Stability analysis of explicit MPM

Contents

1 Schemes of interest **1**

2 Von Neumann analysis of MPM **2**

2.1 Definitions . . . 3
2.2 Particle to grid transfers . . . 4
2.3 Grid forces . . . 4
2.4 Velocities . . . 4
2.5 Affine state . . . 5
2.6 Deformation gradient . . . 6
2.7 Solving for the vectors and matrices . . . 6
2.8 Eliminating the scalars . . . 7
2.9 Real, complex, and bounds . . . 8
2.10 Stability . . . 8
2.11 Isotropic . . . 9

3 Single particle instability **9**

3.1 Stability formulation . . . 10
3.2 Transfers to grid and grid update . . . 10
3.3 Grid to particle transfers . . . 11
3.4 Simplifying the system . . . 12
3.5 Solving the eigenvalue problem . . . 13
3.6 Case $S(\lambda; s, r_i, r_j) = 0$ . . . 13
3.7 Case $Q(\lambda; s, p, r_i) = 0$, $\lambda = \pm 1$ . . . 13
3.8 Case $Q(\lambda; s, p, r_i) = 0$, $|\lambda| = 1$, λ complex . . . 14
3.9 Analysis for 2D simulation . . . 14
3.10 Analysis for PIC 3D . . . 15
3.11 Analysis for PIC 2D . . . 16
3.12 Summary and time step restriction . . . 17
3.13 Effects of variable time step sizes . . . 17
3.14 Conclusions about varying time step sizes . . . 19

1 Schemes of interest

We begin by fixing a typical explicit MPM integration scheme based on APIC transfers. From the APIC version it is relatively straightforward to obtain the PIC and CPIC versions. All of the analysis that we perform assume this scheme (or its PIC and CPIC analogues). Portions that should be omitted for PIC are colored.

$$m_i^n = \sum_p w_{ip}^n m_p \tag{1}$$

$$m_i^n \mathbf{u}_i^n = \sum_p w_{ip}^n m_p (\mathbf{u}_p^n + \mathbf{C}_p^n (\mathbf{x}_i^n - \mathbf{x}_p^n)) \tag{2}$$

$$\mathbf{f}_i^n = -\sum_p V_p^0 (\mathbf{P}(\mathbf{F}_p^n))(\mathbf{F}_p^n)^T \nabla w_{ip}^n \tag{3}$$

$$\tilde{\mathbf{u}}_i^{n+1} = \mathbf{u}_i^n + \Delta t (m_i^n)^{-1} \mathbf{f}_i^n \tag{4}$$

$$\mathbf{u}_p^{n+1} = \sum_i w_{ip}^n \tilde{\mathbf{u}}_i^{n+1} \tag{5}$$

$$\mathbf{x}_p^{n+1} = \mathbf{x}_p^n + \Delta t \mathbf{u}_p^{n+1} \tag{6}$$

$$\nabla \mathbf{u}_p^{n+1} = \sum_i \tilde{\mathbf{u}}_i^{n+1} (\nabla w_{ip}^n)^T \tag{7}$$

$$\mathbf{F}_p^{n+1} = (\mathbf{I} + \Delta t \nabla \mathbf{u}_p^{n+1}) \mathbf{F}_p^n \tag{8}$$

$$\mathbf{C}_p^{n+1} = \xi \sum_i w_{ip}^n \tilde{\mathbf{u}}_i^{n+1} (\mathbf{x}_i^n - \mathbf{x}_p^n)^T \tag{9}$$

Here, we have assumed that quadratic or cubic splines are used for the transfer weights. We note the following properties of the weights:

$$\sum_i w_{ip}^n = 1 \tag{10}$$

$$\sum_i w_{ip}^n (\mathbf{x}_i^n - \mathbf{x}_p^n) = 0 \tag{11}$$

$$\sum_i \nabla w_{ip}^n = 0 \tag{12}$$

$$\sum_i (\mathbf{x}_i^n - \mathbf{x}_p^n)(\nabla w_{ip}^n)^T = \mathbf{I} \tag{13}$$

$$\sum_i w_{ip}^n (\mathbf{x}_i^n - \mathbf{x}_p^n)(\mathbf{x}_i^n - \mathbf{x}_p^n)^T = \frac{1}{\xi}\mathbf{I} \tag{14}$$

$$\xi = \begin{cases} \frac{4}{\Delta x^2} & \text{quadratic splines} \\ \frac{3}{\Delta x^2} & \text{cubic splines} \end{cases} \tag{15}$$

Here, (12) and (13) are obtained by taking the gradient of the first two equations.

2 Von Neumann analysis of MPM

In this section, we perform Von Neumann analysis on the material point method. In order to apply Von Neumann analysis, we must assume a (quite restrictive) setup. (1) All cells contain the same number of particles in the same arrangement. (2) Periodic boundary conditions are used. (3) All particles are identical; in particular, they have the same mass ($m_p = m$) and volume ($V_p^0 = V^0$). (4) The configuration is near the rest equilibrium state, so that $\mathbf{F}_p^n \approx \mathbf{I}$, $\mathbf{u}_p^n \approx \mathbf{0}$, and positions can be considered to be stationary. (5) The particle distribution is symmetrical in the sense that for each grid node i and particles p there is a particle $\overline{p}$ such that $(\mathbf{x}_{\overline{p}}^n - \mathbf{x}_i^n) = -(\mathbf{x}_p^n - \mathbf{x}_i^n)$. Stated another way, if the grid is mirrored along the x, y, and z directions, there will be particles in the same locations. This assumption will simplify parts of the analysis by making certain quantities real-valued.

We will use index p to refer to a particle. Since each cell has the same particle layout, it makes sense to use the index q to refer to a particle within some *canonical* cell. This is useful for quantities that are not constant across particles but are the same for corresponding particles in different grid cells. Each p has a unique q associated with it, and each q corresponds with a p for each grid cell. Note that $N_G = \sum_i 1$, $N_p = \sum_p 1$, and $N_q = \sum_q 1$ are the number of cells, number of particles, and number of particles per cell respectively, so that $N_p = N_G N_q$.

Following the usual process of Von Neumann analysis, we fix a wave number $\mathbf{z}$ and look for solutions that look like waves.

$$\mathbf{u}_p^n = \epsilon g^n \mathbf{u}_q e^{i\mathbf{x}_p^n \cdot \mathbf{z}} \qquad \mathbf{C}_p^n = \epsilon g^n \mathbf{C}_q e^{i\mathbf{x}_p^n \cdot \mathbf{z}} \qquad \mathbf{F}_p^n = \mathbf{I} + \epsilon g^n \mathbf{F}_q e^{i\mathbf{x}_p^n \cdot \mathbf{z}} \tag{16}$$

Here g is the growth factor, with g^n being a power. ϵ is a very small number, so small that higher powers of it may be neglected. In this way, the analysis is perturbative. Note that these state variables are not assumed constant across all particles; they are the same from cell to cell but vary for the particles within a cell, thus the index on $\mathbf{u}_q$. Due to the very small velocities, we also assume that the particles do not move appreciably.

2.1 Definitions

Before we start propagating the approximations above through the time integration scheme, we introduce all of the intermediate quantities that we will use during this derivation.

$$h_q = \sum_i w_{ip}^n e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot\mathbf{z}} \qquad \mathbf{k}_q = \sum_i \nabla w_{ip}^n e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot\mathbf{z}} \qquad \mathbf{e}_q = \xi \sum_i w_{ip}^n (\mathbf{x}_i^n - \mathbf{x}_p^n) e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot\mathbf{z}}$$

$$\mathbf{c} = \sum_p \mathbf{K}_q \nabla w_{ip}^n e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot\mathbf{z}} = \sum_p \mathbf{K}_q \mathbf{k}_q \qquad \mathbf{B} = \sum_p \nabla w_{ip}^n \overline{\mathbf{k}_q}^T e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot\mathbf{z}} = \sum_q \mathbf{k}_q \overline{\mathbf{k}_q}^T$$

$$b = \sum_q (\overline{h_q} h_q + \xi^{-1} \overline{\mathbf{e}_q}^T \mathbf{e}_q) \qquad \mathbf{b} = \sum_p w_{ip}^n (\mathbf{u}_q + \mathbf{C}_q (\mathbf{x}_i^n - \mathbf{x}_p^n)) e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot\mathbf{z}} = \sum_q (h_q \mathbf{u}_q + \xi^{-1} \mathbf{C}_q \mathbf{e}_q)$$

$$w = \sum_p w_{ip}^n = N_g \qquad \mathbf{M} = \frac{\partial \mathbf{P}}{\partial \mathbf{F}}(\mathbf{I}) \qquad \mathbf{K}_q = \mathbf{M} : \mathbf{F}_q \qquad A_{ik} = M_{ijkl} B_{jl} \qquad \tau = \frac{\Delta t g}{g - 1}$$

$$\mathbf{u} = \frac{\mathbf{b} m - \Delta t V^0 \mathbf{c}}{w m g} \qquad s = \frac{b}{w} \qquad \beta = \frac{\Delta t V^0}{m(b - w g)} \qquad c = \frac{\Delta t^2 V^0}{m w} \qquad r = \frac{c\sigma - s - 1}{2}$$

Lowercase non-bold quantities (h_q, b, w, τ, s, β, c, and r) are scalars. Lowercase bold quantities ($\mathbf{k}_q$, $\mathbf{e}_q$, $\mathbf{c}$, $\mathbf{b}$, and $\mathbf{u}$) are vectors. Uppercase bold quantities are matrices ($\mathbf{B}$, $\mathbf{K}_q$, $\mathbf{A}$) or higher order tensors ($\mathbf{M}$). Note that we drop the bold when using indexing notation, so that M_{ijkl} refers to a component of the tensor $\mathbf{M}$. Quantities with a q index (h_q, $\mathbf{k}_q$, $\mathbf{e}_q$, and $\mathbf{K}_q$) are associated with canonical particles. They live on particles and vary from particle to particle within a cell, but they are the same from cell to cell.

