

H group is a heavy atom and all hydrogens covalently bonded to it. A patch resides completely in one processor. *Equilibrium* and the B-spline methods parallelize well, since the averaging is performed on each H group independently of all other H groups (that is, H groups decouple for the averaging). Since all the information needed to do the averaging and mollification (computing $\mathcal{A}(x)$ and $\mathcal{A}_x(x)^T F^{\text{slow}}$) is local to the patch, there is no extra communication requirement. There is, however, an extra memory requirement needed to store the averaged positions. The filter matrices $\mathcal{A}_x(x)^T$ are not formed completely, but rather the matrix–vector product $\mathcal{A}_x(x)^T F^{\text{slow}}$ is computed from x , $\mathcal{A}(x)$. There is only one scalar μ_k per fastest interaction. An efficient computation of $\mathcal{A}_x(x)^T$ for *Equilibrium** is presented in [11]. It makes use of the special structure of this matrix.

5 Conclusions

This paper has shown that it is possible to stabilize symplectic integrators for molecular dynamics using very mild damping that is less than the decay rate of the velocity autocorrelation function. This was illustrated by combining a mollified impulse method, that extends the stability of the Verlet-I/r-RESPA impulse method, and a Langevin Impulse method, which is exact for constant forces. The resulting method, called LM, can take long time steps of 16 fs with a γ of 0.5 ps^{-1} . LM has been implemented and tested with biomolecules in explicit flexible solvent in the parallel molecular dynamics program NAMD 2.1. It is possible that combining even stabler MOLLY methods with mild damping, one may achieve time steps for molecular dynamics that are only limited by accuracy, rather than stability. An important remaining question is the effect of adding even very mild damping on the dynamics for biomolecular systems. Our study of spectral information suggests that adding mild damping does not distort significantly the dynamics. However, more extensive study and consideration of this question would more clearly specify the limitations and applicability of the methods presented here.

References

- [1] Helmut Grubmüller. Dynamiksimulation sehr großer makromoleküle auf einem parallelrechner. Master’s thesis, Physik-Dept. der Tech. Univ., Munich, Germany, 1989.
- [2] H. Grubmüller, H. Heller, A. Windemuth, and K. Schulten. Generalized Verlet algorithm for efficient molecular dynamics simulations with long-range interactions. *Molecular Simulation*, 6:121–142, 1991.
- [3] M. Tuckerman, B. J. Berne, and G. J. Martyna. Reversible multiple time scale molecular dynamics. *J. Chem. Phys.*, 97(3):1990–2001, 1992.
- [4] J. J. Biesiadecki and R. D. Skeel. Dangers of multiple-time-step methods. *J. Comput. Phys.*, 109(2):318–328, December 1993.
- [5] B. García-Archilla, J. M. Sanz-Serna, and Robert D. Skeel. Long-time-step methods for oscillatory differential equations. *SIAM J. Sci. Comput.*, 20(3):930–963, October 20, 1998.
- [6] Eric Barth and Tamar Schlick. Extrapolation versus impulse in multiple-timestepping schemes. II. Linear analysis and applications to Newtonian and Langevin dynamics. *J. Chem. Phys.*, 109(5):1633–1642, August 1, 1998.
- [7] T. Bishop, Robert D. Skeel, and K. Schulten. Difficulties with multiple timestepping and the fast multipole algorithm in molecular dynamics. *J. Comput. Chem.*, 18(14):1785–1791, November 15, 1997.
- [8] J.M. Sanz-Serna and M.P. Calvo. *Numerical Hamiltonian Problems*. Chapman and Hall, London, 1994.

