Title

The divergence of nearby trajectories in soft-sphere DEM

Authors

Fullmer, William D

Porcu, Roberto

Musser, Jordan

et al.

Publication Date

2022-04-01

DOI

10.1016/j.partic.2021.06.008

Short communication

The divergence of nearby trajectories in soft-sphere DEM

William D. Fullmer[1,2*], Roberto Porcu[1,2], Jordan Musser[1], Ann S. Almgren[3], Ishan Srivastava[3]

[1]National Energy Technology Laboratory, Morgantown, WV 26507, USA

[2]NETL Support Contractor, Morgantown, WV 26507, USA

[3]Center for Computational Sciences and Engineering, Lawrence Berkeley National Laboratory, Berkeley, CA 94720, USA

*Corresponding author. E-mail: william.fullmer@netl.doe.gov

Abstract: The *n*-body instability is investigated with the soft-sphere discrete element method. The divergence of nearby trajectories is quantified by the dynamical memory time. Using the inverse proportionality between the dynamical memory time and the largest Lyapunov exponent, the soft-sphere discrete element method results are compared to previous hard-sphere molecular dynamics data for the first time. Good agreement is observed at low concentrations and the degree of instability is shown to increase asymptotically with increasing spring stiffness. At particle concentrations above 30%, the soft-sphere Lyapunov exponents increase faster than the corresponding hard-sphere data, hypothesized to be caused by relatively rare multi-particle collisions. This paper concludes with a demonstration of how this case study may be used in conjunction with regression testing and code verification activities.

Keywords: Soft-sphere DEM; Chaos; Lyapunov exponent

1. Introduction

Particulate gas–solid multiphase flows and the computational codes used for their numerical simulation are notoriously unstable. There are at least three types of instabilities that may be encountered: i) *n*-body instability of the particles (Alligood, Sauer, & Yorke, 1996); ii) turbulence in the fluid (Manneville, 2010); and iii) inherently multiphase instabilities, e.g., bubbling and clustering (Fullmer & Hrenya 2017). In this work, we focus on the *n*-body instability, which is chaotic when three or more discrete entities, or bodies, interact with one another (Poincaré, 1890).

The *n*-body problem is pervasive in physical modeling, spanning scales from stellar dynamics (Goodman, Heggie, & Hut,1993) to molecular dynamics (Norman & Stegailov, 2013). First reported in astrophysical simulations (Miller, 1964), the *n*-body instability exhibits a sensitive dependence on initial conditions which causes infinitesimally close trajectories to diverge from one another exponentially (Goodman, Heggie, & Hut 2003; Norman & Stegailov, 2013). The divergence of nearby trajectories presents a practical challenge to numerical solution (Boekholt & Zwart, 2015) and particularly to code development when seemingly trivial changes to the source code, e.g., order of operations, produce non-trivial changes in the solutions of benchmark problems.

Molecular dynamics (MD) or discrete particle simulation techniques can be separated into (at least) three categories. Historically, the first numerical methods applied a hard-sphere approach (Alder & Wainwright, 1957) where particle interactions are assumed to be instantaneous. Efficient computational strategies for hard-sphere MD stream through time from collision to collision (Pöschel & Schwager, 2005). Conversely, later MD simulations employed "long" range forces, e.g., the Lennard-Jones pair-potential (Rahman, 1964), requiring a time-marching scheme to resolve the continuous forces between particles. Finally, soft-sphere discrete element method (DEM) approaches arose in the granular dynamics community (Cundall & Strack, 1979) to model enduring and multi-particle collisions which cannot be resolved by either of the previous MD methods. Due to similar requirements in modeling dense fluidized beds, the soft-sphere technique is by far the most common method employed in coupled computational fluid dynamics (CFD)-DEM modeling of particle fluidization.

The chaotic dynamics, specifically the rate of divergence of nearby trajectories, has been studied in both hard-sphere MD (Dellago, Posch, & Hoover, 1996; Dellago & Posch, 1997) and long-range potential MD (Norman & Stegailov, 2002; 2013), in addition to stellar dynamics (Heggie, 1991). However, to the best of the authors' knowledge, the same problem has not received attention for the soft-sphere DEM method. The purpose of this study is to apply techniques previously established for other MD methods (Dellago & Posch, 1997; Norman & Stegailov, 2002) to quantify the rate of divergence present in the soft-sphere DEM technique.

The remainder of this work is organized as follows. The soft-sphere DEM model is described in section "Model and method". The specific system being simulated in this work is described in section "Problem setup". The results are provided in section "Results", where we begin by defining key criteria and end with an example of a practical application. We conclude with a summary of the key findings in section "Conclusions".

2. Model and method

This work uses MFIX-Exa, a new CFD-DEM code being developed by the National Energy Technology Laboratory and Lawrence Berkeley National Laboratory as part of the U.S. Department of Energy's Exascale Computing Project. MFIX-Exa originated by combining the DEM modules of the classic MFIX code (mfix.netl.doe.gov) with a modern low Mach number projection method for the continuous fluid phase. The new algorithm is implemented using the AMReX (amrex-codes.github.io) software framework for massively parallel block-structured applications (Zhang et al., 2021).

A vast majority of the MFIX-Exa code base is not exercised in this work. Specifically, by focusing on the *n*-body instability, the fluid phase is not solved at all. The remaining particle phase is modeled by solving Newton's equations of motion for every particle. Contacts are resolved with a soft-sphere model, specifically the linear spring dashpot (LSD) model (Cundall & Strack, 1979; Garg, et al., 2012). The particles are assumed to be frictionless, $\mu_{\mathrm{pp}} = 0$, and elastic, $e_{\mathrm{pp}} = 1$. Eliminating all sources of dissipation ensures that granular instabilities will not develop, which would strongly affect the nonlinear dynamics of the system (McNamara & Mareschal, 2001). The frictionless assumption removes particle angular momentum as a dependent variable (reducing the degrees of freedom per particle from nine to six) and reduces the contact forces to only normal contributions. The latter assumption eliminates the damping term, i.e., the dashpot coefficient $\eta_{\mathrm{pp}} = 0$ (Garg et al., 2012). Consequently, the originally complex model reduces to simply,

$$\dot{\boldsymbol{x}}_{\mathrm{i}} = \boldsymbol{u}_{\mathrm{i}}\,,\;\; m_{\mathrm{i}}\dot{\boldsymbol{u}}_{\mathrm{i}} = \sum_{\mathrm{j}=1}^{N_{\mathrm{p}}} k_{\mathrm{n}}\delta_{\mathrm{n}}^{(\mathrm{ij})}\boldsymbol{n}_{\mathrm{ij}}, \tag{1}$$

where m_{i}, $\boldsymbol{x}_{\mathrm{i}}$, and $\boldsymbol{u}_{\mathrm{i}}$ are the mass, position, and (linear) velocity vectors of the i^{th} particle of N_{p} total particles, k_{n} is the normal spring constant,

$$\boldsymbol{n}_{\mathrm{ij}} = \frac{\boldsymbol{x}_{\mathrm{j}} - \boldsymbol{x}_{\mathrm{i}}}{|\boldsymbol{x}_{\mathrm{j}} - \boldsymbol{x}_{\mathrm{i}}|}\,, \tag{2}$$

is the normal vector pointing from the i^{th} to the j^{th} particle, and

$$\delta_{\mathrm{n}}^{(\mathrm{ij})} = \max\left[0, \tfrac{1}{2}\left(d_{\mathrm{j}} + d_{\mathrm{i}}\right) - \left|\boldsymbol{x}_{\mathrm{j}} - \boldsymbol{x}_{\mathrm{i}}\right|\right], \tag{3}$$

is the strictly positive overlap between the i^{th} and j^{th} particle when $i \neq j$. In this work, the particles are assumed to be monodispersed so the diameter, d, and mass, m, are constants, along with k_{n}.