2.2 Particle to grid transfers

We begin with the particle to grid transfers.

$$m_i^n = \sum_p w_{ip}^n m_p \tag{17}$$

$$m_i^n = wm \tag{18}$$

$$m_i^n \mathbf{u}_i^n = \sum_p w_{ip}^n m_p (\mathbf{u}_p^n + \mathbf{C}_p^n (\mathbf{x}_i^n - \mathbf{x}_p^n)) \tag{19}$$

$$wm\mathbf{u}_i^n = \sum_p w_{ip}^n m (\epsilon g^n \mathbf{u}_q e^{i\mathbf{x}_p^n \cdot \mathbf{z}} + \epsilon g^n \mathbf{C}_q e^{i\mathbf{x}_p^n \cdot \mathbf{z}} (\mathbf{x}_i^n - \mathbf{x}_p^n)) \tag{20}$$

$$w\mathbf{u}_i^n = \epsilon g^n e^{i\mathbf{x}_i^n \cdot \mathbf{z}} \sum_p w_{ip}^n (\mathbf{u}_q + \mathbf{C}_q (\mathbf{x}_i^n - \mathbf{x}_p^n)) e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot \mathbf{z}} \tag{21}$$

$$\mathbf{u}_i^n = \frac{\epsilon g^n \mathbf{b}}{w} e^{i\mathbf{x}_i^n \cdot \mathbf{z}} \tag{22}$$

Note that $\mathbf{b}$ effectively encodes the direction of the grid velocity separately from changes in time and space and that the form of the grid velocity follows the same wave form as the state variables.

2.3 Grid forces

We next consider forces. We compute forces through the first Piola-Kirchhoff stress tensor $\mathbf{P}$. Neglecting higher order terms in ϵ, the stress takes the form

$$\mathbf{F}_p^n = \mathbf{I} + \epsilon g^n \mathbf{F}_q e^{i\mathbf{x}_p^n \cdot \mathbf{z}} \tag{23}$$

$$\mathbf{P}(\mathbf{F}_p^n) = \mathbf{P}(\mathbf{I}) + \mathbf{M} : (\mathbf{F}_p^n - \mathbf{I}) \tag{24}$$

$$= \epsilon g^n e^{i\mathbf{x}_p^n \cdot \mathbf{z}} (\mathbf{M} : \mathbf{F}_q) \tag{25}$$

$$= \epsilon g^n e^{i\mathbf{x}_p^n \cdot \mathbf{z}} \mathbf{K}_q. \tag{26}$$

Next, we can evaluate forces, discarding ϵ^2 terms.

$$\mathbf{f}_i^n = -\sum_p V_p^0 (\mathbf{P}(\mathbf{F}_p^n))(\mathbf{F}_p^n)^T \nabla w_{ip}^n \tag{27}$$

$$= -\sum_p V_p^0 \epsilon g^n e^{i\mathbf{x}_p^n \cdot \mathbf{z}} \mathbf{K}_q (\mathbf{I} + \epsilon g^n \mathbf{F}_q e^{i\mathbf{x}_p^n \cdot \mathbf{z}})^T \nabla w_{ip}^n \tag{28}$$

$$= -\epsilon V^0 g^n \sum_p e^{i\mathbf{x}_p^n \cdot \mathbf{z}} \mathbf{K}_q \nabla w_{ip}^n = -\epsilon V^0 g^n e^{i\mathbf{x}_i^n \cdot \mathbf{z}} \mathbf{c}. \tag{29}$$

2.4 Velocities

Next, we apply forces to the grid velocities.

$$\tilde{\mathbf{u}}_i^{n+1} = \mathbf{u}_i^n + \Delta t (m_i^n)^{-1} \mathbf{f}_i^n \tag{30}$$

$$= \frac{\epsilon \mathbf{b}}{w} g^n e^{i\mathbf{x}_i^n \cdot \mathbf{z}} - \Delta t (wm)^{-1} \epsilon V^0 g^n e^{i\mathbf{x}_i^n \cdot \mathbf{z}} \mathbf{c} \tag{31}$$

$$= \frac{\epsilon g^n}{wm} e^{i\mathbf{x}_i^n \cdot \mathbf{z}} (\mathbf{b}m - \Delta t V^0 \mathbf{c}) \tag{32}$$

$$= \epsilon g^{n+1} e^{i\mathbf{x}_i^n \cdot \mathbf{z}} \mathbf{u} \tag{33}$$

We can then transfer these velocities to particles.

$$\begin{aligned}
\mathbf{u}_p^{n+1} &= \sum_i w_{ip}^n \tilde{\mathbf{u}}_i^{n+1} && (34)\\
&= \sum_i w_{ip}^n \epsilon g^{n+1} e^{i\mathbf{x}_i^n\cdot\mathbf{z}}\mathbf{u} && (35)\\
&= \epsilon g^{n+1} e^{i\mathbf{x}_p^n\cdot\mathbf{z}}\mathbf{u}\sum_i w_{ip}^n e^{i(\mathbf{x}_i^n-\mathbf{x}_p^n)\cdot\mathbf{z}} && (36)\\
&= \epsilon g^{n+1}\overline{h_q} e^{i\mathbf{x}_p^n\cdot\mathbf{z}}\mathbf{u} && (37)
\end{aligned}$$

From our original assumption we have $\mathbf{u}_p^{n+1} = \epsilon g^{n+1}\mathbf{u}_q e^{i\mathbf{x}_p^n\cdot\mathbf{z}}$, so that

$$\begin{aligned}
\epsilon g^{n+1}\mathbf{u}_q e^{i\mathbf{x}_p^n\cdot\mathbf{z}} &= \epsilon g^{n+1}\overline{h_q} e^{i\mathbf{x}_p^n\cdot\mathbf{z}}\mathbf{u} && (38)\\
\mathbf{u}_q &= \overline{h_q}\mathbf{u} && (39)
\end{aligned}$$

This tells us that $\mathbf{u}_q$ must all be aligned; we did not originally assume that. They may however vary in scale.

2.5 Affine state

Next, we transfer velocity derivatives to particles to compute $\mathbf{C}_p^{n+1}$.

$$\begin{aligned}
\mathbf{C}_p^{n+1} &= \xi\sum_i w_{ip}^n \tilde{\mathbf{u}}_i^{n+1}(\mathbf{x}_i^n-\mathbf{x}_p^n)^T && (40)\\
&= \xi\sum_i w_{ip}^n \epsilon g^{n+1} e^{i\mathbf{x}_i^n\cdot\mathbf{z}}\mathbf{u}(\mathbf{x}_i^n-\mathbf{x}_p^n)^T && (41)\\
&= \epsilon g^{n+1} e^{i\mathbf{x}_p^n\cdot\mathbf{z}}\mathbf{u}\xi\sum_i w_{ip}^n e^{i(\mathbf{x}_i^n-\mathbf{x}_p^n)\cdot\mathbf{z}}(\mathbf{x}_i^n-\mathbf{x}_p^n)^T && (42)\\
&= \epsilon g^{n+1} e^{i\mathbf{x}_p^n\cdot\mathbf{z}}\mathbf{u}\overline{\mathbf{e}_q}^T && (43)
\end{aligned}$$

As with velocities, this must match our definition for $\mathbf{C}_p^n$.

$$\begin{aligned}
\epsilon g^{n+1}\mathbf{C}_q e^{i\mathbf{x}_p^n\cdot\mathbf{z}} &= \epsilon g^{n+1} e^{i\mathbf{x}_p^n\cdot\mathbf{z}}\mathbf{u}\overline{\mathbf{e}_q}^T && (44)\\
\mathbf{C}_q &= \mathbf{u}\overline{\mathbf{e}_q}^T. && (45)
\end{aligned}$$

With this, we have learned about the form of $\mathbf{u}_q$ and $\mathbf{C}_q$, making it possible to identify the direction for $\mathbf{b}$ as well.

$$\begin{aligned}
\mathbf{b} &= \sum_p w_{ip}^n(\mathbf{u}_q+\mathbf{C}_q(\mathbf{x}_i^n-\mathbf{x}_p^n))e^{i(\mathbf{x}_p^n-\mathbf{x}_i^n)\cdot\mathbf{z}} && (46)\\
\mathbf{b} = \frac{1}{N_g}\sum_i\mathbf{b} &= \frac{1}{N_g}\sum_p\sum_i w_{ip}^n(\mathbf{u}_q+\mathbf{C}_q(\mathbf{x}_i^n-\mathbf{x}_p^n))e^{i(\mathbf{x}_p^n-\mathbf{x}_i^n)\cdot\mathbf{z}} && (47)\\
&= \frac{1}{N_g}\sum_p(h_q\mathbf{u}_q+\xi^{-1}\mathbf{C}_q\mathbf{e}_q) && (48)\\
&= \sum_q(h_q\overline{h_q}\mathbf{u}+\xi^{-1}\mathbf{u}\overline{\mathbf{e}_q}^T\mathbf{e}_q) && (49)\\
&= b\mathbf{u}. && (50)
\end{aligned}$$

2.6 Deformation gradient

The final remaining step in the transfers from grid to particles is the deformation gradient.

$$\begin{aligned}
\nabla \mathbf{u}_p^{n+1} &= \sum_i \tilde{\mathbf{u}}_i^{n+1} (\nabla w_{ip}^n)^T && (51)\\
&= \sum_i \epsilon g^{n+1} e^{i\mathbf{x}_i^n \cdot \mathbf{z}} \mathbf{u} (\nabla w_{ip}^n)^T && (52)\\
&= \epsilon g^{n+1} \mathbf{u} e^{i\mathbf{x}_p^n \cdot \mathbf{z}} \left(\sum_i e^{i(\mathbf{x}_i^n - \mathbf{x}_p^n)\cdot \mathbf{z}} \nabla w_{ip}^n \right)^T && (53)\\
&= \epsilon g^{n+1} e^{i\mathbf{x}_p^n \cdot \mathbf{z}} \mathbf{u} \overline{\mathbf{k}_q}^T && (54)\\
\mathbf{F}_p^{n+1} &= (\mathbf{I} + \Delta t \nabla \mathbf{u}_p^{n+1}) \mathbf{F}_p^n && (55)\\
&= (\mathbf{I} + \Delta t \epsilon g^{n+1} e^{i\mathbf{x}_p^n \cdot \mathbf{z}} \mathbf{u} \overline{\mathbf{k}_q}^T) \mathbf{F}_p^n && (56)
\end{aligned}$$

Substituting in the definition for $\mathbf{F}_p^n$ and neglecting higher order terms we have

$$\begin{aligned}
\mathbf{I} + \epsilon g^{n+1} \mathbf{F}_q e^{i\mathbf{x}_p^n \cdot \mathbf{z}} &= \left(\mathbf{I} + \Delta t \epsilon g^{n+1} \mathbf{u} \overline{\mathbf{k}_q}^T e^{i\mathbf{x}_p^n \cdot \mathbf{z}} \right) (\mathbf{I} + \epsilon g^n \mathbf{F}_q e^{i\mathbf{x}_p^n \cdot \mathbf{z}}) && (57)\\
\epsilon g^{n+1} \mathbf{F}_q e^{i\mathbf{x}_p^n \cdot \mathbf{z}} &= \Delta t \epsilon g^{n+1} e^{i\mathbf{x}_p^n \cdot \mathbf{z}} \mathbf{u} \overline{\mathbf{k}_q}^T + \epsilon g^n \mathbf{F}_q e^{i\mathbf{x}_p^n \cdot \mathbf{z}} && (58)\\
g \mathbf{F}_q &= \Delta t g \mathbf{u} \overline{\mathbf{k}_q}^T + \mathbf{F}_q && (59)\\
(g-1) \mathbf{F}_q &= \Delta t g \mathbf{u} \overline{\mathbf{k}_q}^T && (60)\\
\mathbf{F}_q &= \tau \mathbf{u} \overline{\mathbf{k}_q}^T. && (61)
\end{aligned}$$

Since positions are assumed not to move appreciably, we have completed the time step. In the process of keeping the equations relatively short, we have defined many intermediate quantities. We must next unravel these.

