Physics-Based Animation

The Meg I Scanline VFX

Last Video: Introduction and Variational Mechanics

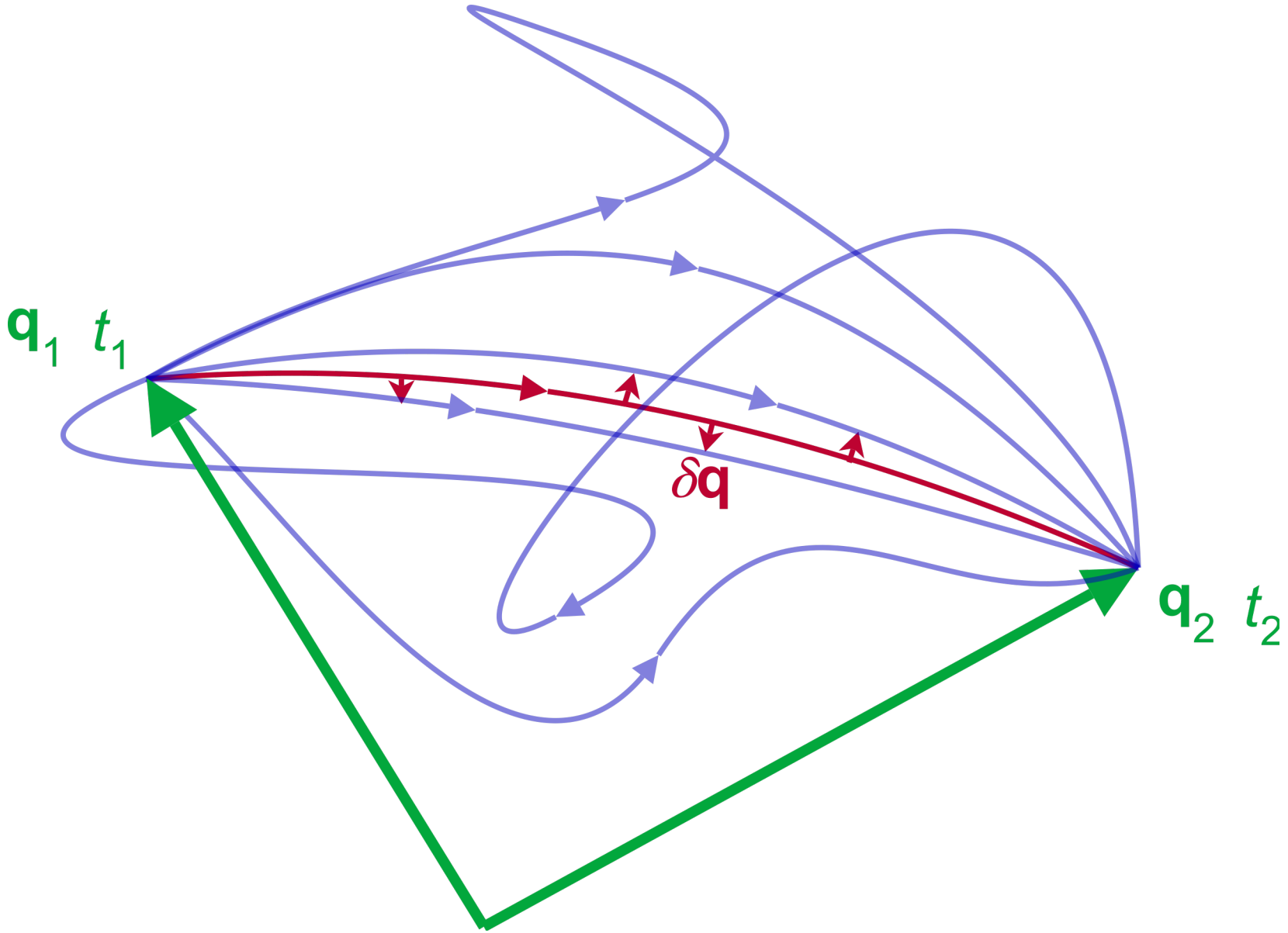
Last Video

$$m\ddot{q} = -kq$$

How do we solve this ?