- [9] Jesús Izaguirre, Sebastian Reich, and Robert D. Skeel. Longer time steps for molecular dynamics. *J. Chem. Phys.*, 110(19):9853–9864, May 15, 1999.
- [10] Robert D. Skeel and Jesús Izaguirre. The five femtosecond time step barrier. In P. Deuffhard, J. Hermans, B. Leimkuhler, A. Mark, S. Reich, and R. D. Skeel, editors, *Computational Molecular Dynamics: Challenges, Methods, Ideas*, volume 4 of *Lecture Notes in Computational Science and Engineering*, pages 303–318. Springer-Verlag, Berlin, November 1998.
- [11] Jesús A. Izaguirre. *Longer Time Steps for Molecular Dynamics*. PhD thesis, University of Illinois at Urbana-Champaign, 1999. Also UIUC Technical Report UIUCDCS-R-99-2107. Available online via <http://www.cs.uiuc.edu/research/tech-reports.html>.
- [12] Andrew R. Leach. *Molecular Modelling, Principles and Applications*. Addison Wesley Longman Limited, Essex, 1996.
- [13] Eric Barth and Tamar Schlick. Overcoming stability limitations in biomolecular dynamics. I. Combining force splitting via extrapolation with Langevin dynamics in LN. *J. Chem. Phys.*, 109(5):1617–1632, August 1, 1998.
- [14] Robert D. Skeel. Integration schemes for molecular dynamics and related applications. In M. Ainsworth, J. Levesley, and M. Marletta, editors, *The Graduate Student’s Guide to Numerical Analysis*, SSCM. Springer-Verlag, Berlin, 1999. To appear.
- [15] Walter Nadler, Axel Brünger, Klaus Schulten, and Martin Karplus. Molecular and stochastic dynamics of proteins. *Proc. Natl. Acad. Sci. USA*, 84:7933–7937, 1987.
- [16] M. P. Allen and D. J. Tildesley. *Computer Simulation of Liquids*. Clarendon Press, Oxford, New York, 1987. Reprinted in paperback in 1989 with corrections.
- [17] Peter E. Kloeden and Eckhard Platen. *Numerical Solution of Stochastic Differential Equations*, volume 23 of *Applications of Mathematics: Stochastic Modelling and Applied Probability*. Springer-Verlag, New York, 1992. Second corrected printing 1995.
- [18] R. W. Pastor, B. R. Brooks, and A. Szabo. An analysis of the accuracy of Langevin and molecular dynamics algorithm. *Mol. Phys.*, 65:1409–1419, 1988.
- [19] A. Brünger, C. B. Brooks, and M. Karplus. Stochastic boundary conditions for molecular dynamics simulations of ST2 water. *Chem. Phys. Lett.*, 105:495–500, 1982.
- [20] W. F. van Gunsteren and H. J. C. Berendsen. Algorithms for brownian dynamics. *Mol. Phys.*, 45(3):637–647, 1982.
- [21] A. Sandu and T. Schlick. Masking resonance artifacts in force-splitting methods for biomolecular simulations by extrapolative langevin dynamics. *J. Comput. Phys.*, 151(1):74–113, May 1, 1999.
- [22] Laxmikant Kalé, Robert Skeel, Milind Bhandarkar, Robert Brunner, Attila Gursoy, Neal Krawetz, James Phillips, Aritomo Shinozaki, Krishnan Varadarajan, and Klaus Schulten. NAMD2: Greater scalability for parallel molecular dynamics. *J. Comp. Phys.*, 151:283–312, 1999.

Appendix A Linear stability of *Equilibrium** MOLLY

We assume that U^{fast} and U^{slow} are given by

$$U^{\text{fast}} = \frac{1}{2}(x - x_0)^{\text{T}} A^{\text{fast}}(x - x_0), \quad (29)$$

$$U^{\text{slow}} = \frac{1}{2}x^{\text{T}} A^{\text{slow}}x + b^{\text{T}}x + \gamma, \quad (30)$$

where $U_{xx}^{\text{fast}} = A^{\text{fast}}$ is symmetric semi-positive definite and $A^{\text{fast}} + A^{\text{slow}}$ is symmetric positive definite (s.p.d.).

Note that if U^{fast} includes a linear term $a^{\text{fast}}x$ more general than $x_0 A^{\text{fast}}x$, the limit $\tau \rightarrow \infty$ of BE may not be defined, and *Equilibrium** would not either. The requirement on $A^{\text{fast}} + A^{\text{slow}}$ ensures that the analytical solution is bounded. Thus, our model problem is the most general suitable linear problem.