2.7 Solving for the vectors and matrices

We can eliminate $\mathbf{b}$ from $\mathbf{u}$

$$\mathbf{u} = \frac{b \mathbf{u} m - \Delta t V^0 \mathbf{c}}{w m g} \qquad (62)$$

$$\mathbf{u} = \frac{\Delta t V^0}{m(b - wg)} \mathbf{c} = \beta \mathbf{c}. \qquad (63)$$

Next, we eliminate $\mathbf{F}_q$ and $\mathbf{K}_q$ from $\mathbf{c}$, dropping into indexing notation to deal with tensors and defining a new matrix $\mathbf{A}$.

$$\mathbf{K}_q = \mathbf{M} : (\tau \mathbf{u} \overline{\mathbf{k}_q}^T) \tag{64}$$

$$\mathbf{c} = \sum_p \mathbf{K}_q \nabla w_{ip}^n e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot \mathbf{z}} \tag{65}$$

$$= \sum_p (\mathbf{M} : (\tau \mathbf{u} \overline{\mathbf{k}_q}^T)) \nabla w_{ip}^n e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot \mathbf{z}} \tag{66}$$

$$c_i = \tau \sum_p M_{ijkl} u_k \overline{k_{ql}} (\nabla w_{ip}^n)_j e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot \mathbf{z}} \tag{67}$$

$$= \tau M_{ijkl} u_k \sum_p \overline{k_{ql}} (\nabla w_{ip}^n)_j e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot \mathbf{z}} \tag{68}$$

$$= \tau M_{ijkl} u_k B_{jl} \tag{69}$$

$$= \tau A_{ik} u_k \tag{70}$$

$$\mathbf{c} = \tau \mathbf{A} \mathbf{u} \tag{71}$$

We can now eliminate $\mathbf{c}$ from the formula for $\mathbf{u}$ giving

$$\mathbf{u} = \beta \mathbf{c} = \beta \tau \mathbf{A} \mathbf{u}. \tag{72}$$

We are left with the eigenvalue problem $\mathbf{A}\mathbf{u} = \sigma \mathbf{u}$, where $\mathbf{u}$ is an eigenvector with eigenvalue σ and $\sigma \beta \tau = 1$.

2.8 Eliminating the scalars

We have now computed all of the vector and matrix quantities. We are however left with some scalars, which we must eliminate to get g, which is ultimately what tells us whether we are stable.

$$\tau = \frac{\Delta t g}{g - 1} \tag{73}$$

$$\beta = \frac{\Delta t V^0}{m(b - wg)} \tag{74}$$

$$\sigma \beta \tau = 1 \tag{75}$$

$$\frac{\Delta t^2 g V^0 \sigma}{m(g-1)(b-wg)} = 1 \tag{76}$$

$$m(g-1)(b-wg) = \Delta t^2 V^0 g \sigma \tag{77}$$

$$g^2 + \left(\frac{\Delta t^2 V^0 \sigma}{mw} - \frac{b}{w} - 1 \right) g + \frac{b}{w} = 0 \tag{78}$$

$$g^2 + 2rg + s = 0 \tag{79}$$

What remains is to examine the roots g. To do this, we will need to determine which quantities are real and also some bounds.

Before we do that, it is helpful to step back and take a look at the bigger picture, which is easy to lose in the details. By assumption, V^0, m, Δt, and $\mathbf{M}$ are known. The grid and particle locations ($\mathbf{x}_p^n$ and $\mathbf{x}_i^n$) were fixed in a regular arrangement, and the weights are known. We also fixed a wave-number $\mathbf{z}$, which for now we assume is known. This allows us (in principle at least) to directly calculate h_q, $\mathbf{k}_q$, $\mathbf{e}_q$, $\mathbf{B}$, b, w, s, c, and $\mathbf{A}$. We can then solve an eigenvalue problem to get $\mathbf{u}$ and σ. This allows us to compute r and $\mathbf{b}$ and solve the quadratic for g. The simulation is stable if $|g| \leq 1$.

2.9 Real, complex, and bounds

It is clear from the definitions that w, m, V^0, Δt, ξ, and b are all real and positive (except that $b = 0$ is possible), so c and s must be as well. In fact, $b \le w$. In the PIC case, where the $\mathbf{e}_q$ term would be missing, this can be shown easily.

$$w = \sum_p w_{ip}^n = \frac{1}{N_g} \sum_i \sum_p w_{ip}^n = \frac{1}{N_g} \sum_p \sum_i w_{ip}^n = \frac{1}{N_g} \sum_p 1 = \frac{N_p}{N_g} = N_q \tag{80}$$

$$|h_q| \le \sum_i w_{ip}^n |e^{i(\mathbf{x}_i^n - \mathbf{x}_p^n)\cdot\mathbf{z}}| = \sum_i w_{ip}^n = 1 \tag{81}$$

$$N_g b = \sum_p h_q \overline{h_q} = \sum_p |h_q|^2 \le \sum_p 1 = N_p \tag{82}$$

$$b \le N_q = w \tag{83}$$

We have also verified numerically that $b \le w$ in the APIC/CPIC case for quadratic and cubic splines in 2D and 3D. From this we conclude $0 \le s \le 1$.

When we defined the matrix $\mathbf{B}$, we provided two expressions, which we now show are equal.

$$\mathbf{B} = \sum_p \nabla w_{ip}^n \overline{\mathbf{k}_q}^T e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot\mathbf{z}} \tag{84}$$

$$= \frac{1}{N_g} \sum_i \sum_p \nabla w_{ip}^n \overline{\mathbf{k}_q}^T e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot\mathbf{z}} \tag{85}$$

$$= \frac{1}{N_g} \sum_p \left(\sum_i \nabla w_{ip}^n e^{i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot\mathbf{z}} \right) \overline{\mathbf{k}_q}^T \tag{86}$$

$$= \frac{1}{N_g} \sum_p \mathbf{k}_q \overline{\mathbf{k}_q}^T \tag{87}$$

$$= \sum_q \mathbf{k}_q \overline{\mathbf{k}_q}^T \tag{88}$$

From this form we see that $\mathbf{B}$ is Hermitian.

Using the symmetry assumption, we let $\bar{i}$ and $\bar{p}$ be the nodes and particles such that $\mathbf{x}_i = -\mathbf{x}_{\bar{i}}$ and $\mathbf{x}_p = -\mathbf{x}_{\bar{p}}$. Then, noting that weight gradients are odd functions,

$$\mathbf{k}_{\bar{q}} = \sum_{\bar{i}} \nabla w_{\bar{i}\bar{p}}^n e^{i(\mathbf{x}_{\bar{p}}^n - \mathbf{x}_{\bar{i}}^n)\cdot\mathbf{z}} \tag{89}$$

$$= -\sum_{\bar{i}} \nabla w_{ip}^n e^{-i(\mathbf{x}_p^n - \mathbf{x}_i^n)\cdot\mathbf{z}} \tag{90}$$

$$= -\overline{\mathbf{k}_q}. \tag{91}$$

Thus, $\mathbf{k}_q$ is pure imaginary, and $\mathbf{B}$ is a real, symmetric positive definite matrix. Noting that $M_{ijkl} = M_{klij}$, we conclude that $\mathbf{A}$ is also real and symmetric, so that its eigenvalues σ are real (and thus r is real as well).

2.10 Stability

We return to our quadratic equation $g^2 + 2rg + s = 0$. If $r^2 < s$ then g is complex and the roots are a complex conjugate pair whose product is $g\overline{g} = |g|^2 = s \le 1$, which is always stable. We conclude that the limit of stability (where $|g| = 1$) is reached when $g = \pm 1$. When $g = 1$,

$$0 = (1)^2 + 2r(1) + s = 1 + 2r + s = c\sigma.$$

σ does not depend on Δt, and $c > 0$. Thus, the time step restriction must be reached when $g = -1$.

$$\begin{aligned} 0 &= (-1)^2 + 2r(-1) + s = 1 - 2r + s = 2 - c\sigma + 2s \\ c &= \frac{2(s+1)}{\sigma} \end{aligned}$$

This leads to the bound on the time step size.

$$\Delta t \le \sqrt{\frac{mwc}{V^0}} = \sqrt{\frac{mw}{V^0}}\sqrt{\frac{2(s+1)}{\sigma}},$$

which must be true for all $\mathbf{z}$. The dependence on $\mathbf{z}$ is entirely within the second factor, where both s and σ depend on $\mathbf{z}$.

At this point it is difficult to make further progress, since the constitutive model, particle position, and wavenumber are all baked into σ. To make progress, we must assume an isotropic constitutive model.