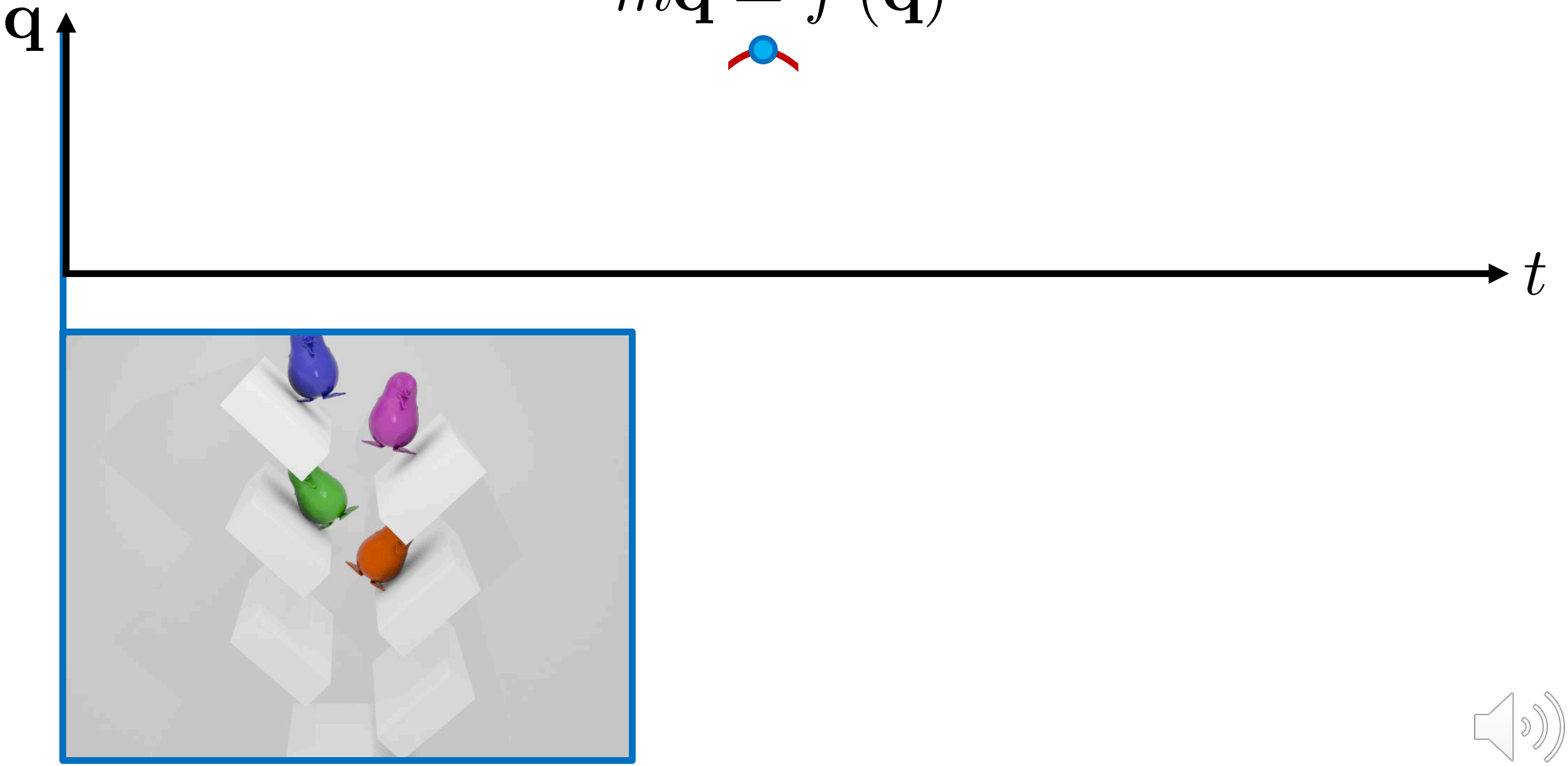
This Video: Time Integration

Generalized Coordinates



Time Integration

$$m\ddot{\mathbf{q}} = f(\mathbf{q})$$



Time Integration

Initial Conditions

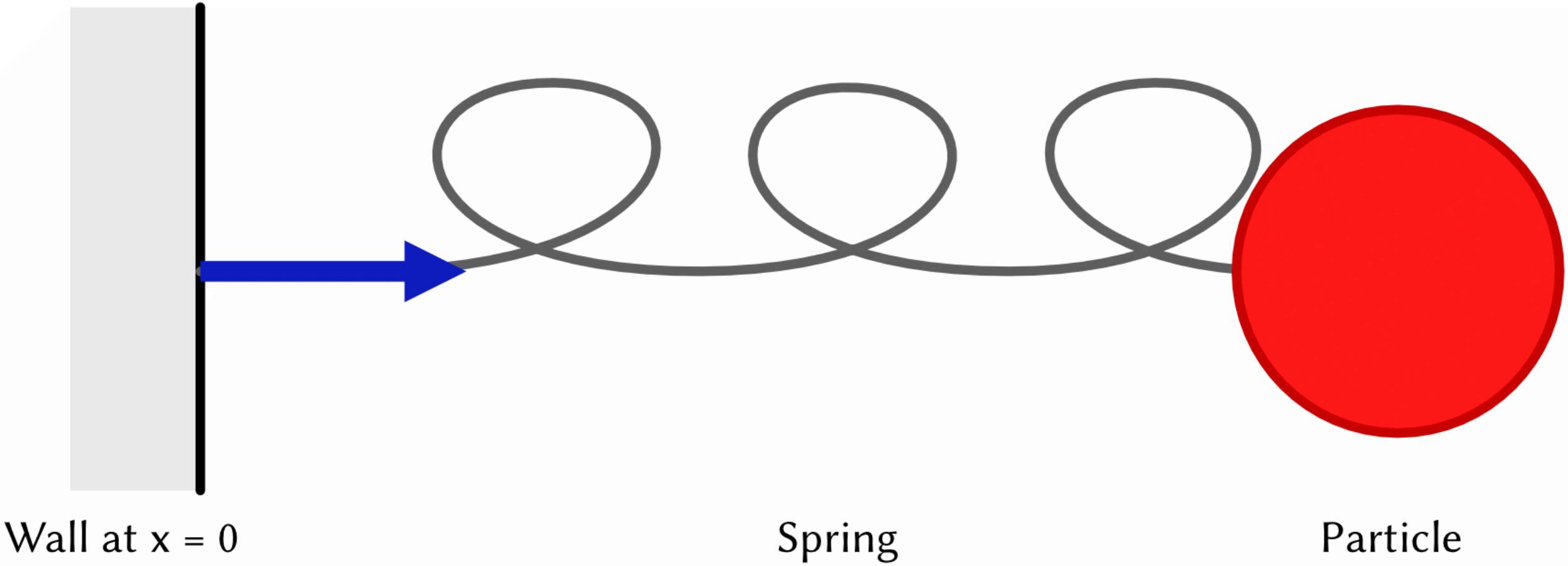
Input: $\ddot{\mathbf{q}} = \mathbf{f}\left(\mathbf{q}, \dot{\mathbf{q}}\right)$ $\quad q_0 = \mathbf{q}\left(t_0\right) \quad \dot{q}_0 = \dot{\mathbf{q}}\left(t_0\right)$

Ordinary Differential Equation (ODE)

Output: $\mathbf{q}^{t+1} = \mathbf{f}\left(\mathbf{q}^t, \mathbf{q}^{t+1}, \ldots, \dot{\mathbf{q}}^t, \dot{\mathbf{q}}^{t+1}, \ldots\right)$

Discrete Update Equation

Generalized Coordinates for Mass Spring System



$$q = x(t)$$
$$\dot{q} = v(t)$$

The Coupled First Order System

$$m\ddot{q} = -kq$$ Second-Order ODE

$$\dot{q} = v$$ Introduce velocity

$$m\dot{v} = -kq$$ First-Order ODE

Rewrite in matrix form

$$\underbrace{\begin{pmatrix} m & 0 \\ 0 & 1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\frac{d}{dt}\begin{pmatrix} v \\ q \end{pmatrix}}_{\dot{\mathbf{y}}} = \overbrace{\begin{pmatrix} 0 & -k \\ 1 & 0 \end{pmatrix} \underbrace{\begin{pmatrix} v \\ q \end{pmatrix}}_{\mathbf{y}}}^{\mathbf{f}(\mathbf{y})}$$