The governing equations of Eqs. (1)–(3) are solved with a simple first-order Euler time integration method, as typically found in general purpose CFD-DEM codes. Given the incredibly small system sizes studied here, all of the high-performance computing aspects of the MFIX-Exa code (Musser et al. 2021) reduced to a single grid, serial computation.

3. Problem setup

Ultimately, the results of this work will be compared to the hard-sphere Lyapunov exponents of Dellago and Posch (1997), therefore, their original problem setup is replicated here. The particles are ordered in a face-centered cubic lattice with an edge length of three unit cells, $s = 3$, totaling $N_\mathrm{p} = 4s^3 = 108$ particles. The lattice length, L, is determined from the particle concentration, ϕ, given by $L^* = L/d = (\pi N_\mathrm{p}/6\phi)^{1/3}$. Periodic boundary conditions are applied in all directions. Each component of the particle velocity is taken from a random normal distribution which is then renormalized to have a zero-mean velocity in each direction and a specified thermal speed,

$$T = \frac{1}{3}\frac{1}{N_\mathrm{p}}\sum_{\mathrm{i}=1}^{N_\mathrm{p}} \boldsymbol{u}_\mathrm{i}^2 \ . \tag{4}$$

where T is the (massless) granular temperature which is proportional to the fluctuating kinetic energy. Three sample lattices are provided in Fig. 1 with particles colored by their velocity magnitude. MFIX-Exa is a dimensional code and values of $d = 0.1$ m and $T = 2/3$ m^2/s^2 are prescribed. The particle density is set to give unit mass, $m = 1$ kg.

Fig. 1

The spring constant of the LSD collision model must also be specified. Because the results will be compared to hard-sphere MD, the collisions should appear nearly instantaneous. We assume a priori that this means the LSD collision time,

$$\delta t_\mathrm{coll} = \pi\sqrt{\frac{m}{2k_\mathrm{n}}}\,, \tag{5}$$

i.e., the duration of a collision, should be much less than the kinetic collision time,

$$\tau_\mathrm{coll} = \frac{d}{24\phi\chi(\phi)}\sqrt{\frac{\pi}{T}}\,, \tag{6}$$

i.e., the mean time between successive collisions. The Carnahan and Starling (1969) expression for the radial distribution function at contact, $\chi(\phi) = (1-\phi/2)/(1-\phi)^3$, is applied in Eq. (6).

The original hard-sphere study was carried out at a very high density. However, there is a discontinuity in the dynamical properties of the system at $\phi \approx 0.497$, corresponding to a critical point where the particles can no longer escape the lattice structure. We will avoid this region of limited particle mobility in this work by only considering a moderate range of particle concentrations: $0.02 \leq \phi \leq 0.40$. Therefore, the minimum (kinetic) collision time is $\tau_{\text{coll}} \approx 6.1\times10^{-3}$ s. We set $\delta t_{\text{coll}} = 1\times10^{-4}$ s, so that $\tau_{\text{coll}}/\delta t_{\text{coll}} > 60$. From Eq. (5), this gives a spring constant of $k_{\text{n}} \approx 4.9348\times10^{8}$ N/m. It remains to be seen whether or not this is sufficiently stiff.

4. Results

4.1. Dynamical memory time

In this section we study the divergence of a slightly perturbed DEM simulation from a reference DEM simulation. The perturbed positions, $\boldsymbol{x}_{\text{i}}'$, and velocities, $\boldsymbol{u}_{\text{i}}'$, are denoted by a prime, and the square of their separation is given by

$$\delta_{\text{x}} = \frac{1}{N_{\text{p}}}\sum_{i=1}^{N_{\text{p}}}(\boldsymbol{x}_{\text{i}} - \boldsymbol{x}_{\text{i}}')^2 \,, \tag{7}$$

and

$$\delta_{\text{u}} = \frac{1}{N_{\text{p}}}\sum_{i=1}^{N_{\text{p}}}(\boldsymbol{u}_{\text{i}} - \boldsymbol{u}_{\text{i}}')^2 \,. \tag{8}$$

The separation of the perturbed solution from the reference solution grows similarly in space and velocity, and both signals are heavily polluted with noise, see Fig. 2, albeit for different reasons. Particles in a collision may have very similar spatial coordinates with velocities of opposing sign. Conversely, particles near a periodic boundary may have very similar velocities but very different coordinates. Of the two, we find working with the separation of the velocity field slightly easier.

Fig. 2

To study the divergence of trajectories, a perturbation must be introduced relative to the reference solution. Here, we follow Norman and Stegailov (2002) who use an ingenious method of simply choosing a smaller time step in the perturbed solution. The smaller (than reference) time step introduces a roundoff level-error which is converted to (and mixed with) a truncation-level error through the numerical method. Further, an initial perturbation amplitude does not need to be explicitly prescribed which allows the reference and perturbed states separate naturally from zero. The time step is set to one twentieth of the LSD collision time, $dt = \delta t_{\text{coll}}/20 = 5\times10^{-6}$ s, in all reference simulations. Fig. 2 shows the divergence of four perturbed simulations from the reference simulation using increasingly smaller time steps dt'. The separation of all trajectories grows very similarly from zero and saturates at $2c^2$ where $c = \sqrt{3T}$ is the thermal speed, i.e., the most probable speed from the Maxwellian velocity distribution. Hence, the separation is nondimensionalized by $\delta_{\text{u}}^* = \delta_{\text{u}}/6T$. Time is nondimensionalized by $t^* = t\sqrt{3T/2d^2}$.

The time it takes for the separation to reach saturation is the dynamical memory time, t_{m}^{*} (Norman & Stegailov, 2002). The dynamical memory time identifies the time it takes the DEM system to "forget" the initial conditions, i.e., numerical solutions at times $t > t_{\mathrm{m}}^{*}$ no longer represent an exact Newtonian trajectory. It should be noted that t_{m}^{*} is specific to this initial condition, LSD model, spring constant, integration method and reference time step. Technically, t_{m}^{*} is achieved in the limit of $dt'/dt \rightarrow 0$. However, our results indicate that t_{m}^{*} is fairly insensitive to changes in dt'/dt from 1/2 to 1/20, and $dt'/dt = 1/10$ is used throughout the remainder of this study.

The noise of the signal due to collisions is smoothed (to some degree, compare Fig. 7 to Fig. 2) by ensemble averaging over the separations. The same spatial lattice is used in each case, but the random particle velocity is redrawn. The dynamical memory time is approximated by the time it takes the separation to reach 50% of saturation, i.e., the first occurrence of $\delta_{\mathrm{u}}^{*} \geq 0.5$. Using this definition, the dynamical memory time is marked with a circle in Fig. 2 for a single trajectory at concentration $\phi = 0.40$. The particle concentration is varied from $\phi = 0.40$ down to 0.02 by $\Delta\phi = 0.02$. The largest Lyapunov exponent, λ_1, determines the divergence rate and is inversely proportional to the dynamical memory time, $\lambda_1^{*} \propto 1/t_{\mathrm{m}}^{*}$. The maximal Lyapunov exponent can be approximated from the dynamical memory time by,

$$\lambda_1^{*} \approx \frac{1}{t_{\mathrm{m}}^{*}} \ln \frac{6T}{Cm}, \tag{9}$$

where C is a proportionality constant determined by the integration method and the time step size (Norman & Stegailov, 2002). By comparing our results to the hard-sphere MD data of Dellago and Posch (1997), we find generally good agreement with the approximation $\lambda_1^{*} = 11/t_{\mathrm{m}}^{*}$ ($C = 4e^{-11}$) as shown in Fig. 3. The best agreement is observed in the dilute regime. Beginning around $\phi \approx 0.3$ the observed trend diverges from the reported hard-sphere data, rising at a faster rate with increasing ϕ. The increasing discrepancy at $\phi > 0.3$ is investigated further in the following sections.