2.11 Isotropic

If we assume that our constitutive model is isotropic, $M_{ijkl} = \lambda\delta_{ij}\delta_{kl} + \mu\delta_{ik}\delta_{jl} + \mu\delta_{il}\delta_{jk}$.

$$A_{ik} = M_{ijkl}B_{lj} \tag{92}$$

$$\mathbf{A} = \mu\,\mathrm{tr}(\mathbf{B})\mathbf{I} + (\lambda + \mu)\mathbf{B} \tag{93}$$

The eigenvectors of $\mathbf{A}$ are the same as the eigenvalues of $\mathbf{B}$. Let $\mathbf{B}\mathbf{y}_i = \kappa_i\mathbf{y}_j$. Since $\mathbf{B}$ is SPSD, $\kappa_i \ge 0$. Note that $\kappa_i \le \sum_j \kappa_j = \mathrm{tr}(\mathbf{B})$.

$$\mathbf{A}\mathbf{y}_i = (\lambda\,\mathrm{tr}(\mathbf{B}) + 2\mu\kappa_i)\mathbf{y}_i \tag{94}$$

$$\max_i \sigma_i = \lambda\,\mathrm{tr}(\mathbf{B}) + 2\mu\max_i \kappa_i \le (\lambda + 2\mu)\,\mathrm{tr}(\mathbf{B}) \tag{95}$$

with equality when $\mathbf{B}$ is rank one. This happens, for example, when there is only one particle per cell, located in the center of the cell. This leads to the condition

$$\Delta t = \sqrt{\frac{2mw(s+1)}{V^0(\lambda + 2\mu)\,\mathrm{tr}(\mathbf{B})}} = \sqrt{\frac{m/V^0}{\lambda + 2\mu}}\sqrt{\frac{2(s+1)}{\mathrm{tr}(\mathbf{B})/w}} \tag{96}$$

Note that the bound for Δt is split into two factors. The first is the sound speed (where $\rho = m/V^0$ is the density). The second factor depends on the integration scheme (PIC/APIC/CPIC, spline order, dimension), configuration (particle placement), and wavenumber $\mathbf{z}$, but it does not depend on the constitutive model. Define f as the fraction

$$f^2 = \frac{2(s+1)}{\Delta x^2\,\mathrm{tr}(\mathbf{B})/w} = \frac{2\sum_q(|h_q|^2 + \xi^{-1}\|\mathbf{e}_q\|^2 + 1)}{\Delta x^2 \sum_p \|\mathbf{k}_q\|^2} \le \max_q \frac{2(|h_q|^2 + \xi^{-1}\|\mathbf{e}_q\|^2 + 1)}{\Delta x^2\|\mathbf{k}_q\|^2}, \tag{97}$$

where we have used the fact that if $a, b, c, d > 0$ then $\frac{a}{c} \le \frac{a+b}{c+d} \le \frac{b}{d}$ or $\frac{b}{d} \le \frac{a+b}{c+d} \le \frac{a}{c}$. We conclude that the smallest possible value of f, and thus the tightest time step restriction, is obtained with a single particle. The problem of bounding f can be done by brute force. This leads to a constrained optimization problem in the particle position $\mathbf{x}_p^n$ and wavenumber $\mathbf{z}$ (six degrees of freedom total). The solutions to these problems are shown in Figure 1.

3 Single particle instability

It is possible for a simulation that satisfies its CFL restriction to be unstable. One particularly simple case where this is observed is for a single particle. This case occurs in practice when a particle becomes isolated,

scheme	spline	Analytic f	Sim 2D f	Sim 3D f	Analytic f exact form
APIC	quad	1.0000	1.0007	1.0017	1
CPIC	quad	1.0000	1.0011	1.0029	1
PIC	quad	0.7071	0.7133	0.7182	$\frac{1}{\sqrt{2}}$
APIC	cubic	1.7042	1.7055	1.7072	$\frac{\sqrt{9594+1365\sqrt{35}}}{78}$
CPIC	cubic	1.3952	1.3993	1.4028	$\sqrt{\frac{146}{75}}$
PIC	cubic	1.4033	1.4055	1.4130	$12960f^6 - 25902f^4 + 780f^2 - 59$

Figure 1: The Von Neumann time step restriction is $\Delta t \leq f \Delta x / c$, where c is the time step restriction. This table shows the constants f for each version of the scheme.

such as when water splashes or sand spreads out. In [1], an effective time step restriction for the single particle instability was proposed in the case of fluids. In that paper, no solution was proposed in the more difficult case of solids. In this section, we derive a time step restriction for solids in the vicinity of the rest configuration ($\mathbf{F}_p^n \approx \mathbf{I}$). This assumption is not as limiting as it might seem. Particles that are at risk of becoming isolated tend to be near the surface, where they would not be expected to experience significant strain.

Our strategy for deriving a time step restriction is to consider the stability of perturbations from the rest configuration. The momentum of an isolated particle is constant in the absence of outside forces, so that its velocity does not change. Ignoring grid-related effects, the velocity of the particle is decoupled from the rest of the dynamics. In practice, the location of a particle relative to the grid can affect its stability, with some placements within a cell requiring a smaller time step for stability. A particle traveling through the grid would experience a sort of average of the local stability along its trajectory. To handle this, we evaluate the stability of a stationary particle at an arbitrary location within a cell and then select a time step size that is stable for all such locations.

3.1 Stability formulation

At the beginning of each time step, the isolated particle has state m_p, $\mathbf{x}_p^n$, $\mathbf{v}_p^n$, $\mathbf{C}_p^n$, and $\mathbf{F}_p^n$. The mass m_p does not change. Since we are assuming that the particle is stationary, $\mathbf{v}_p^n = \mathbf{0}$ and $\mathbf{x}_p^n$ is constant. We assume a small perturbation from the rest configuration, so that $\mathbf{C}_p^n = \epsilon \mathbf{A}^n$ and $\mathbf{F}_p^n = \mathbf{I} + \epsilon \mathbf{E}^n$. After the time step, we will have a new state $\mathbf{C}_p^{n+1} = \epsilon \mathbf{A}^{n+1}$ and $\mathbf{F}_p^{n+1} = \mathbf{I} + \epsilon \mathbf{E}^{n+1}$. The new state variables $(\mathbf{E}^{n+1}, \mathbf{F}_p^{n+1})$ are related to the originals $(\mathbf{E}^n, \mathbf{F}_p^n)$ by a matrix $\mathbf{N}$. Since the changing portion of the state consists of two 3×3 matrices, $\mathbf{N}$ will be a 18×18 matrix. A stable time step size is one such that the spectral radius (largest eigenvalue magnitude) of $\mathbf{N}$ is no larger than 1.

3.2 Transfers to grid and grid update

We begin by transferring the particle's state to the grid.

$$m_i^n = m_p w_{ip}^n \tag{98}$$

$$m_i^n \mathbf{u}_i^n = w_{ip}^n m_p (\mathbf{u}_p^n + \mathbf{C}_p^n (\mathbf{x}_i^n - \mathbf{x}_p^n)) \tag{99}$$

$$\mathbf{u}_i^n = \mathbf{C}_p^n (\mathbf{x}_i^n - \mathbf{x}_p^n) \tag{100}$$

Next, we compute particle forces. Let $\mathbf{M} = \frac{\partial \mathbf{P}}{\partial \mathbf{F}}(\mathbf{F}_p^n)$ and $\mathbf{K} = \mathbf{M} : \mathbf{E}^n$ so that the forces are

$$
\begin{aligned}
\mathbf{P}(\mathbf{F}_p^n) &= \epsilon \mathbf{M} : \mathbf{E}^n = \epsilon \mathbf{K} && (101)\\
\mathbf{f}_i^n &= -\sum_p V_p^0 (\mathbf{P}(\mathbf{F}_p^n))(\mathbf{F}_p^n)^T \nabla w_{ip}^n && (102)\\
&= -V_p^0 \epsilon \mathbf{K}(\mathbf{I} + \epsilon \mathbf{E}^n)^T \nabla w_{ip}^n && (103)\\
&= -\epsilon V_p^0 \mathbf{K} \nabla w_{ip}^n && (104)
\end{aligned}
$$

Finally, we update the grid velocities.

$$
\begin{aligned}
\tilde{\mathbf{u}}_i^{n+1} &= \mathbf{u}_i^n + \Delta t (m_i^n)^{-1} \mathbf{f}_i^n && (105)\\
\tilde{\mathbf{u}}_i^{n+1} &= \epsilon(\mathbf{A}^n(\mathbf{x}_i^n - \mathbf{x}_p^n) - \Delta t (w_{ip}^n m_p)^{-1} V_p^0 \mathbf{K} \nabla w_{ip}^n) && (106)
\end{aligned}
$$

3.3 Grid to particle transfers

We must now transfer the updated grid state back to particles. As expected, the new velocities will be zero since

$$
\begin{aligned}
\mathbf{u}_p^{n+1} &= \sum_i w_{ip}^n \tilde{\mathbf{u}}_i^{n+1} && (107)\\
&= \sum_i w_{ip}^n \epsilon(\mathbf{A}^n(\mathbf{x}_i^n - \mathbf{x}_p^n) - \Delta t (w_{ip}^n m_p)^{-1}(V_p^0 \mathbf{K} \nabla w_{ip}^n)) && (108)\\
&= \epsilon \mathbf{A}^n \sum_i w_{ip}^n (\mathbf{x}_i^n - \mathbf{x}_p^n) - \frac{\epsilon \Delta t V_p^0}{m_p} \mathbf{K} \sum_i \nabla w_{ip}^n = 0 && (109)
\end{aligned}
$$

Before moving on, we introduce the definition

$$
\mathbf{R} = \frac{1}{\xi} \sum_i \frac{\nabla w_{ip}^n (\nabla w_{ip}^n)^T}{w_{ip}^n}. \tag{110}
$$

Using equation (13) we have

$$
\begin{aligned}
\nabla \mathbf{u}_p^{n+1} &= \sum_i \tilde{\mathbf{u}}_i^{n+1} (\nabla w_{ip}^n)^T && (111)\\
&= \sum_i \epsilon(\mathbf{A}^n(\mathbf{x}_i^n - \mathbf{x}_p^n) - \Delta t (w_{ip}^n m_p)^{-1}(V_p^0 \mathbf{K} \nabla w_{ip}^n))(\nabla w_{ip}^n)^T && (112)\\
&= \epsilon \mathbf{A}^n \sum_i (\mathbf{x}_i^n - \mathbf{x}_p^n)(\nabla w_{ip}^n)^T - \frac{\epsilon \Delta t V_p^0}{m_p} \mathbf{K} \sum_i (w_{ip}^n)^{-1} \nabla w_{ip}^n (\nabla w_{ip}^n)^T && (113)\\
&= \epsilon \mathbf{A}^n - \frac{\epsilon \xi \Delta t V_p^0}{m_p} \mathbf{K} \mathbf{R} && (114)\\
\mathbf{F}_p^{n+1} &= (\mathbf{I} + \Delta t \nabla \mathbf{u}_p^{n+1}) \mathbf{F}_p^n && (115)\\
&= \left(\mathbf{I} + \epsilon \Delta t \mathbf{A}^n - \frac{\epsilon \xi \Delta t^2 V_p^0}{m_p} \mathbf{K} \mathbf{R} \right)(\mathbf{I} + \epsilon \mathbf{E}^n) && (116)\\
&= \mathbf{I} + \epsilon \Delta t \mathbf{A}^n - \frac{\epsilon \xi \Delta t^2 V_p^0}{m_p} \mathbf{K} \mathbf{R} + \epsilon \mathbf{E}^n && (117)\\
\mathbf{E}^{n+1} &= \Delta t \mathbf{A}^n - \frac{\xi \Delta t^2 V_p^0}{m_p} (\mathbf{M} : \mathbf{E}^n) \mathbf{R} + \mathbf{E}^n && (118)
\end{aligned}
$$