The Coupled First Order System

Rewrite in matrix form

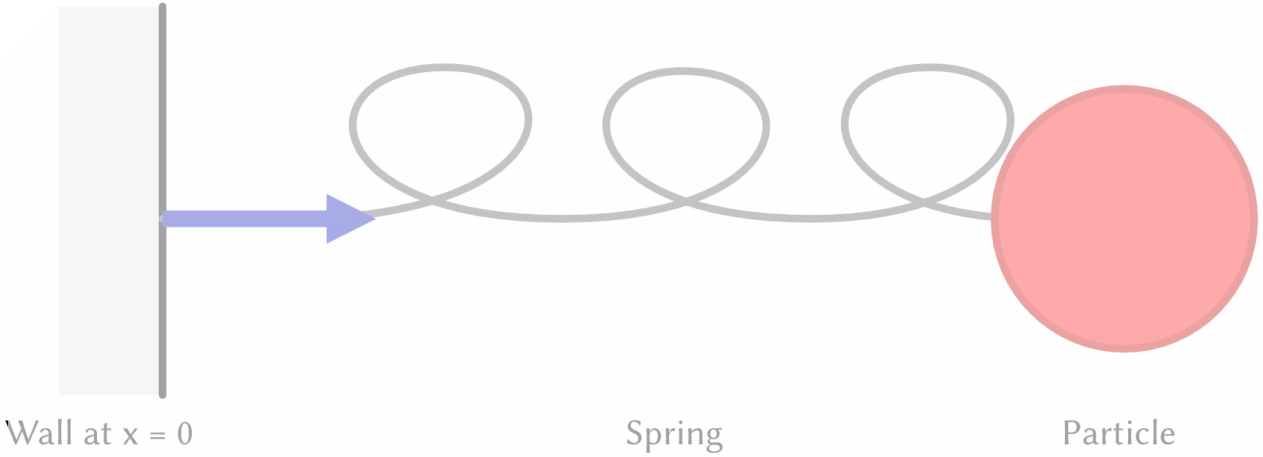
$$\underbrace{\begin{pmatrix} m & 0 \\ 0 & 1 \end{pmatrix}}_{\mathrm{A}} \underbrace{\frac{d}{dt}\begin{pmatrix} v \\ q \end{pmatrix}}_{\dot{\mathbf{y}}} = \overbrace{\begin{pmatrix} 