Fig. 3

4.2. Spring stiffness

The most obvious culprit for differences between a hard-sphere MD and soft-sphere DEM comparison is the spring stiffness, k_{n}. In Sec. 3, we set $\tau_{\mathrm{coll}}/\delta t_{\mathrm{coll}} > 60$ for the densest case, $\phi = 0.40$, and had hoped that it would be sufficient for the collisions to appear instantaneous compared to the mean collision time. However, we note that this goes against our physical intuition of the situation: one would expect a softer collision to reduce the instability, i.e., to lower λ_1, not to increase it.

To check this assumption, we would like to increase k_n at ϕ = 0.40 and observe how it affects λ_1^*. However, increasing k_n would require decreasing dt (and dt') so that the collisions are resolved by a reasonable number of time steps (we prefer at least 20). This would change the proportionality constant, C, and the fit $\lambda_1^* = 11/t_\mathrm{m}^*$ would no longer hold. Therefore, we instead test the spring stiffness by taking the case ϕ = 0.20, which shows good agreement, and softening the stiffness of the LSD collision model while maintaining dt and dt'. This means that the collisions are resolved by an increasing number of time steps as k_n decreases, but, we believe, the fit $\lambda_1^* = 11/t_\mathrm{m}^*$ should still be consistent. The spring constant is reduced by six orders of magnitude, increasing δt_coll three orders of magnitude, up to 0.1 s. For reference, the kinetic collision time for ϕ = 0.20 is τ_coll = 0.0257 s, so the lowest two values of k_n spend more time in collisions than between collisions. The maximal Lyapunov exponent decreases with decreasing k_n as shown in Fig. 4. The trend observed in Fig. 4 is consistent with our physical picture of the problem: Eq. (1) becomes increasingly stiff in a mathematical sense as the LSD spring constant is increased, increasing the degree of the n-body instability eventually saturating at a level consistent with hard-sphere MD. It is interesting to note that similar observations have been reported for softened gravitational potentials in stellar dynamics, e.g., see Fig. 8 of Goodman, Heggie, and Hut (1993). This provides some evidence that the selected k_n for this system is sufficiently stiff to approximate the hard-sphere limit. Furthermore, because the Lyapunov exponent decreases with decreasing k_n, the over-prediction of λ_1^* at higher ϕ must not be caused by overly soft particles.

Fig. 4

4.3. Multi-particle contacts

Another consequence of the finite collision time in soft-sphere DEM is the existence of multi-particle collisions, i.e., a particle in simultaneous contact with more than one other neighboring particles. In fact, multi-particle and enduring contact is the reason why soft-sphere DEM is utilized so extensively in fluidization applications, as these interactions become increasingly common in dense beds. At the concentration and stiffness of the present condition, multi-particle contacts were assumed not to occur. To evaluate this assumption, the reference simulations were re-run, and every particle was checked at every time step for the occurrence of more than one colliding neighbor, i.e., a particle for which $\delta_\mathrm{n}^{(\mathrm{ij})}$ of Eq. (3) is greater than zero for at least two j-particles. Any DEM time step in which at least one particle has at least two colliding neighbors is flagged as a multi-particle collision step and normalized by the total number of time steps for that simulation. The ensemble-averaged likelihood of a multi-particle collision, L_mpc, is collected in Fig. 5, which shows a small but increasingly non-trivial occurrence with increasing concentration. The likelihood increases proportionally to the (kinetic) collision frequency to a power of approximately two over most of the ϕ range studied here. The dashed trendline in the inset of Fig. 5 shows $2.5 \times 10^{-7}\ \tau_{coll}^{2.11}$. It is easy to imagine how three-particle collisions are significantly more unstable than typical binary collisions.

Although the likelihood of occurrence of a multi-particle collision is at most slightly over 1%, it seems plausible that such complex interactions could be the cause for the deviation of the DEM results from the hard-sphere MD data.

Fig. 5

The previous analyses in Fig. 4 and 5 obscure a noteworthy aspect: L_{mpc} also depends on k_{n}. It is expected that L_{mpc} should vanish as $k_{\mathrm{n}} \rightarrow \infty$. Therefore, not only are L_{mpc} and k_{n} related, but they have an opposing influence on the stability of the system. To better understand this behavior, the divergence of trajectories is studied at the highest concentration, $\phi = 0.4$, with increasing the spring stiffness above the previously selected $k_{\mathrm{n}} = 4.9348\times10^{8}$ N/m. To study this region, the time step must be reduced. Here, we set $dt = \delta t_{\mathrm{coll}}/20$ where δt_{coll} depends on k_{n} as in Eq. (5) which is increased several orders of magnitude. The resulting λ_1^* can not be compared with the same generality as before (e.g., as in Fig. 6 where k_{n} was reduced with dt fixed), however one test of $\phi = 0.2$ with an elevated spring constant suggests that the $\lambda_1^* = 11/t_{\mathrm{m}}^*$ may hold reasonably well. To denote this minor difference, the results are reported as $11/t_{\mathrm{m}}^*$ instead of λ_1^* in Fig. 6. For each spring stiffness, the likelihood of multi-particle collisions is also collected and provided in the inset of Fig. 6. At low k_{n}, the results of Fig. 6 are similar to those in Fig. 4: increasing k_{n} increases the mathematical stiffness of the problem leading to an increase in λ_1^*. For the softest particles studied, L_{mpc} is nearly one, i.e., multi-particle collisions are almost always occurring. As k_{n} increases further, L_{mpc} starts to decay and multi-particle collisions start to become infrequent while λ_1^* appears to plateau. Somewhat unexpectedly, however, as k_{n} increases further beyond the originally selected value, λ_1^* decreases slightly before asymptoting to a value still slightly higher than the hard sphere result of Dellago and Posch (1997), which is shown as a dotted blue line in Fig. 6 for reference. This local maximum in λ_1^* suggests that the influence of k_{n} on λ_1^* is a shifting balance. For very soft-particle systems, the mathematical stiffness represented by k_{n} is more important (i.e., increasing k_{n} increases λ_1^*) while for nearly-hard-sphere systems, the dependence of L_{mpc} on k_{n} becomes more important (i.e., increasing k_{n} decreases λ_1^*).

Fig. 6

While this investigation has revealed an interesting and complex relationship between k_{n} and λ_1^*, it has also demonstrated that multi-particle collisions must not be responsible for the discrepancy with the hard-sphere data at high ϕ. At the highest k_{n} studied, the system has reached effective zero L_{mpc} (i.e., is more likely that a multi-particle collision will not occur within t_{m}^* for any given simulation) and yet the approximate λ_1^* is still noticeably larger than the true hard-sphere result. In an effort to exhaust all options, the effect of the initial perturbation method and the influence of the selected numerical code are further studied in Appendix 1 and 2, respectively. Both additional investigations show nearly identical results to those presented in Fig. 3. At present, the ultimate case of the difference between soft-sphere and hard-sphere Lyapunov exponents for $\phi >$ 0.3 remains unknown.