What remains is to update the affine state.

$$\begin{aligned}
\mathbf{C}_p^{n+1} &= \xi \sum_i w_{ip}^n \tilde{\mathbf{u}}_i^{n+1} (\mathbf{x}_i^n - \mathbf{x}_p^n)^T \qquad &(119)\\
&= \xi \sum_i w_{ip}^n (\epsilon(\mathbf{A}^n(\mathbf{x}_i^n - \mathbf{x}_p^n) - \Delta t (w_{ip}^n m_p)^{-1} V_p^0 \mathbf{K} \nabla w_{ip}^n))(\mathbf{x}_i^n - \mathbf{x}_p^n)^T \qquad &(120)\\
&= \xi \epsilon \mathbf{A}^n \sum_i w_{ip}^n (\mathbf{x}_i^n - \mathbf{x}_p^n)(\mathbf{x}_i^n - \mathbf{x}_p^n)^T - \frac{\epsilon \xi \Delta t V_p^0}{m_p} \mathbf{K} \sum_i \nabla w_{ip}^n (\mathbf{x}_i^n - \mathbf{x}_p^n)^T \qquad &(121)\\
&= \epsilon \mathbf{A}^n - \frac{\epsilon \xi \Delta t V_p^0}{m_p} \mathbf{M} : \mathbf{E}^n \qquad &(122)\\
\mathbf{A}^{n+1} &= \mathbf{A}^n - \frac{\xi \Delta t V_p^0}{m_p} \mathbf{M} : \mathbf{E}^n \qquad &(123)
\end{aligned}$$

Let $\hat{\mathbf{M}} = \frac{\xi \Delta t^2 V_p^0}{m_p} \mathbf{M}$. Then, we can express these as a sort of matrix equation using indexing notation

$$\begin{pmatrix} \Delta t A_{ij}^{n+1} \\ E_{ij}^{n+1} \end{pmatrix} = \begin{pmatrix} \delta_{ik}\delta_{jl} & -\hat{M}_{ijkl} \\ \delta_{ik}\delta_{jl} & \delta_{ik}\delta_{jl} - \hat{M}_{imkl} R_{mj} \end{pmatrix} \begin{pmatrix} \Delta t A_{kl}^n \\ E_{kl}^n \end{pmatrix} \tag{124}$$

$$\mathbf{N} = \begin{pmatrix} \delta_{ik}\delta_{jl} & -\hat{M}_{ijkl} \\ \delta_{ik}\delta_{jl} & \delta_{ik}\delta_{jl} - \hat{M}_{imkl} R_{mj} \end{pmatrix} \tag{125}$$

Note that we drop the boldface in index notation, since the components are scalars. We define $\mathbf{N}$ to be the resulting 18×18 matrix.

3.4 Simplifying the system

Assuming an isotropic constitutive model, we can write $M_{ijkl} = \lambda \delta_{ij}\delta_{kl} + \mu \delta_{ik}\delta_{jl} + \mu \delta_{il}\delta_{jk}$. We define new unitless scalars p and s so that $s = \frac{\xi \Delta t^2 V_p^0 \mu}{m_p}$ and $\lambda = p\mu$. The scalar p is a function of the Poisson's ratio given by $p = \frac{2\nu}{1-2\nu}$. We assume that $\mu, \lambda > 0$ so that $s > 0$ and $p > 0$.

Before continuing, it is helpful to note the special form of $\mathbf{R}$. In the case of CPIC transfers, we conveniently have $\mathbf{R} = \mathbf{I}$. In the case of PIC and APIC, $\mathbf{R}$ is diagonal. Let $a = \frac{x}{\Delta x}$. The first entry of $\mathbf{R}$ is then

$$\mathbf{R}_{11} = \frac{3}{3 - 4a^2} \qquad \text{Quadratic splines} \qquad -\frac{1}{2} \leq a \leq \frac{1}{2} \tag{126}$$

$$\mathbf{R}_{11} = \frac{-2(9a^4 - 18a^3 + 2a^2 + 7a + 2)}{(3a^3 - 6a^2 + 4)(3a^3 - 3a^2 - 3a - 1)} \qquad \text{Cubic splines} \qquad 0 \leq a \leq 1 \tag{127}$$

The other nonzero entries $\mathbf{R}_{22}$ and $\mathbf{R}_{33}$ are similar. Note that the entries are decoupled. That is, $\mathbf{R}_{11}$ depends on x but not y or z. Let $r_i = R_{ii}$ be the diagonal entries and $b = r_1 + r_2 + r_3$, $c = r_1 r_2 + r_2 r_3 + r_3 r_1$, and $d = r_1 r_2 r_3$ be the invariants. For CPIC, $r_i = 1$. For APIC with quadratic splines, $1 \leq r_i \leq \frac{3}{2}$. For APIC with cubic splines[1], $1 \leq r_i \leq 1.045606358$. The APIC/CPIC distinction and the particle placement are entirely encapsulated in r_i, which conveniently allows all of these cases to be treated together. we address the PIC case separately. We proceed under the assumption of 3D.

[1] The upper bound is a root of $27x^4 - 108x^3 + 378x^2 - 506x + 207$.

3.5 Solving the eigenvalue problem

The characteristic polynomial $P(\lambda)$ of $\mathbf{N}$ is

$$P(\lambda; s, p, r_i) = (1-\lambda)^6 Q(\lambda; s, p, r_i) \tag{128}$$

$$S(\lambda; s, r_1, r_2) S(\lambda; s, r_2, r_3) S(\lambda; s, r_1, r_3) \tag{129}$$

$$S(\lambda; s, r_i, r_j) = \lambda^2 + (r_i s + r_j s - 2)\lambda - s r_i - s r_j + 2s + 1 \tag{130}$$

$$Q(\lambda; s, p, r_i) = \lambda^6 + (-6 + b(p+2)s)\lambda^5 + (15 + 4c(p+1)s^2 - (p+2)(5b-3)s)\lambda^4 \tag{131}$$

$$+ (-20 + 4d(3p+2)s^3 + 8(p+1)(b-2c)s^2 + 2(p+2)(5b-6)s)\lambda^3 \tag{132}$$

$$+ (15 + 4(3p+2)(c-3d)s^3 - 12(p+1)(2b-2c-1)s^2 - 2(p+2)(5b-9)s)\lambda^2 \tag{133}$$

$$+ (-6 + 4(3p+2)(b-2c+3d)s^3 + 8(p+1)(3b-2c-3)s^2 + (p+2)(5b-12)s)\lambda \tag{134}$$

$$+ 1 - 4(3p+2)(b-c+d-1)s^3 - 4(p+1)(2b-c-3)s^2 - (p+2)(b-3)s \tag{135}$$

The goal is to find

$$s^* = \min_{1 \le r_i \le r} s \qquad \text{such that} \qquad P(\lambda; s, p, r_i) = 0, |\lambda| = 1, s > 0.$$

The time step restriction is then readily deduced from s^*. Note that the factors of $P(\lambda; s, p, r_i)$ can be treated separately. The $(1-\lambda)^6$ factor does not impose a time step restriction and can be discarded.[2] There are now two high level cases to consider: $S(\lambda; s, r_i, r_j) = 0$ or $Q(\lambda; s, p, r_i) = 0$.

3.6 Case $S(\lambda; s, r_i, r_j) = 0$

This is a quadratic equation in s. Let's first consider the case of complex conjugate solutions. Note that $(\lambda - \kappa)(\lambda - \overline{\kappa}) = \lambda^2 - (\kappa + \overline{\kappa})\lambda + \kappa\overline{\kappa}$, so that the square of the magnitude of the eigenvalues are just the constant term of the quadratic. That is, we need $1 = |\lambda|^2 = -s r_i - s r_j + 2s + 1$ which leads to $s(2 - r_i - r_j) = 0$. This case does not lead to a useful time step restriction ($s = 0$ implies $\Delta t = 0$). Next, consider the case of real eigenvalues. $0 = S(1; s, r_1, r_2) = 2s$ does not lead to a useful solution, but $0 = S(-1; s, r_1, r_2) = 2(s + 2 - r_1 s - r_2 s)$ leads to

$$s \le \frac{2}{r_1 + r_2 - 1} \le \frac{2}{2r - 1} = \alpha, \tag{136}$$

where r is the appropriate upper bound on r_i (listed above). We will use α to refer to this bound.

3.7 Case $Q(\lambda; s, p, r_i) = 0$, $\lambda = \pm 1$

The positive case $Q(1; s, p, r_i) = 4s^3(3p+2)$ does not lead to a time step restriction. The case $Q(-1; s, p, r_i) = 0$ leads to a cubic polynomial in s, which could in theory be solved for $s(p, r_1, r_2, r_3)$. This would then need to be minimized over all feasible $1 \le r_1, r_2, r_3 \le r$ to produce the bound $s \le s^*(p)$. At first, this task appears

[2] This factor implies that $\mathbf{N}$ always has 1 as an eigenvalue with multiplicity (at least) six. This raises two interesting questions. (1) Is the matrix diagonalizable? (No.) (2) What are the corresponding (generalized) eigenvalues? The three eigenvectors are of the form $\mathbf{A}^n = -(\mathbf{A}^n)^T$ with $\mathbf{E}^n = \mathbf{0}$. These correspond to conservation of angular momentum. The three generalized eigenvectors are of the form $\mathbf{E}^n = -(\mathbf{E}^n)^T$ with $\mathbf{A}^n = \mathbf{0}$. These correspond to $\mathbf{F}_p^n$ being a (linearized) rotation. In particular, this implies that there will always be *linear growth* in a single particle simulation, at least near the rest configuration. This is not unlike the evolution of a particle moving with constant velocity, where velocity is an eigenvector with eigenvalue 1, and position is a corresponding generalized eigenvalue. We do not consider such linear growth to be unstable. In 2D, there are only two trivial eigenvalues, since there is only one degree of freedom for rotation. For PIC, the number of trivial eigenvalues is halved, with skew-symmetric $\mathbf{E}^n$ as eigenvalues. In practice, the defectiveness of $\mathbf{N}$ also makes numerical eigenvalue procedures less accurate, which complicates numerical tests. Since the eigenvectors and generalized eigenvectors are known, we can use Householder reflections to transform these eigenvectors into $\mathbf{e}_1, \ldots, \mathbf{e}_6$, which moves the defective part to the top left 6×6 block. Then, we then simply discard this part, reducing $\mathbf{N}$ to a 12×12 matrix with the same nontrivial eigenvalues. This matrix is smaller and generally diagonalizable, which makes numerical eigenvalues more accurate and cheaper to compute.

to be infeasible. However, we note that p is constant for out purposes (the Poisson's ratio is not changing). Further, the partial derivatives $\frac{\partial s}{\partial r_i}$ can be computed implicitly by differentiating the polynomial equation $Q(-1; s, p, r_i) = 0$. Indeed, $0 = \frac{\partial}{\partial r_k} Q(-1; s, p, r_i) = \frac{\partial Q}{\partial s}(-1; s, p, r_i)\frac{\partial s}{\partial r_k} + \frac{\partial Q}{\partial r_k}(-1; s, p, r_i)$. Since $\frac{\partial s}{\partial r_k} = 0$, we have $\frac{\partial Q}{\partial r_k}(-1; s, p, r_i) = 0$. Since this is a constrained minimization with box constraints each r_i may independently be at its lower ($r_i = 1$) or upper ($r_i = r$) bound or unconstrained ($\frac{\partial Q}{\partial r_k}(-1; s, p, r_i) = 0$). We consider the cases in turn. The r_k are equivalent, so permutations need not be considered.