Next, we derive *Equilibrium** as the infinite τ limit of the Backward Euler method, which is given by

$$a = x - \tau^2 M^{-1} A^{\text{fast}} (a - x_0), \quad (31)$$

where $a = \mathcal{A}(x)$.

We require that the method be stable in the sense that the energy is bounded. The following theorem establishes that the solution and hence the energy is bounded.

Theorem 1. *Let the general linear problem $U^{\text{fast}} = \frac{1}{2}(x - x_0)^\top (A^{\text{fast}})(x - x_0)$, $U^{\text{slow}} = \frac{1}{2}x^\top A^{\text{slow}}x + b^\top x + \gamma$, with A^{fast} symmetric semi-positive definite, and $A^{\text{fast}} + A^{\text{slow}}$ s.p.d, be discretized using the Equilibrium MOLLY method. This numerical integrator is stable (the solution is bounded) if $\Delta t \rho(M^{-1/2} A^{\text{slow}} M^{-1/2})^{1/2} < 2$.*

Proof.

After applying a symplectic transformation of variables to Eq. (31) we get

$$\tilde{a} = (I + \Delta t^2 \Omega^2)^{-1} \tilde{x} + \tilde{x}_0. \quad (32)$$

where Ω^2 is semi-positive definite.

If we assume that the eigenvalues of Ω^2 are in non-increasing order, then we can rewrite this matrix as $\Omega^2 = \begin{bmatrix} \hat{\Omega}^2 & 0 \\ 0 & 0 \end{bmatrix}$ where the elements of $\hat{\Omega}$ are positive. Thus, in the limit $\tau \rightarrow \infty$ Eq. (32) becomes

$$\tilde{a} = \Phi \tilde{x} + \tilde{x}_0. \quad (33)$$

where

$$\Phi = \begin{bmatrix} 0 & \\ & \hat{I} \end{bmatrix}. \quad (34)$$

At this point we will drop the tildes from the transformed variables. The discretization of the problem becomes

half a mollified kick:

$$p^{n-1+\epsilon} = p^{n-1} - \frac{\Delta t}{2} \Phi L (\Phi x^{n-1} + x_0) - \frac{\Delta t}{2} \Phi b. \quad (35)$$

a vibration:

Propagate $p^{n-1+\epsilon}$, x^{n-1} by solving (36)

$$\begin{aligned} \frac{d}{dt} x = p, \quad \frac{d}{dt} p = -\Omega^2 x \\ \text{to get } p^{n-\epsilon}, x^n. \end{aligned}$$

half a mollified kick:

$$p^n = p^{n-\epsilon} - \frac{\Delta t}{2} \Phi L (\Phi x^n + x_0) - \frac{\Delta t}{2} \Phi b. \quad (37)$$

Now, let $y^n = \begin{bmatrix} (x^n)^\top & (p^n)^\top \end{bmatrix}^\top$. The numerical solution is given by $y^{n+1} = E y^n + e$, where E is a matrix and e is a vector given by

$$E = \begin{bmatrix} I & 0 \\ -\frac{\Delta t}{2} \Phi L \Phi & I \end{bmatrix} \begin{bmatrix} \cos(\Omega \Delta t) & \Omega^{-1} \sin(\Omega \Delta t) \\ -\Omega \sin(\Omega \Delta t) & \cos(\Omega \Delta t) \end{bmatrix} \begin{bmatrix} I & 0 \\ -\frac{\Delta t}{2} \Phi L \Phi & I \end{bmatrix}, \quad (38)$$

and

$$e = \begin{bmatrix} -\frac{\Delta t}{2} \Omega^{-1} \sin(\Omega \Delta t) (\Phi L x_0 + \Phi b) \\ -\frac{\Delta t}{2} (\cos(\Omega \Delta t) + I) (\Phi L x_0 + \Phi b) \end{bmatrix}. \quad (39)$$

Stability depends on the eigenvalues of E . To prove that the solution is bounded, we require that E be power bounded, which is the case if $\rho(E) \leq 1$. We also require that $(I + E + \dots + E^n)e$ be bounded itself. The proof proceeds in two steps: First, the power boundedness of E is proven; Second, it is shown that $(I + E + \dots + E^n)e$ is bounded.