4.4. Application

This study was originally undertaken to benchmark the degree of n-body instability in a developmental soft-sphere (CFD-)DEM code, MFIX-Exa. This allows us to evaluate whether the divergence of solutions due to changes in the source code, domain decomposition or computation method is within the expected limits of the n-body problem (as opposed to indicating a programming error). Having established, at least for $\phi < 0.30$, good agreement with hard-sphere MD data (Dellago & Posch, 2002), we can now use this divergence rate as a metric to measure perturbations arising from unintended or unexpected sources.

Three different sources of implicit perturbations are studied: i) changing the architecture and compiler, ii) explicitly changing the order of operations in the code, and iii) introducing an error in the numerical method. With these three implicit perturbation methods, only the reference inputs are simulated. Specifically, the reference time step, *dt*, is used throughout and divergence from the "true" reference solutions arise purely numerically. To investigate the first perturbation above, we compared reference results as run on NETL's Joule 2 supercomputer using the Intel Xeon Gold 6148 processors against the NVIDIA Tesla P100 accelerators. Both codes are built with the GNU gcc compiler release 9.3 and the GPU accelerated code additionally used CUDA nvcc release 11.0 as the AMReX GPU backend (Zhang et al., 2021). It was anticipated that a different GNU compiler version and/or a different level of compilation optimization might order floating point operations differently and yield a perturbed solution (Dietiker, 2012). However, to our surprise, this did not happen (at least for $t^{*} < 1$) using several different GNU compiler versions nor an unoptimized (debug) build. To investigate changes in the order of operations, we explicitly reordered the three components of the particle force vector. Specifically, $k_{\mathrm{n}}\delta_{\mathrm{n}}^{(\mathrm{ij})}\boldsymbol{n}_{\mathrm{ij}}$ in Eq. (1) was replaced by $\boldsymbol{n}_{\mathrm{ij}}k_{\mathrm{n}}\delta_{\mathrm{n}}^{(\mathrm{ij})}$. Lastly, to test the third type of perturbation listed above, we introduced a deliberate error in the source code by omitting collisions of the 1st particle.

The divergence of the ensemble-averaged velocity field separation for the three perturbations discussed above are compared to the explicit perturbation of using a smaller time step, $dt' = dt/10$, in Fig. 7 at concentration $\phi = 0.20$. All four perturbations diverge from the reference solution in a similar manner, separated in time due to different (unspecified) values of initial perturbation amplitude, *C*. Recall that the dynamical memory time, t_{m}^{*}, determines the time it takes the system, model and numerical method to lose dependence on initial condition. This test shows it takes longer than the dynamical memory time for trajectories due to a different processor and a different order of operations to diverge from the reference trajectory (i.e., the red and green lines are to the right of the black reference line in Fig. 7). Therefore, we can say that the divergence is beyond what can be expected of this system, contact model and numerical method due to the interaction of truncation level error (dt') with the n-body instability. In other words, this divergence is expected. Conversely, when an intentional error is introduced to the code, the trajectory diverges from reference within the dynamical memory time (i.e., the blue line is to the left of the black reference line). In this case we can say that something has caused a larger than truncation level disturbance to this system, model and method. Such a test can be a useful accompaniment to standard regression testing, in which small changes are the norm for short simulations and little beyond expert judgement can be used to say if the level of divergence is expected or not. In this case, the bug may have been easy to spot. Fig. 7 shows that the granular temperature of the system begins to exceed the initial value as mechanical energy is erroneously added to the system from excessive overlap due to the missed collision bug. Finally, we note that the separation from a smaller time step is virtually indistinguishable when both reference and perturbation simulations are carried out on the same processor (i.e., in this case either CPU or GPU). Therefore, if a coding error is more nuanced than the heavy-handed example bug and only occurs for specific processors or on specific architectures, it can still be identified by comparison to the expected truncation-level separation (i.e., from dt') for the processor, architecture, etc. of interest.

Fig. 7

6. Conclusions

Regression testing and verification activities can be challenging when developing codes to model complex multi-physics phenomena due to the intrinsically unstable and dynamic nature of these types of problems. In this work, the developmental CFD-DEM code MFIX-Exa is used as a prototype. The fluid phase is removed to isolate instability to the many-body or n-body problem by simulating a fully periodic granular (no interstitial fluid) system of 108 elastic, frictionless particles. The particles are originally arranged in a face-centered cubic lattice with a thermal (Maxwellian) velocity distribution. As the system evolves in time, the reference trajectory (velocity field) diverges from a nearby trajectory: the same system is solved with a smaller time step, $dt' = dt/10$, which naturally introduces a truncation and roundoff level perturbation. The dynamical memory time was computed for a range of particle concentrations from 2% to 40% (below the phase transition of the lattice) and compared to hard-sphere molecular dynamics (MD) data of the largest Lyapunov exponent (Dellago and Posch, 1997). The agreement between the (stiff) soft-sphere DEM and the hard-sphere MD data is good below approximately 30% particle concentration.

The spring constant, k_n, is the most obvious cause of the difference between the two methods; as one would expect soft-sphere should approach hard-sphere as $k_n \rightarrow \infty$, perhaps the guideline ($\tau_{coll} >> \delta t_{coll}$) to select k_n *a priori* was not sufficiently stiff. Yet, it was found that divergence rates increase with k_n and begin to plateau near the selected value. However, further investigation revealed a more complex relationship between k_n and the divergence rate, λ_1^*. For soft particles, λ_1^* increases with k_n due to an increase in the mathematical stiffness of the system. For hard particles, λ_1^* decreases with k_n due to a decreased likelihood of multiparticle collisions, which are more unstable than binary collisions. However, even when k_n is increased to levels where multiparticle collisions become effectively eliminated, a discrepancy still exists between soft-sphere DEM at the highest simulated particle stiffness and the hard-sphere MD data at the highest concentrations. For very soft particles, the mathematical stiffness represented by k_n dominates the nonlinear instability (i.e., increasing k_n increases λ_1^*) while for very stiff particles, the dependence of L_mpc on k_n becomes more important (i.e., increasing k_n decreases λ_1^*). In addition to hard- and soft-sphere modeling differences, the impact of how the initial perturbation is introduced and the effect of the code itself were also investigated as possible causes for this discrepancy. The results, summarized in Appendix 1 and 2, respectively, show that neither the choice of perturbation method nor choice of code cause such discrepancies.

Finally, the original motivation for this work is demonstrated by studying perturbations that are not explicit like dt'. The system is simulated using different processor architectures (CPU vs. GPU), different order of operations (the three components of the normal force are rearranged), and a deliberate error (collision forces of the first particle is ignored). By comparing the separation of these solutions (from the reference solution) to the separation of the dt' perturbation, we observe that the different processor and order of operations diverge slower than the dynamical memory time. Hence, these differences are smaller than truncation level error (dt vs. dt') and the divergence is a natural and expected consequence of noise being amplified by the *n*-body problem. On the other hand, when a true error is introduced in the form of a bug in the code, the trajectory diverges from the reference within the dynamical memory time. This can provide a warning that something has caused a larger than truncation error level change. We hope to use this simple test to distinguish between expected instabilities and errors during ongoing development of the MFX-Exa and other soft-sphere DEM codes.