0 & -k \\ 1 & 0 \end{pmatrix} \underbrace{\begin{pmatrix} v \\ q \end{pmatrix}}_{\mathbf{y}}}^{\mathbf{f}(\mathbf{y})}$$

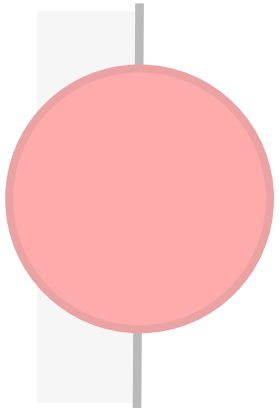
1st Order ODE

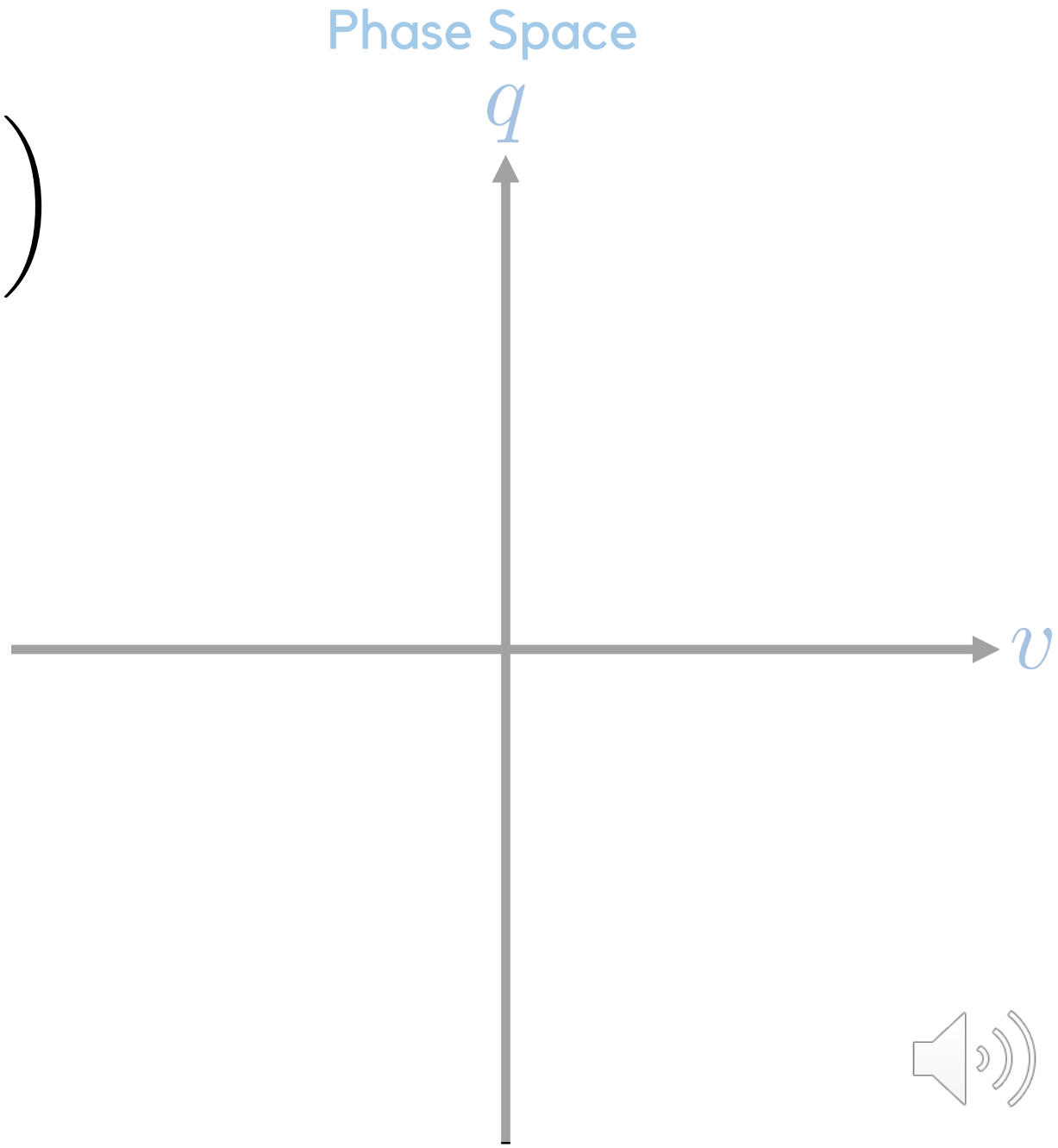
$$\mathrm{A}\dot{\mathbf{y}} = \mathbf{f}(\mathbf{y})$$