Partially unconstrained. Consider that at least one r_i is unconstrained (say, r_1). Then, $Q(-1; s, p, r_i) = \frac{\partial Q}{\partial r_1}(-1; s, p, r_i) = 0$. Eliminating r_1, p from this system of equations leads to the equation $(2r_3 s - s - 2)^2(2r_2 s - s - 2)^2 = 0$, which implies $2r_3 s - s - 2 = 0$ or $2r_2 s - s - 2 = 0$. Both of these lead back to (136). We are left with the case that all r_i are at their bounds 1 or r.

Boundary cases. There are four boundary cases (depending on how many of the r_i are 1 and r). $0 = Q(-1; s, p, r_1, r_1, r_1) = -4(2r_1 s - s - 2)^2(6pr_1 s - 3ps + 4r_1 s - 2s - 4)$. The middle factor leads to (136). The last factor produces the restriction $s^* \le \beta$ where

$$s^* \le \frac{4}{(2r_1 - 1)(3p + 2)} \le \frac{4}{(2r - 1)(3p + 2)} = \beta, \tag{137}$$

which is more strict than (136) since $0 < \beta < \alpha$. $0 = Q(-1; s, p, r_1, r_2, r_2) = -4(2r_2 s - s - 2)u(s, p, r_1, r_2)$ where $u(s, p, r_1, r_2) = (2r_2 - 1)(2r_1 - 1)(3p + 2)s^2 + (-4pr_1 - 8pr_2 + 6p - 8r_1 - 8r_2 + 8)s + 8$. The factor $(2r_2 s - s - 2)$ leads to (136). The last factor $u(s, p, r_1, r_2)$ is a quadratic in s. Further, one can show that $u(\beta, p, r_1, r_2) > 0$, $u_s(\beta, p, r_1, r_2) < 0$, and $u_{ss}(\beta, p, r_1, r_2) > 0$ for all $1 \le r_1, r_2 \le r$ and $p > 0$. We conclude that any roots s of $u(s, p, r_1, r_2)$, if they are real, must satisfy $s > \beta$. $s^* \le \beta$ from (137) will be the time step restriction for the single particle instability in 3D.

3.8 Case $Q(\lambda; s, p, r_i) = 0$, $|\lambda| = 1$, λ complex

We are now left with just one case, where the solution to $Q(\lambda; s, p, r_i) = 0$ with the largest magnitude is complex. Since $Q(\lambda; s, p, r_i)$ has real coefficients, we will also have $Q(\overline{\lambda}) = 0$. From the system of equations $\{Q(\lambda; s, p, r_i) = 0, Q(\overline{\lambda}) = 0, \lambda\overline{\lambda} = 1\}$ we eliminate λ and $\overline{\lambda}$. This yields a polynomial $R(s, p, r_1, r_2, r_3)$ that is degree 5 in s and has total degree 21. The optimization strategy we used earlier is intractable on this polynomial. Instead, we randomly sampled a billion configurations with $p > 0$, $1 \le r_1 \le r_2 \le r_3 \le \frac{3}{2}$, and $0 < s < \beta$. We then evaluated $R(s, p, r_1, r_2, r_3)$ at each configuration and established that, within the limits of quad floating point precision, all have the same sign. This strongly suggests that no smaller time bounds exist within this case.[3]

3.9 Analysis for 2D simulation

The analysis above was for 3D. In 2D, things are slightly different. The characteristic polynomial $P(\lambda)$ of $\mathbf{N}$ is

$$P(\lambda; s, p, r_i) = (1 - \lambda)^2 U(\lambda; s, p, r_i) S(\lambda; s, p, r_1, r_2) \tag{138}$$

$$U(\lambda; s, p, r_i) = \lambda^4 + (bps + 2bs - 4)\lambda^3 + (4cps^2 - 3bps + 4cs^2 - 6bs + 2ps + 4s + 6)\lambda^2 \tag{139}$$

$$+ (4bps^2 - 8cps^2 + 3bps + 4bs^2 - 8cs^2 + 6bs - 4ps - 8s - 4)\lambda \tag{140}$$

$$- 4bps^2 + 4cps^2 - bps - 4bs^2 + 4cs^2 + 4ps^2 - 2bs + 2sp + 4s^2 + 4s + 1 \tag{141}$$

[3] It is of course possible that there is a multiple root. For example, $x^2 = 0$ has a solution, even though the polynomial is never negative. Numerical tests show that the largest stable time step size occurs when $\lambda = -1$ for some eigenvalue, though when this occurs there may also be complex eigenvalues with $|\lambda| = 1$ as well. This suggests that we are not missing a tighter time step restriction from this case.

where $b = r_1 + r_2$ and $c = r_1 r_2$ are the invariants and $S(\lambda; s, p, r_1, r_2)$ is the same as in the 3D analysis. As in the 3D case, the first factor produces no restrictions and the last factor leads to (136). This leaves $U(\lambda; s, p, r_i)$.

Real cases $U(1; s, p, r_i) = 4s^2(p + 1)$ leads to no time step restrictions. The system of polynomials $U(-1; s, p, r_i) = 0$ and $\frac{\partial}{\partial r_1} U(-1; s, p, r_i) = 0$ implies $p = 0$, which is not allowed. Thus, the variables r_1 and r_2 must both be at their bounds. Letting $r_1 = r_2$ leads to $4(2r_1 s - s - 2)(2pr_1 s - ps + 2r_1 s - s - 2) = 0$. The second factor leads to (136), and the third leads to

$$s \le \frac{4}{(2r_1 - 1)(3p + 2)} \le \frac{2}{(2r - 1)(p + 1)} = \gamma. \tag{142}$$

This is directly analogous to (137). Otherwise, we are in the case $r_1 = r$, $r_2 = 1$, which leads to $V(s, p, r) = 4(2r-1)(p+1)s^2 - 8r(p+2)s + 16 = 0$. From $V(\gamma, p, r) > 0$, $V_s(\gamma, p, r) < 0$, and $V_{ss}(\gamma, p, r) > 0$ we conclude that any solutions to $V(s, p, r) = 0$ satisfy $s > \gamma$.

Complex case Unlike the 3D case, the case of $U(\lambda; s, p, r_i)$ with λ complex is quite manageable analytically. Eliminating λ and $\overline{\lambda}$ from the system $[U(\lambda; s, p, r_i), U(\overline{\lambda}; s, p, r_i), \lambda\overline{\lambda} - 1]$ yields $V(s, p, r_i) = 0$ where

$$V(s, p, r_i) = 16(r_2 - 1)^2(r_1 - 1)^2(r_1 + r_2 - 1)(p + 1)^2 s^2 - (4(r_2 - 1)(r_1 - 1)(p + 2)(p + 1)s + p^2)(r_1 + r_2 - 2)^2.$$

The system $[V(s, p, r_i), \frac{\partial V}{\partial r_1}(s, p, r_i)]$ with $r_1, r_2 > 1$ and $p > 0$ has no feasible solutions. $r_1 = 1$ implies $r_2 = 1$ but produces no time step restriction. This leaves $r_1 = r_2 = r > 1$, which leads to

$$W(s, p, r) = 4(r - 1)^2(2r - 1)(p + 1)^2 s^2 - 4(r - 1)^2(p + 2)(p + 1)s - p^2 = 0$$

Noting that $W(0, r) < 0$, $W(\gamma, r) < 0$, and $W_{ss}(s, r) > 0$ we conclude that there are no roots with $0 < s < \gamma$.

3.10 Analysis for PIC 3D

In the PIC case, there are no $\mathbf{A}$ dofs, so that $\mathbf{N}$ becomes just the lower right 9×9 block of (124): $\mathbf{N}_{(ik)(jl)} = \delta_{ik}\delta_{jl} - \hat{M}_{imkl} R_{mj}$, where (ik) and (jl) are interpreted as a single flat index $1, \dots, 9$. The characteristic polynomial $P(\lambda)$ of $\mathbf{N}$ is

$$P(\lambda; s, p, r_i) = (1 - \lambda)^3(r_2 s + r_3 s + \lambda - 1)(r_1 s + r_3 s + \lambda - 1)(r_1 s + r_2 s + \lambda - 1)Q(\lambda; s, p, r_i) \tag{143}$$

$$Q(\lambda; s, p, r_i) = \lambda^3 + (bps + 2bs - 3)\lambda^2 + (4cps^2 - 2bps + 4cs^2 - 4bs + 3)\lambda \tag{144}$$

$$+ 12dps^3 - 4cps^2 + 8ds^3 + psb - 4cs^2 + 2sb - 1 \tag{145}$$

where $b = r_1 + r_2 + r_3$, $c = r_1 r_2 + r_2 r_3 + r_3 r_1$, and $d = r_1 r_2 r_3$. The first factor does not depend on s. The second, third, and fourth factors lead to

$$s \le \frac{2}{r_i + r_j} \le \frac{1}{r} = \kappa. \tag{146}$$

This leaves $Q(\lambda; s, p, r_i)$. $0 = Q(1; s, p, r_i) = 4ds^3(3p + 2)$ does not lead to a time step restriction.