Appendix 1

In this work, we used the method of Norman and Stegailov (2002) to calculate the dynamical memory time, t_m^*, by solving a perturbed trajectory with a smaller time step than the reference trajectory. For this system, we found that $dt' = dt/10$ was sufficient, see Fig. 2. This method has the benefit that the amplitude of perturbation does not need to be explicitly prescribed; it arises naturally due to truncation error. However, there was a minor concern with this method that the perturbation is not just to the initial condition. In fact, it does not perturb the initial condition at all. Rather, the perturbation is introduced continuously with every time integration. To verify that the dt' perturbation method was not responsible differences with the hard sphere data (Dellago and Posch, 1997), the dynamical memory time was computed using a more traditional perturbation method. Here, the x-position of the first particle is shifted at time zero by a small amount: $x_1{}'(t = 0) = x_1(t = 0) + (\varepsilon, 0, 0)$. The initial perturbation amplitude is set to $\varepsilon/d = 1\times10^{-9}$. The dynamical memory time is approximated in the same fashion, i.e., as the first occurrence of 50% saturation of the separation of the velocity field. The results are inverted into an approximate Lyapunov exponent and compared to the original results (from dt' perturbation) and the hard-sphere data Fig. A1. Although the proportionality constant has changed (due to a different initial perturbation amplitude) the results are virtually identical aside from some expected noise.

Fig. A1

Appendix 2

As a final check, we repeat the study again using a different soft-sphere DEM code. For this purpose, we used the GRANULAR package in Sandia National Laboratories' opensource, general-purpose MD code LAMMPS (https://lammps.sandia.gov) (Plimpton, 1995), specifically the 3 March 2020 version (LAMMPS, 2020). As before, 108 particles are initialized with a kinetic temperature in a lattice of 3×3 face-centered cubic unit cells. The same dimensional properties of system are applied: $d = 0.1$ m, $m = 1$ kg, $T = 2/3$ m^2/s^2 and $k_\mathrm{n} \approx 4.9348\times10^8$ N/m. The tangential spring constant is set to zero along with no viscous damping between contacting particles. The same time step, $dt = 5\times10^{-6}$ s, is also specified, however a second-order velocity-Verlet method is used for time integration in LAMMPS. The code and associated packages are built with GNU compiler version 9.3 and the simulations were run on Joule2's Intel Xeon Gold 6148 processors. Ensembles of 30 reference trajectories are simulated for each concentration ranging from $\phi = 0.02$ to 0.40. Perturbed trajectories are computed with $dt' = dt/10$. The dynamical memory time is defined as the time it takes the ensemble-averaged separation of the velocity field, Eq. (8), to reach a normalized value of 0.5. The largest Lyapunov exponent is approximated by $\lambda_1^* = 11/t_\mathrm{m}^*$ and compared to the MFIX-Exa results and the reference hard-sphere data in Fig. A2. The two soft-sphere results are very similar—both showing a divergence from the hard-sphere data around $\phi = 0.20$. The result verifies that the discrepancy is not due to the MFIX-Exa soft-sphere implementation and its associated numerical method. Rather, Fig. A2 helps confirm that the difference in the observed Lyapunov exponent (and dynamical memory time) is due to differences inherent in the hard-sphere and soft-sphere models themselves. As discussed in Sec.4.3, we believe the key difference to be the occurrence of multi-particle collisions in the soft-sphere DEM simulations, which are not possible in hard-sphere MD. This hypothesis could warrant further study but is beyond the scope of this work.

Fig. A2

Acknowledgments

The authors would like to thank Andrew Myers for insightful discussion. This research was supported by the Exascale Computing Project (17-SC-20-SC), a collaborative effort of the U.S. Department of Energy Office of Science and the National Nuclear Security Administration.

Disclaimer

This project was funded by the United States Department of Energy, National Energy Technology Laboratory, in part, through a site support contract. Neither the United States Government nor any agency thereof, nor any of their employees, nor the support contractor, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

References

Alder, B. J., & Wainwright, T. E. (1957). Phase transition for a hard sphere system. *The Journal of Chemical Physics*, *27*(5), 1208-1209.

Allen, M. P. & Tildesley, D. J. (2017). *Computer simulation of liquids*. Oxford University Press, Oxford, UK.

Alligood, K. T., Sauer, T. D., & Yorke, J. A. (1996) *Chaos - An Introduction to Dynamical Systems*. Springer-Verlag, New York, USA.

Boekholt, T., & Zwart, S. P. (2015). On the reliability of *N*-body simulations. *Computational Astrophysics and Cosmology*, *2*(1), 1-21.

Carnahan, N. F., & Starling, K. E. (1969). Equation of state for nonattracting rigid spheres. *The Journal of Chemical Physics*, *51*(2), 635-636.

Cocco, R., Fullmer, W.D., Liu, P., & Hrenya, C. M. (2017). CFD-DEM: Modeling the small to understand the large. *Chemical Engineering Progress*, *113*(9), 38-45.

Cundall, P.A., & Strack, O.D. (1979). A discrete numerical model for granular assemblies. *Geotechnique*, *29*, 47–65.

Dellago, C., Posch, H. A., & Hoover, W. G. (1996). Lyapunov instability in a system of hard disks in equilibrium and nonequilibrium steady states. *Physical Review E*, *53*(2), 1485.

Dellago, C., & Posch, H. A. (1997). Kolmogorov-Sinai entropy and Lyapunov spectra of a hard-sphere gas. *Physica A: Statistical Mechanics and its Applications*, *240*(1-2), 68-83.

Dietiker, J.-F., (2012) MFIX results sensitivity of Fortran compilers, National Energy Technology Laboratory Technical Report. URL: https://mfix.netl.doe.gov/doc/mfix-archive/mfix_current_documentation/MFIX_results_sensitivity_to_Fortran_compilers.pdf

Fullmer, W.D., & Hrenya, C.M. (2017). The clustering instability in rapid granular and gas-solid flows. *Annual Review of Fluid Mechanics*, *49*, 485–510

Garg R., Galvin J., Li, T., & Pannala S. (2012) Open-source MFIX-DEM software for gas-solids flows: Part I-Verification studies. *Powder Technology, 220,* 122–137.

Goodman, J., Heggie, D. C., & Hut, P. (1993). On the exponential instability of *N*-body systems. *The Astrophysical Journal*, *415*, 715.

Heggie, D. C. (1991). Chaos in the *N*-body problem of stellar dynamics. In *Predictability, Stability, and Chaos in* N-*Body Dynamical Systems* (pp. 47-62). Springer, Boston, MA.

LAMMPS, (2020) Stable Release 3 March 2020, https://dx.doi.org/10.5281/zenodo.3726417

Manneville, P. (2010). *Instabilities, Chaos and Turbulence: An Introduction to Nonlinear Dynamics and Complex Systems*. (Vol. 1). World Scientific.

McNamara, S. and Mareschal, M. (2001). Lyapunov spectrum of granular gases. *Physical Review E*, *64*, 061306.

Miller, R. H. (1964). Irreversibility in Small Stellar Dynamical Systems. *The Astrophysical Journal*, *140*, 250.

Musser, J., Almgren, A. S., Fullmer, W. D., Antepara, O., Bell, J. B., Blaschke, Gott, K., Myers, A., Porcu, R., Rangarajan, D., Rosso, M., Zhang, W., & Syamlal, M. (2021). MFIX-Exa: A Path Towards Exascale CFD-DEM Simulations. *The International Journal of High-Performance Computing Applications,* submitted.

Norman, G. E., & Stegailov, V. V. (2002). Stochastic and dynamic properties of molecular dynamics systems: Simple liquids, plasma and electrolytes, polymers. *Computer Physics Communications*, *147*(1-2), 678-683.