The Phase Space

$$\begin{pmatrix} m & 0 \\ 0 & 1 \end{pmatrix} \frac{d}{dt} \begin{pmatrix} v \\ q \end{pmatrix} = \begin{pmatrix} 0 & -k \\ 1 & 0 \end{pmatrix} \begin{pmatrix} v \\ q \end{pmatrix}$$

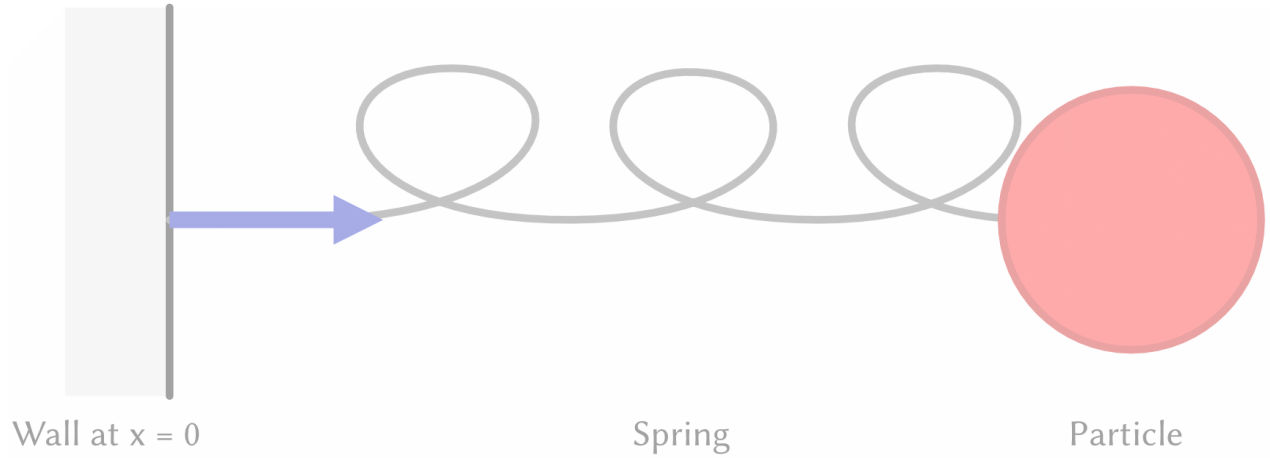


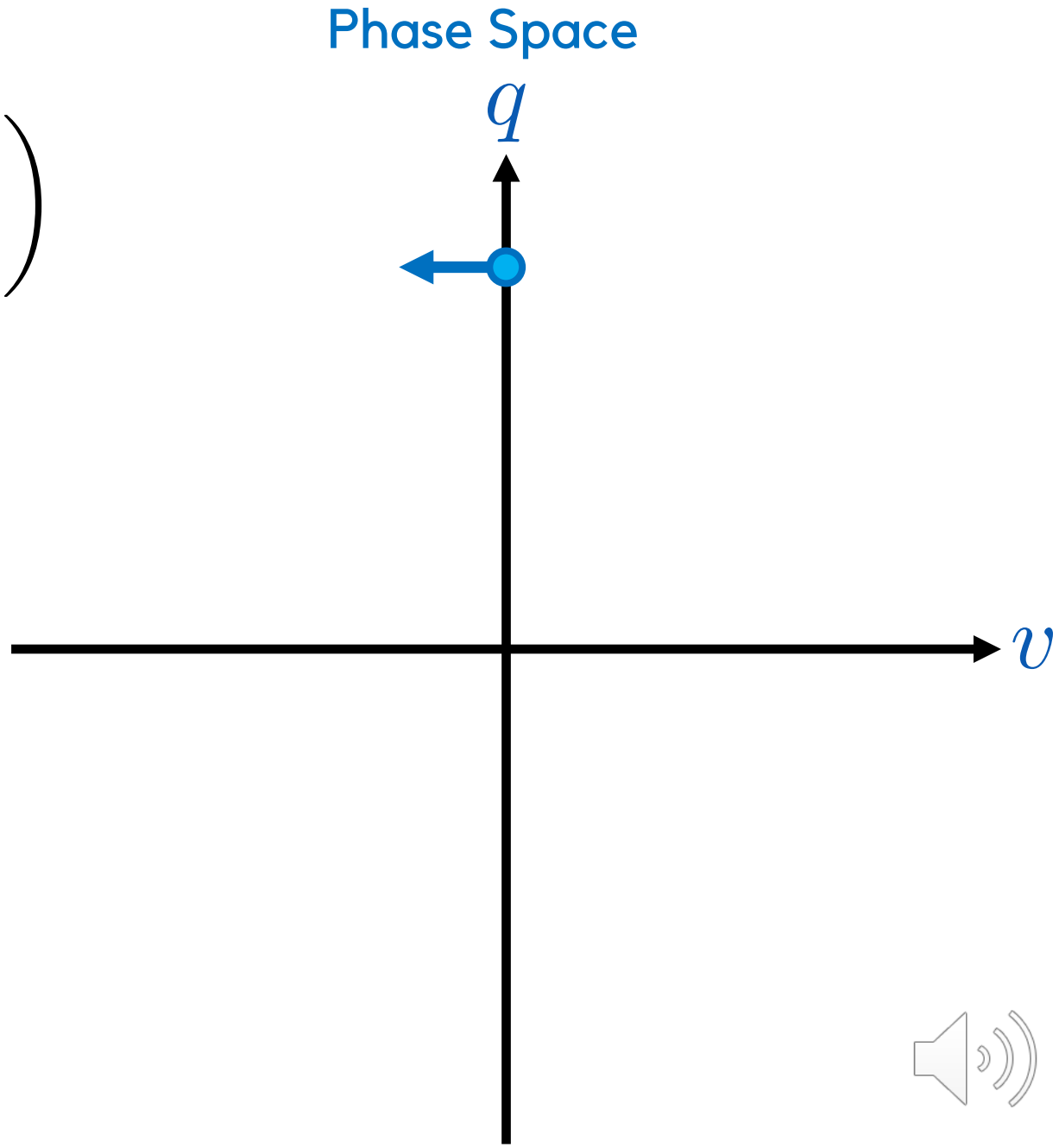


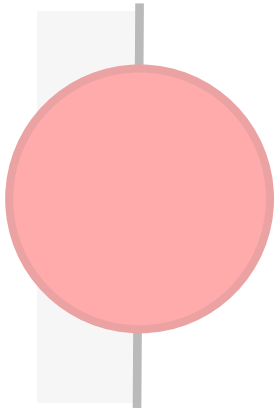


The Phase Space

$$\begin{pmatrix} m & 0 \\ 0 & 1 \end{pmatrix} \frac{d}{dt} \begin{pmatrix} v \\ q \end{pmatrix} = \begin{pmatrix} 0 & -k \\ 1 & 0 \end{pmatrix} \begin{pmatrix} v \\ q \end{pmatrix}$$