There are two key observations in the remainder of this proof:

1. The separation of the solution into fast and slow parts produces two decoupled propagator matrices E for the fast and slow parts .
2. These matrices are rotations in phase space; if the stability condition obtained on the slow part is satisfied, then the solution is bounded. This is so because the vector e becomes zero for values of E which would make the solution unbounded.

The development of the above points is skipped for brevity. The reader is referred to [11, pp. 50-53].

Appendix B Generation of random variables for the Langevin Impulse integrator

The procedure for generating the random variables for Langevin Impulse, given in Appendix B of [14], is summarized here for reference. To integrate

$$I(f) = \int_0^T f(t) dW(t) \quad (40)$$

one can use the following facts:

- $I(f)$ is a Gaussian random variable,
- $E[I(f)] = 0$,
- $E[I(f)^2] = \int_0^T f(t)^2 dt$,
- $E[I(f)I(g)] = \int_0^T f(t)g(t)dt$.

Thus one can obtain $\int_0^T f(t) dW(t)$ from $(\int_0^T f(t)^2 dt)^{1/2} Z$, where Z is a Gaussian random number of mean 0 and variance 1. To generate other integrals $\int_0^T g(t) dW(t)$ we need to take the covariance of these random variables into account. Jointly Gaussian random variables Y_1, Y_2, \dots, Y_k with mean vector μ and covariance matrix C can be generated by the following formula

$$\vec{Y} = C_{1/2} \vec{Z} + \mu \quad (41)$$

where \vec{Z} is a vector of independent standard Gaussian random numbers of mean 0 and variance 1 and $C_{1/2}$ is a Cholesky factor satisfying $C_{1/2} C_{1/2}^T = C$. For our integrator the elements of C are the variances

$$c_{11} = \int_{t^n}^{t^{n+1}} \left(e^{-\gamma(t^{n+1}-t)} \right)^2 dt, \quad c_{22} = \int_{t^n}^{t^{n+1}} \left(1 - e^{-\gamma(t^{n+1}-t)} \right)^2 dt \quad (42)$$

and the covariance

$$c_{12} = c_{21} = \int_{t^n}^{t^{n+1}} e^{-\gamma(t^{n+1}-t)} \left(1 - e^{-\gamma(t^{n+1}-t)}\right) dt. \quad (43)$$

We choose $[R_1^{n+1}, R_2^{n+1}]^T = C_{1/2}[Z_1^{n+1}, Z_2^{n+1}]^T$ as in Eq. (41).

Given next are the analytical values for the integrals in Eqs. (42) and (43):

$$c_{11} = \frac{1}{2\gamma}(1 - e^{-2\gamma\Delta t}), \quad (44)$$

$$c_{22} = \Delta t - \frac{2}{\gamma}(1 - e^{-\gamma\Delta t}) + \frac{1}{2\gamma}(1 - e^{-2\gamma\Delta t}), \quad (45)$$

$$c_{12} = c_{21} = \frac{1}{\gamma}(1 - e^{-\gamma\Delta t}) - \frac{1}{2\gamma}(1 - e^{-2\gamma\Delta t}). \quad (46)$$

The Cholesky factor for the 2 by 2 covariance matrix is

$$C_{1/2} = \begin{bmatrix} c_{11}^{1/2} & 0 \\ c_{21}c_{11}^{-1/2} & (c_{22} - c_{21}^2c_{11}^{-1})^{1/2} \end{bmatrix}. \quad (47)$$

Let $\tau_1 = \frac{1}{\gamma}(1 - e^{-\gamma\Delta t})$, and $\tau_2 = \frac{1}{2\gamma}(1 - e^{-2\gamma\Delta t})$. Then the random numbers needed for the algorithm are

$$\begin{bmatrix} R_1^{n+1} \\ R_2^{n+1} \end{bmatrix} = \begin{bmatrix} \tau_2^{1/2} & 0 \\ (\tau_1 - \tau_2)\tau_2^{-1/2} & (\Delta t - \tau_1^2\tau_2^{-1})^{1/2} \end{bmatrix} \begin{bmatrix} Z_1^{n+1} \\ Z_2^{n+1} \end{bmatrix}. \quad (48)$$