Real eigenvalues. Next, let's consider $0 = Q(-1; s, p, r_i)$, which is a cubic equation in s. If we assume that r_1 is unconstrained, then we have a system of two equations $0 = Q(-1; s, p, r_i)$, $0 = \frac{\partial Q}{\partial r_1}(-1; s, p, r_i)$. Eliminating p, r_1 yields the equation $(r_3 s - 1)^2(r_2 s - 1)^2 = 0$, which leads back to (146). This leaves the case where all r_i are constrained to 1 or r. $0 = Q(-1; s, p, r_1, r_1, r_1) = 4(r_1 s - 1)^2(3pr_1 s + 2r_1 s - 2)$ leads to (146) and

$$s \le \frac{1}{r_1(1 + \frac{3}{2}p)} \le \frac{1}{r(1 + \frac{3}{2}p)} = \zeta. \tag{147}$$

This leaves the case with only two equal, $0 = Q(-1; s, p, r_1, r_2, r_2) = 4(r_2s-1)U(s, p, r_1, r_2)$ with $U(s, p, r_1, r_2) = r_1r_2(3p + 2)s^2 - (pr_1 + 2pr_2 + 2r_1 + 2r_2)s + 2$. The second factor leads to (146). From $U(\zeta, p, r_1, r_2) > 0$, $U_s(\zeta, p, r_1, r_2) < 0$, and $U_{ss}(s, p, r_1, r_2) > 0$ we conclude that there are no solutions $0 < s < \zeta$. Note that $\zeta < \kappa$.

Complex eigenvalues. The last case that remains is when $0 = Q(\lambda; s, p, r_i)$ with $|\lambda| = 1$ and λ complex. Eliminating λ and $\overline{\lambda}$ from the system $[Q(\lambda; s, p, r_i) = 0, Q(\overline{\lambda}; s, p, r_i) = 0, \lambda\overline{\lambda} = 1]$ yields

$$\begin{aligned} 0 = U(s, p, r_i) &= -4d^2(3p + 2)^2s^3 + 8dc(p + 1)(3p + 2)s^2 && (148)\\ &\quad + (-3bdp^2 - 4c^2p^2 - 8bdp - 8c^2p - 4bd - 4c^2)s + bcp^2 + 3bcp + 2bc - 3dp - 2d && (149)\end{aligned}$$

Complex eigenvalues: equality. We begin this case by considering addressing $r_3 = r_2$.

$$\begin{aligned} 0 = U(s, p, r_1, r_2, r_2) &= r_2(6pr_1r_2s + 4r_1r_2s - pr_1 - 2pr_2 - 2r_1 - 2r_2)V(s, p, r_1, r_2) && (150)\\ V(s, p, r_1, r_2) &= 6pr_1r_2^2s^2 + 4r_1r_2^2s^2 - 7pr_1r_2s - 2pr_2^2s - 6r_1r_2s - 2r_2^2s + 2pr_1 + pr_2 + 2r_1 + 2r_2 && (151)\end{aligned}$$

The second factor leads to $s = \frac{1+\frac{3}{4}p}{r(1+\frac{3}{2}p)} > \zeta$. Noting that $V(\zeta, p, r_1, r_2) > 0$, $V_s(\zeta, p, r_1, r_2) < 0$, and $V_{ss}(\zeta, p, r_1, r_2) > 0$, we conclude that there are no solutions s to $V(s, p, r_1, r_2) = 0$ with $0 < s < \zeta$. Thus, $r_3 = r_2$. does not lead to a tighter time step bound.

Complex eigenvalues: two unconstrained. We may now assume that r_i are all distinct. Note that this implies that at least one r_i is unconstrained. If two of r_i are unconstrained, then we have $[U(s, p, r_1, r_2, r_3) = 0, \frac{\partial U}{\partial r_1}(s, p, r_1, r_2, r_3) = 0, \frac{\partial U}{\partial r_2}(s, p, r_1, r_2, r_3) = 0]$ along with the assumptions $r_1 \neq r_2$, $r_2 \neq r_3$, $r_1 \neq r_3$, $p > 0$, $s > 0$, which has no solutions.

Complex eigenvalues: $1 = r_2 < r_1 < r_3 = r$. This leaves the case $[U(s, p, r_1, r_2, r_2) = 0, \frac{\partial U}{\partial r_1}(s, p, r_1, r_2, r_2) = 0, r_2 = 1, r_3 = r]$. Eliminating r_1 gives us $0 = W(s, p, r)$ where

$$\begin{aligned} W(s, p, r) &= r^2(3p + 2)^2(p^2r^2 + 2p^2r + 4pr^2 + p^2 - 8pr + 4r^2 + 4p - 8r + 4)s^2 && (152)\\ &\quad - 2r(r + 1)(3p + 2)(p^3r^2 + p^3r + 5p^2r^2 + p^3 - 6p^2r + 8pr^2 + 5p^2 - 16pr + 4r^2 + 8p - 8r + 4)s\\ &\quad + p^4r^4 + 2p^4r^3 + 6p^3r^4 + 3p^4r^2 + 6p^3r^3 + 13p^2r^4 + 2p^4r + 4p^2r^3 + 12pr^4 + p^4 + 6p^3r\\ &\quad - 18p^2r^2 + 4r^4 + 6p^3 + 4p^2r - 24pr^2 + 13p^2 - 8r^2 + 12p + 4.\end{aligned}$$

Then, we can use $W(\zeta, p, r) > 0$, $W_s(\zeta, p, r) < 0$, and $W_{ss}(\zeta, p, r) > 0$ to conclude that any solutions s to $0 = W(s, p, r)$ must satisfy $s > \zeta$. This completes the PIC 3D case. The tightest bound is $s \leq \zeta$.

3.11 Analysis for PIC 2D

In the PIC 2D case, the characteristic polynomial of $\mathbf{N}$ is

$$\begin{aligned} P(\lambda; s, p, r_1, r_2) &= (\lambda - 1)(r_1s + r_2s + \lambda - 1)U(s, p, r_1, r_2) && (153)\\ U(\lambda; s, p, r_1, r_2) &= 4r_2r_1(p + 1)s^2 + (r_1 + r_2)(p + 2)(\lambda - 1)s + (\lambda - 1)^2 && (154)\end{aligned}$$

The first factor $\lambda-1$ does not lead to a time step restriction. The second leads to (146). $0 = U(1; s, p, r_1, r_2) = 4r_2r_1(p + 1)s^2$ produces no time step restriction. Next we handle the case $\lambda = -1$.

$$0 = U(-1; s, p, r_1, r_2) = 4r_2r_1(p + 1)s^2 - 2(r_1 + r_2)(p + 2)s + 4 \qquad (155)$$

We note that

$$0 = U(-1; s, p, r, r) = 4(rs - 1)(prs + rs - 1) \qquad (156)$$

which leads to (146) and also

$$s \leq \frac{1}{r(1+p)} = \nu. \tag{157}$$

With this, $U(-1; \nu, p, r_1, r_2) > 0$, $U_s(-1; \nu, p, r_1, r_2) < 0$, and $U_{ss}(-1; \nu, p, r_1, r_2) > 0$, from which we conclude there are no solutions $0 < s < \nu$.

Complex eigenvalues. Noting that $U(\lambda; s, p, r_1, r_2)$ is quadratic in λ, if the roots are complex they must be a complex conjugate pair. Setting the product of the eigenvalues $\lambda\overline{\lambda} = |\lambda|^2 = 1$ gives

$$s(4pr_1r_2s + 4r_1r_2s - pr_1 - pr_2 - 2r_1 - 2r_2) = 0 \tag{158}$$

Solving for $s > 0$ we get

$$s = \frac{(r_1 + r_2)(p+2)}{4r_2r_1(p+1)} \leq \frac{1 + \frac{1}{2}p}{r(1+p)} = \chi. \tag{159}$$

Since $\chi > \nu$, the bound is $s \leq \nu$.

3.12 Summary and time step restriction

The tightest restriction on s was obtained by (137) (3D) or (142) (2D). s is related to Δt by

$$s = \frac{\xi \Delta t^2 V_p^0 \mu}{m_p}. \tag{160}$$

Solving for Δt in dimension d we have

$$\begin{aligned} \Delta t &= \sqrt{\frac{s m_p}{\xi V_p^0 \mu}} && (161) \\ &\leq \sqrt{\frac{4m_p}{\xi V_p^0 \mu (2r-1)(3p+2)}} && (162) \\ &= \sqrt{\frac{m_p/V_p^0}{\xi (r - \frac{k}{2})(\mu + \frac{d}{2}\lambda)}}. && (163) \end{aligned}$$

Here, $\rho = m_p/V_p^0$ is the density and $\lambda + \frac{2}{d}\mu$ is the bulk modulus. $\xi = \frac{4}{\Delta x^2}$ for quadratic splines and $\xi = \frac{3}{\Delta x^2}$ for cubic splines. For CPIC, $r = 1$. For APIC with cubic splines, $r = 1.045606358$. For APIC with quadratic splines, $r = \frac{3}{2}$. For APIC and CPIC, $k = 1$. For PIC, $k = 0$.

3.13 Effects of variable time step sizes

The analysis of the single particle instability was performed under the implicit assumption that every time step would be of the same size. However, in practice it is common to run with varying time steps sizes. Conventional wisdom suggests that as long as the time step sizes are below some time step restriction the simulation will be stable for any combination of time step sizes. Although this has been our general experience with MPM simulations, we discovered that this is not the case when simulating isolated particles.

We encountered the unexpected stability behavior while running numerical tests with CPIC, which is perhaps fortuitous since this case is much simpler to analyze. Numerically, the failure mode had the form $\mathbf{A}^n = a^n\mathbf{I}$ and $\mathbf{E}^n = e^n\mathbf{I}$, where a^n and e^n are scalars. That is, the unstable mode was purely a compression-pressure feedback loop. With these observations, we can reduce the investigation to a two-dof system.