Norman, G. E., & Stegailov, V. V. (2013). Stochastic theory of the classical molecular dynamics method. *Mathematical Models and Computer Simulations*, *5*(4), 305-333.

Plimpton, S. (1995) Fast parallel algorithms for short-range molecular dynamics. *Journal of Computational Physics*. *117*(1), 1–19.

Pöschel, T., & Schwager, T. (2005). Computational granular dynamics: models and algorithms. Springer Science & Business Media.

Poincaré, H. (1890). *Sur le probleme des trois corps et les equations de la dynamique*. Stockholm: F. & G. Beijer.

Rahman, A. (1964). Correlations in the motion of atoms in liquid argon. *Physical Review*, *136*(2A), A405.

Zhang, W., Myers, A., Gott, K., Almgren, A. & Bell, J. (2021). AMReX: Block-Structured Adaptive Mesh Refinement for Multiphysics Applications. *International Journal of High Performance Computing Applications*, submitted.

The divergence of nearby trajectories in soft-sphere DEM

William D. Fullmer[1,2*], Roberto Porcu[1,2], Jordan Musser[1], Ann S. Almgren[3], Ishan Srivastava[3]

Figure captions:

Fig. 1. Representative face-centered cubic lattices used as the initial condition for particle concentrations of (a) $\phi = 0.1$, (b) $\phi = 0.2$, and (c) $\phi = 0.4$. Particles colored by normalized velocity magnitude, $|\boldsymbol{u}_{\mathrm{i}}|/2c$, ranging from zero (black) to unity (white).

Fig. 2. Divergence of a single reference trajectory from four perturbed trajectories with increasingly smaller time steps measured by the separation of the velocity field. The circle indicates the dynamical memory time, $t_{\mathrm{m}}^{*} \approx 1$, for a particle concentration of $\phi = 0.40$.

Fig. 3. Comparison of the largest Lyapunov exponent between hard-sphere MD reported by Dellago and Posch (1997) and the soft-sphere DEM results of the present study, approximated by the inverse dynamical memory time.

Fig. 4. Dependence of the maximal Lyapunov exponent on the LSD spring constant, k_{n}.

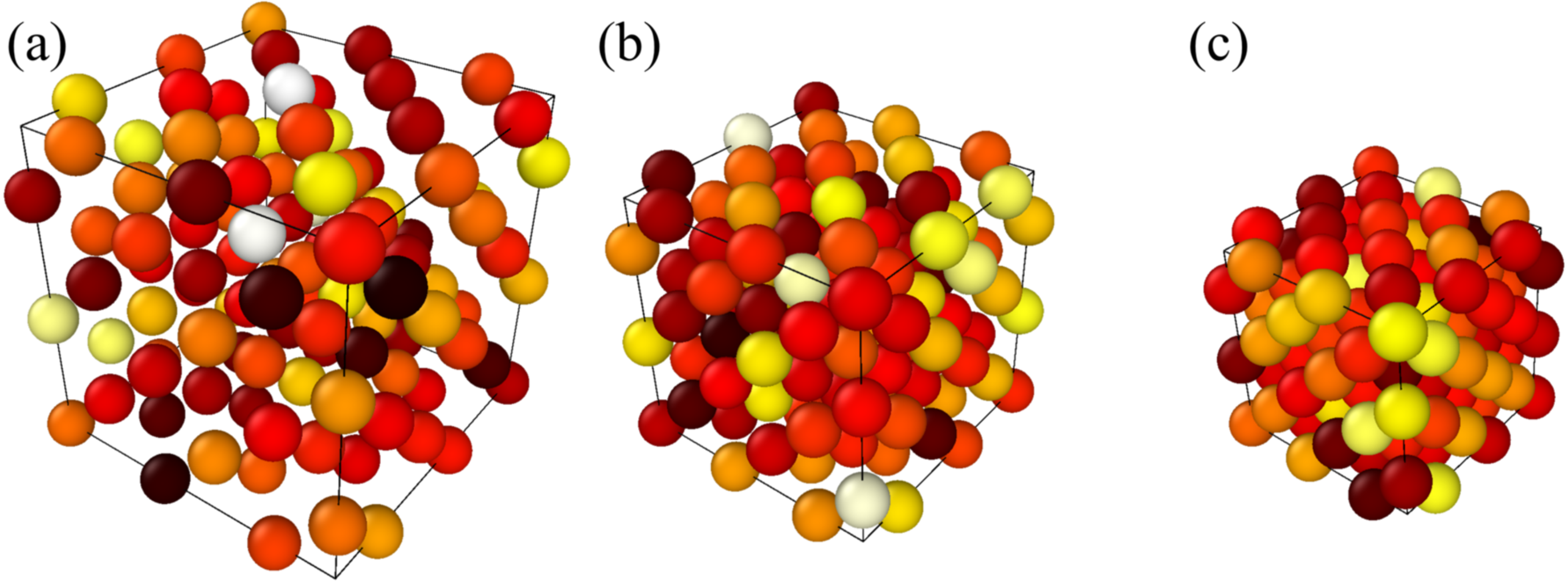
Fig. 5. The likelihood of finding at least one particle with more than one colliding neighbor in a given time step as a function of concentration. Inset gives the same data but as a function of collision frequency.

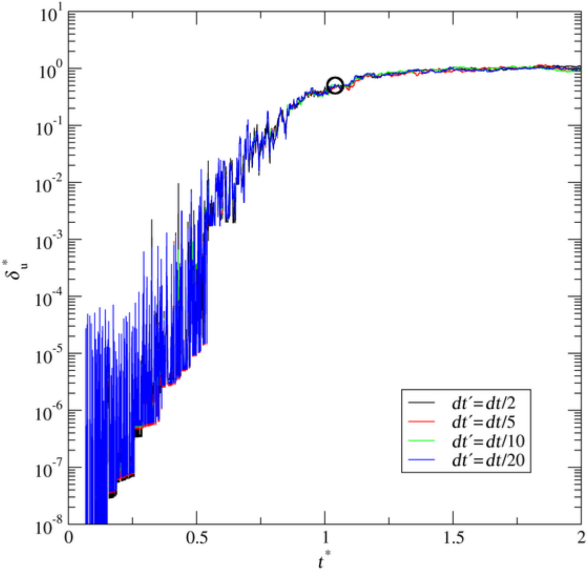
Fig. 6. Effect of the spring constant on the dynamical memory time (shown as approximate λ_1^*) with variable dt = $\delta t_{\mathrm{coll}}/20$. The dashed blue line gives the corresponding hard-sphere data for reference. Inset shows the corresponding likelihood of multiparticle collisions as a function of the spring constant.

Fig. 7. Ensemble average of 30 perturbed trajectories diverging from corresponding reference trajectories for ϕ = 0.20 with perturbations induced by a different time step, a different computation method (GPU), a different order of operations and an intentional coding mistake.

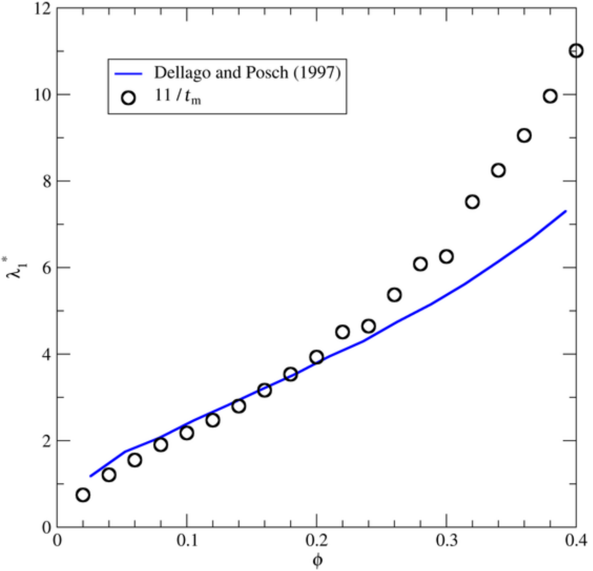
Fig. A1. Impact of different perturbation methods (smaller time step vs. initial spatial disturbance) on the approximate largest Lyapunov exponent compared to the corresponding hard-sphere MD data (Dellago & Posch, 1997) for reference.

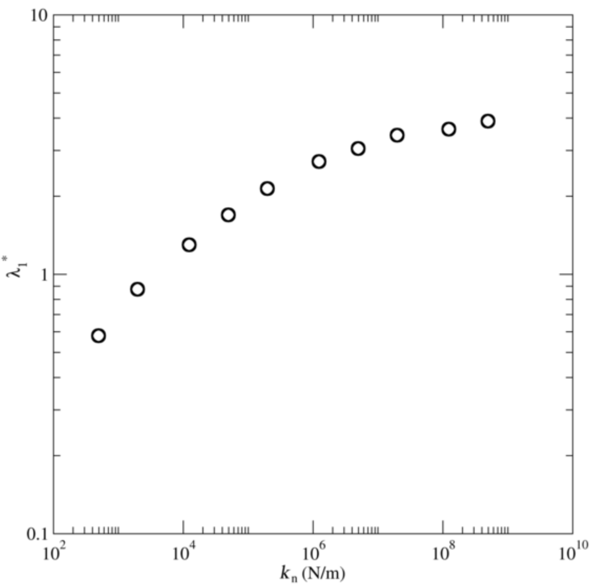
Fig. A2. Approximate largest Lyapunov exponent computed with LAMMPS compared to the MFIX-Exa results and the corresponding hard-sphere MD data (Dellago & Posch, 1997) for reference.

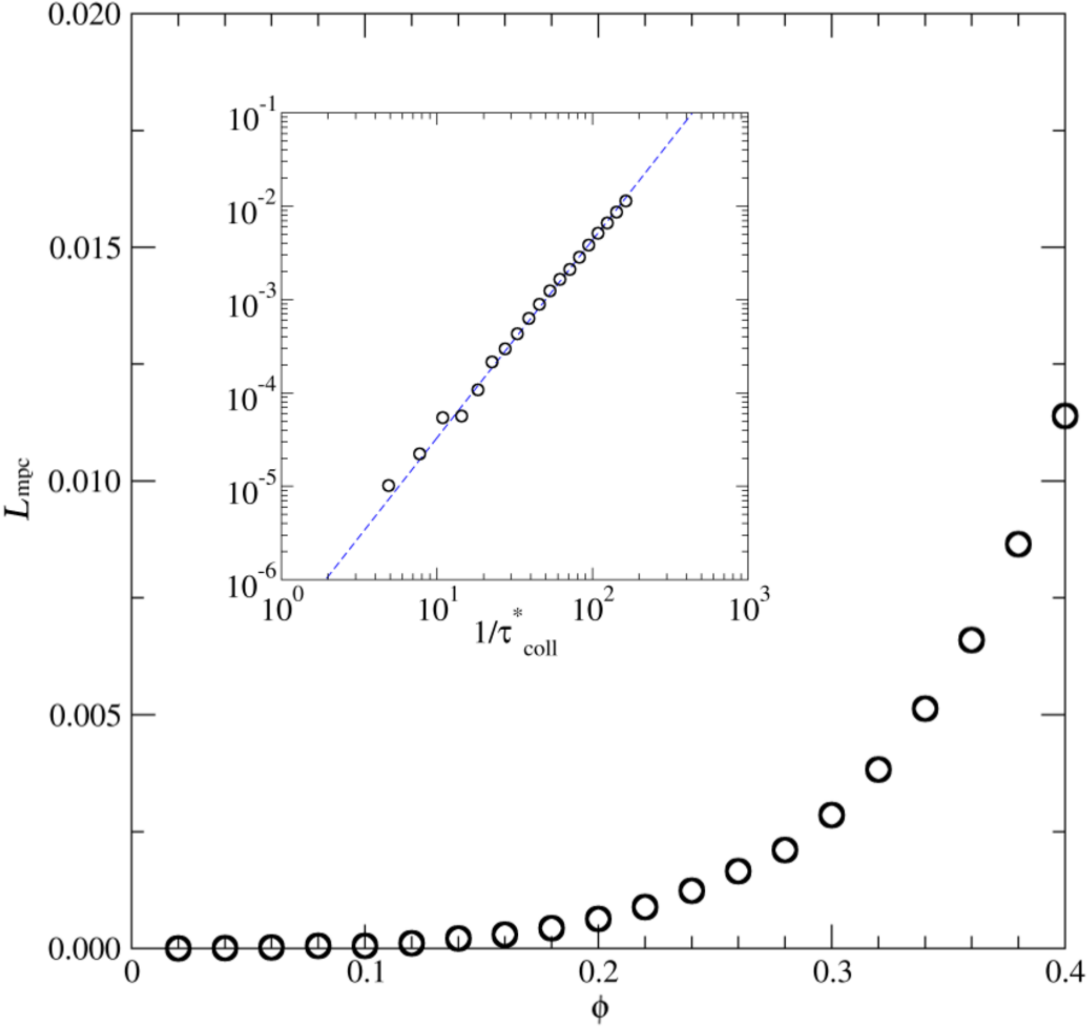

(a)
(b)
(c)