Assuming an isotropic constitutive model, $\mathbf{M} : \mathbf{E}^n = (3\lambda + 2\mu)e^n\mathbf{I}$. Letting $m = \frac{\xi V_p^0}{m_p}(3\lambda + 2\mu)$ and recalling that $\mathbf{R} = \mathbf{I}$ for CPIC we have

$$\begin{aligned}
\mathbf{A}^{n+1} &= \mathbf{A}^n - \frac{\xi \Delta t V_p^0}{m_p}\mathbf{M} : \mathbf{E}^n && (164)\\
&= a^n\mathbf{I} - \frac{\xi \Delta t V_p^0}{m_p}(3\lambda + 2\mu)e^n\mathbf{I} && (165)\\
&= a^n\mathbf{I} - \Delta t m e^n\mathbf{I} && (166)\\
\mathbf{E}^{n+1} &= \Delta t\mathbf{A}^n - \frac{\xi \Delta t^2 V_p^0}{m_p}(\mathbf{M} : \mathbf{E}^n)\mathbf{R} + \mathbf{E}^n && (167)\\
&= \Delta t a^n\mathbf{I} - \Delta t^2 m e^n\mathbf{I} + e^n\mathbf{I} && (168)
\end{aligned}$$

We can express this as a matrix

$$\begin{pmatrix} a^{n+1} \\ e^{n+1} \end{pmatrix} = \begin{pmatrix} 1 & -\Delta t m \\ \Delta t & 1 - \Delta t^2 m \end{pmatrix} \begin{pmatrix} a^n \\ e^n \end{pmatrix}. \tag{169}$$

If we take two steps with different time steps Δt_0 and Δt_1 then

$$\begin{pmatrix} a^{n+2} \\ e^{n+2} \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & -\Delta t_1 m \\ \Delta t_1 & 1 - \Delta t_1^2 m \end{pmatrix} \begin{pmatrix} 1 & -\Delta t_0 m \\ \Delta t_0 & 1 - \Delta t_0^2 m \end{pmatrix}}_{\mathbf{B}} \begin{pmatrix} a^n \\ e^n \end{pmatrix}. \tag{170}$$

If we were to alternate between these two time steps sizes, then the eigenvalues of $\mathbf{B}$ would determine the stability. Observe that each of the 2×2 matrices has determinant 1, so that $\det(\mathbf{B}) = 1$. The product of the eigenvalues is the determinant and thus also 1. If the eigenvalues are complex, they occur as a complex conjugate pair with absolute value 1, which is stable. If the eigenvalues are real (and not both equal to 1 or -1), then one of those eigenvalues will have magnitude larger than 1, making the simulation unstable. The case we are in depends on $\operatorname{tr}(\mathbf{B})$, which is the negative sum of the eigenvalues. In particular, the simulation is stable only if $|\operatorname{tr}(\mathbf{B})| \le 2$. First, consider the positive direction.

$$\begin{aligned}
&2 \ge \operatorname{tr}(\mathbf{B}) && (171)\\
&2 \ge \Delta t_0^2 \Delta t_1^2 m^2 - \Delta t_0^2 m - 2\Delta t_0 \Delta t_1 m - \Delta t_1^2 m + 2 && (172)\\
&0 \ge \Delta t_0^2 \Delta t_1^2 m^2 - \Delta t_0^2 m - 2\Delta t_0 \Delta t_1 m - \Delta t_1^2 m && (173)\\
&0 \ge \Delta t_0^2 \Delta t_1^2 m - (\Delta t_0 + \Delta t_1)^2 && (174)
\end{aligned}$$

Assume $\Delta t_0 \le \Delta t_1$ and let $\Delta t_0 = k\Delta t_1$ with $0 < k \le 1$. Then,

$$\begin{aligned}
0 &\ge k^2 \Delta t_1^4 m - (k\Delta t_1 + \Delta t_1)^2 && (175)\\
0 &\ge k^2 \Delta t_1^2 m - (k + 1)^2 && (176)\\
\Delta t_1 &\le \frac{k + 1}{k\sqrt{m}} && (177)
\end{aligned}$$

The tightest bound occurs when $k = 1$, which is simply the case when the time steps are all the same. Next, consider the negative direction.

$$\begin{aligned}
-2 &\le \operatorname{tr}(\mathbf{B}) && (178)\\
0 &\le \Delta t_0^2 \Delta t_1^2 m^2 - \Delta t_0^2 m - 2\Delta t_0 \Delta t_1 m - \Delta t_1^2 m + 4 && (179)\\
0 &\le k^2 \Delta t_1^2 \Delta t_1^2 m^2 - k^2 \Delta t_1^2 m - 2k\Delta t_1 \Delta t_1 m - \Delta t_1^2 m + 4 && (180)
\end{aligned}$$

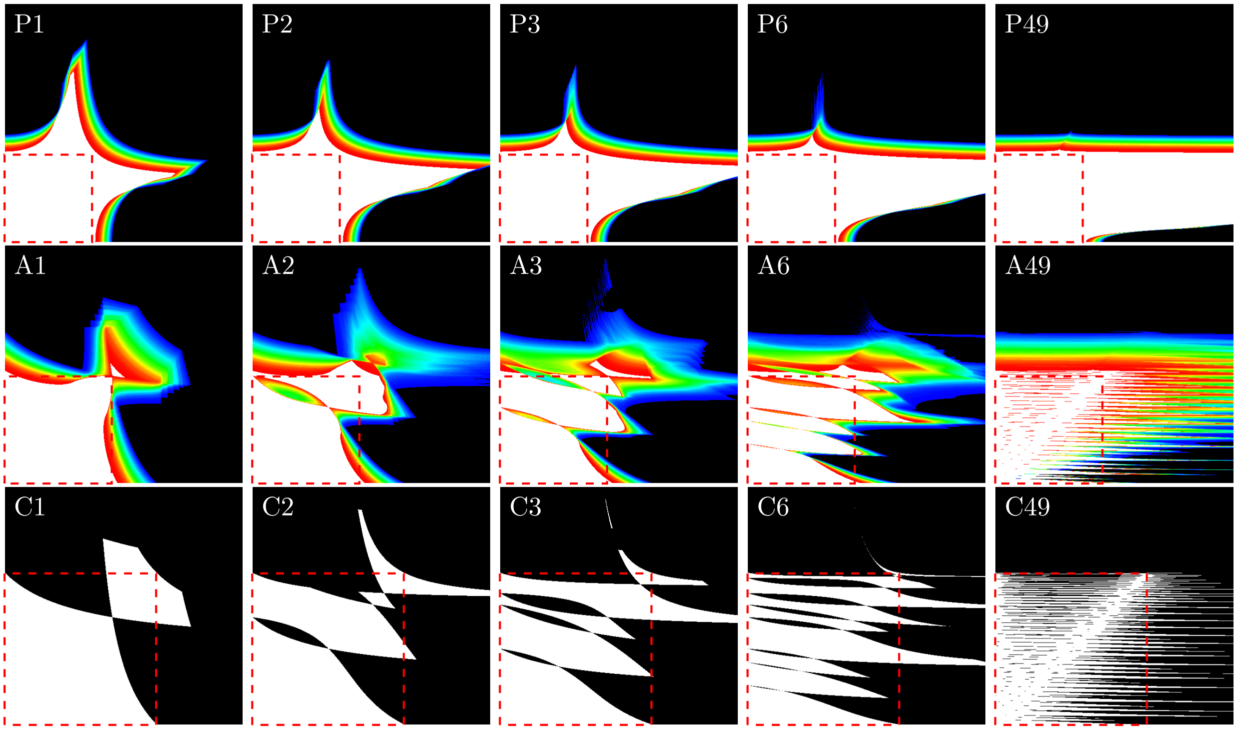

Figure 2: Destabilizing effects of cycling between two time step sizes on stability with one particle. Δt_0 is the x axis and Δt_1 is the y axis. The dashed box indicates the stable region based on constant time steps sizes. Figures are label with a letter (P=PIC, A=APIC, C=CPIC) and a number (1, 2, 3, 6, 49). The number indicates the number of times that Δt_1 is repeated. For example, A3 was run with APIC and time step sizes $\Delta t_0, \Delta t_1, \Delta t_1, \Delta t_1, \Delta t_0, \Delta t_1, \Delta t_1, \Delta t_1, \ldots$. For images A-C, time steps alternate: $\Delta t_0, \Delta t_1, \Delta t_0, \Delta t_1, \ldots$ with PIC (A), APIC (B), and CPIC (C). In (D), APIC is run with $\Delta t_0, \Delta t_1, \Delta t_1, \Delta t_0, \Delta t_1, \Delta t_1, \ldots$. In (E), Δt_0 is followed by 6 time steps at Δt_1 using APIC. In all cases, quadratic splines are used in 3D. All plots are on the same scale. White is stable, and black is unstable. If stability is particle position dependent, colors indicate the likelihood of any particular position being stable, with red being likely stable and blue being likely unstable. The stability region can become quite complex with unstable time step sequences scattered throughout the predicted stable region.

This inequality fails when

$$\frac{\sqrt{2((k+1)^2-(k-1)\sqrt{k^2+6k+1})}}{2k\sqrt{m}} < \Delta t_1 < \frac{\sqrt{2((k+1)^2+(k-1)\sqrt{k^2+6k+1})}}{2k\sqrt{m}}. \tag{181}$$

This gives a range of unstable time step sizes for any $0 < k < 1$. The bounds are equal when $k = 1$, so this type of instability does not occur for constant time step sizes. The minimum of the lower bound occurs when $k = 1 - \epsilon$, at which point the lower bound is $\Delta t_1 = \frac{\sqrt{2}}{\sqrt{m}} + O(\epsilon)$. *When two time step sizes are alternated, the simulation could be unstable at a time step that is nearly a factor of $\sqrt{2}$ lower than what was predicted in the constant time step analysis.* Figure 2 shows the stability landscape with two different time step sizes.

3.14 Conclusions about varying time step sizes

Optimizing the trace while obeying the single particle instability criterion shows that the largest magnitude eigenvalue of $\mathbf{B}$ occurs when $3\Delta t_0 = \Delta t_1 = \frac{2}{\sqrt{m}}$. With these, $\mathbf{B}$ has eigenvalues $-\frac{1}{3}$ and -3. Every two time steps, the state grows by a factor of three. This growth rate is fast enough for a particle that separates from the main bulk to explode before colliding with an obstacle or other particles.

One might be tempted to simply accept this factor of $\sqrt{2}$ reduction in the time step size and be done with the single particle instability. Unfortunately, this does not work. Cycling 3 nearly equal time step sizes admits unstable time step sizes that are smaller by a factor of about 2. For a sequence of 50 time steps, there are unstable time steps that are smaller by a factor of about 31. Of course, this extreme is quite pathological; the unstable time step window is small and the growth factor is barely over 1. But this does suggest that there is no time step bound for which a single-particle simulation with CPIC will be stable for any combination of time step sizes. Since the scenario above is also realized with APIC if the particle is at a grid node, the same conclusion applies to APIC. The analysis does not apply to PIC.

References

[1] Yunxin Sun, Tamar Shinar, and Craig Schroeder. Effective time step restrictions for explicit mpm simulation. In *Computer Graphics Forum*, volume 39, pages 55–67. Wiley Online Library, 2020.