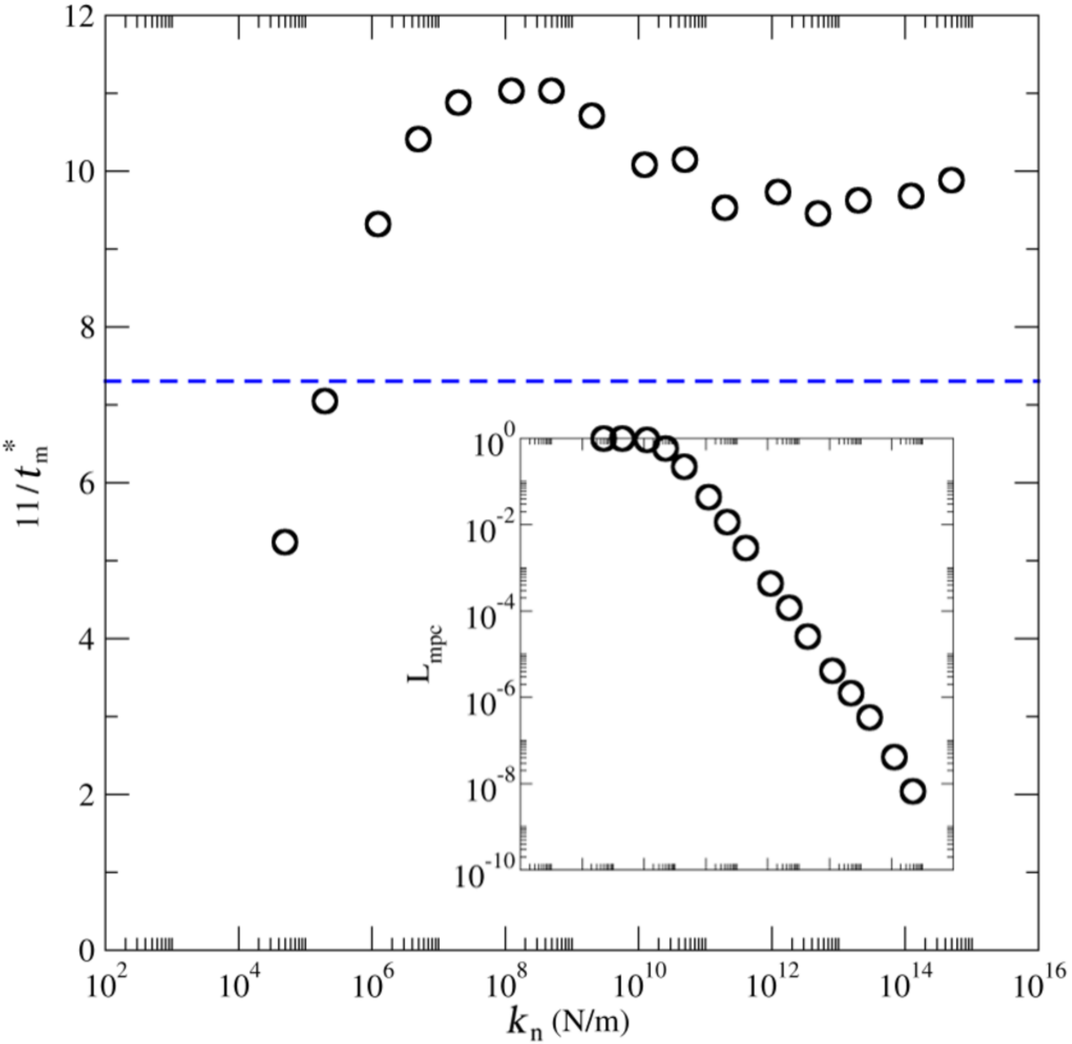


10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{-8}
δ_u^*
0
0.5
1
1.5
2
t^*
$dt'=dt/2$
$dt'=dt/5$
$dt'=dt/10$
$dt'=dt/20$

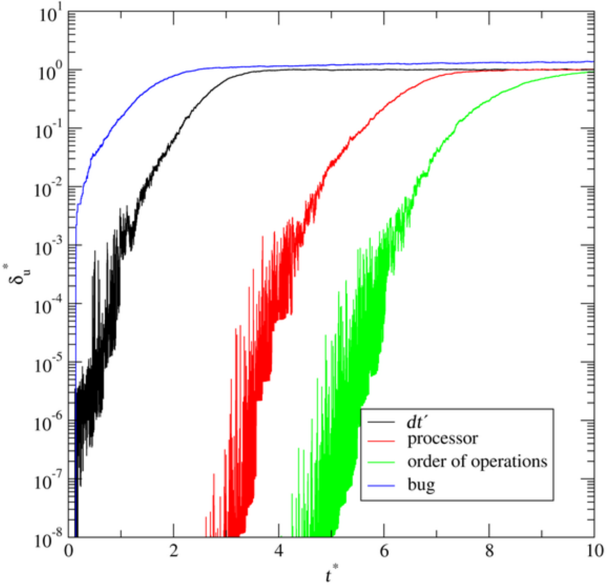

Dellago and Posch (1997)
○ 11 / t_m
λ_1^*
ϕ
0
2
4
6
8
10
12
0
0.1
0.2
0.3
0.4

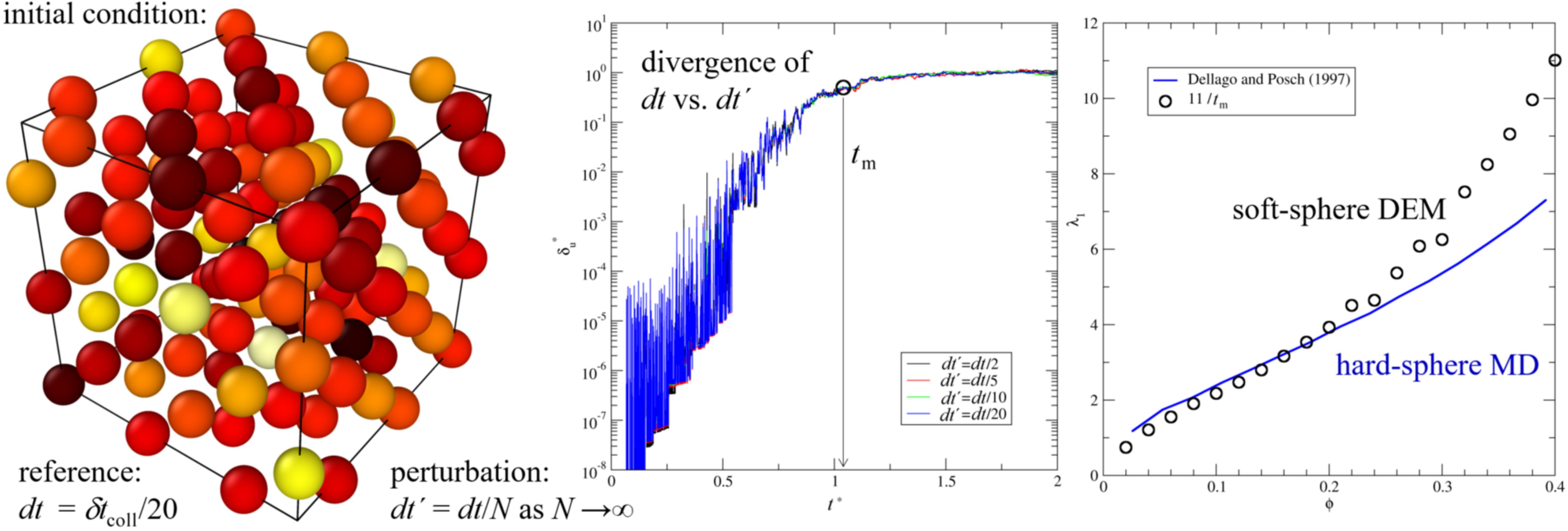

10
1
0.1
λ_1^*
10^2
10^4
10^6
10^8
10^{10}
k_n (N/m)


0.020
0.015
0.010
0.005
0.000
L_{mpc}
0
0.1
0.2
0.3
0.4
ϕ
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{0}
10^{1}
10^{2}
10^{3}
$1/\tau^{*}_{\mathrm{coll}}$



12
10
8
6
4
2
0
$11/t_m^*$
10^2
10^4
10^6
10^8
10^{10}
10^{12}
10^{14}
10^{16}
k_n (N/m)
10^0
10^{-2}
10^{-4}
10^{-6}
10^{-8}
10^{-10}
L_{mpc}


10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{-8}
δ_u^*
0
2
4
6
8
10
t^*
dt'
processor
order of operations
bug


initial condition:
reference:
$dt = \delta t_{\text{coll}}/20$
perturbation:
$dt' = dt/N$ as $N \rightarrow \infty$
divergence of
dt vs. dt'
t_m
10^{1}
10^{0}
10^{-1}
10^{-2}
10^{-3}
10^{-4}
10^{-5}
10^{-6}
10^{-7}
10^{-8}
δ_u^*
0
0.5
1
1.5
2
t^*
$dt'=dt/2$
$dt'=dt/5$
$dt'=dt/10$
$dt'=dt/20$
Dellago and Posch (1997)
$11/t_m$
soft-sphere DEM
hard-sphere MD
0
2
4
6
8
10
12
λ_1
0
0.1
0.2
0.3
0.4
ϕ