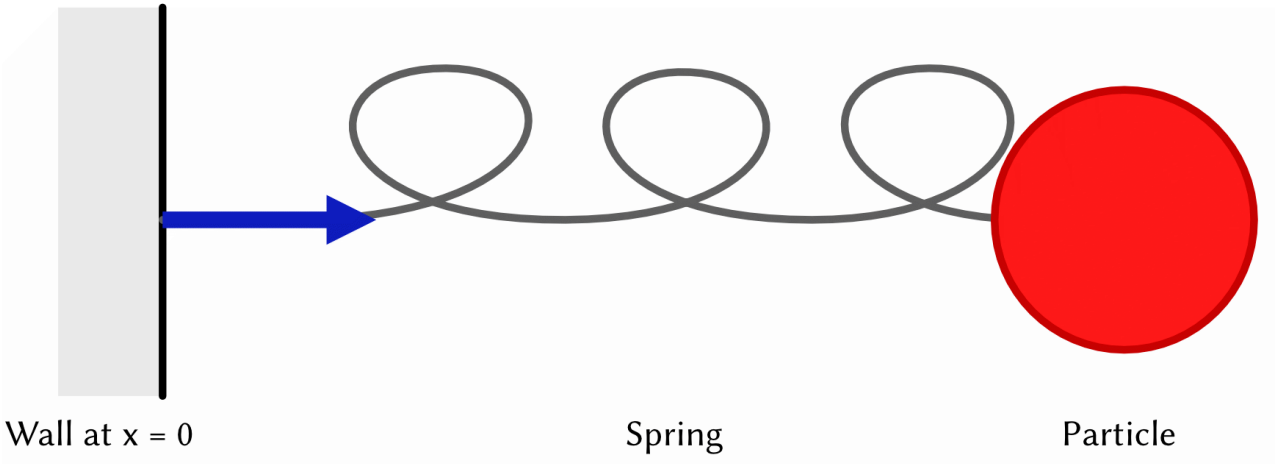


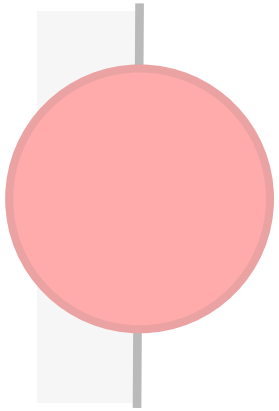


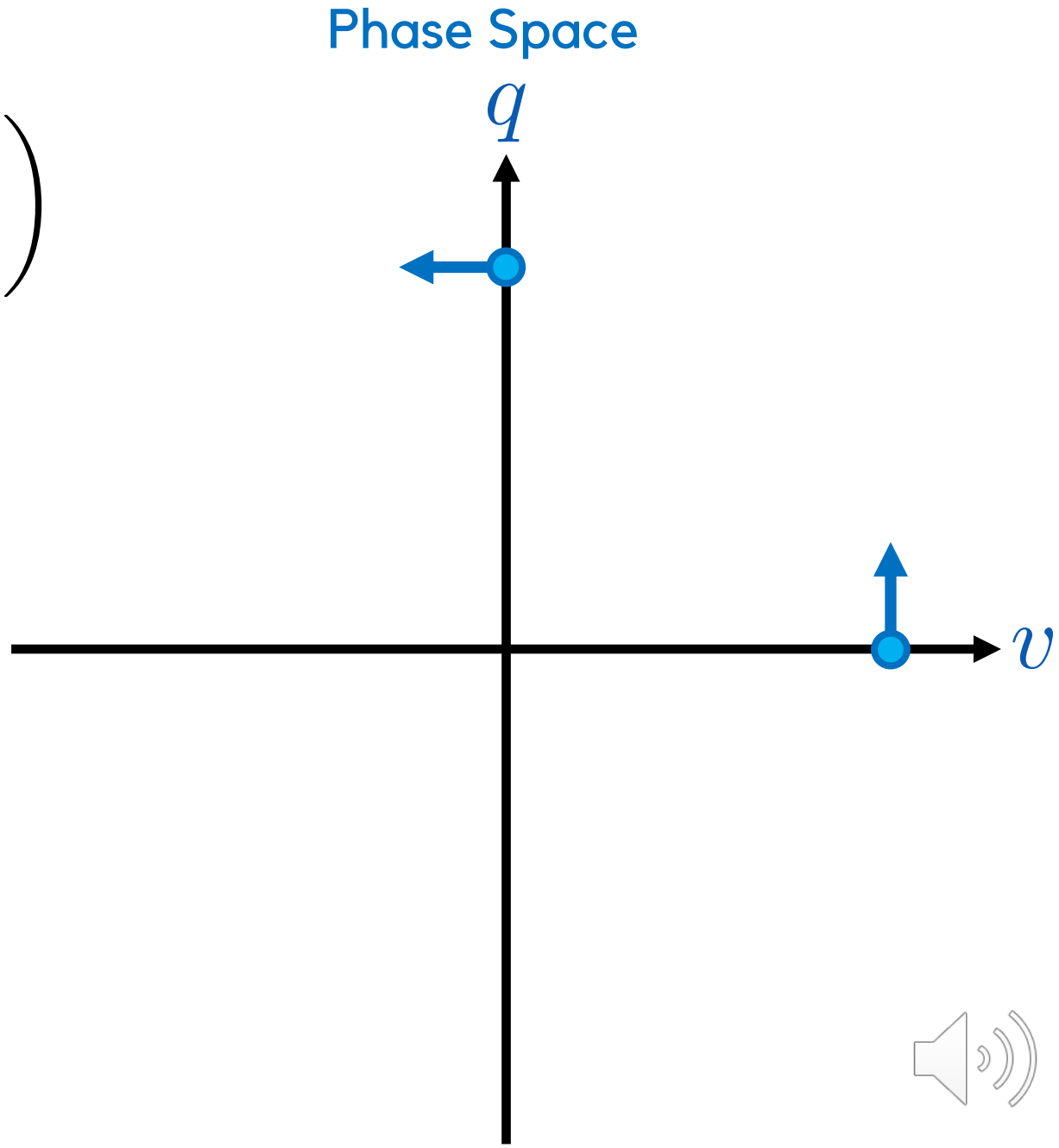


The Phase Space

$$\begin{pmatrix} m & 0 \\ 0 & 1 \end{pmatrix} \frac{d}{dt} \begin{pmatrix} v \\ q \end{pmatrix} = \begin{pmatrix} 0 & -k \\ 1 & 0 \end{pmatrix} \begin{pmatrix} v \\ q \end{pmatrix}$$

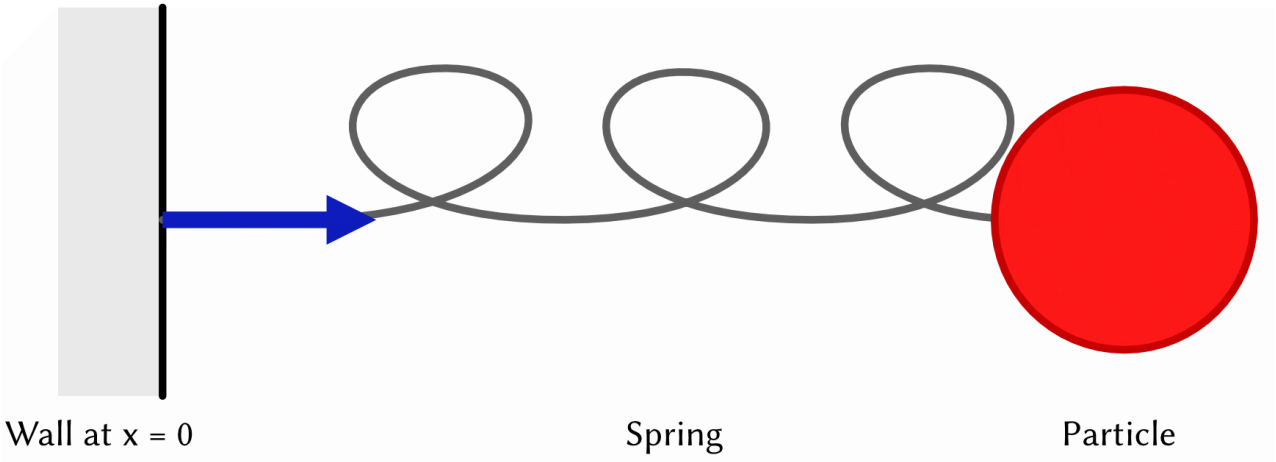


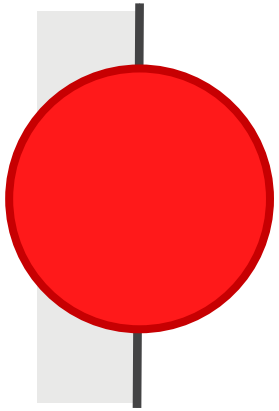


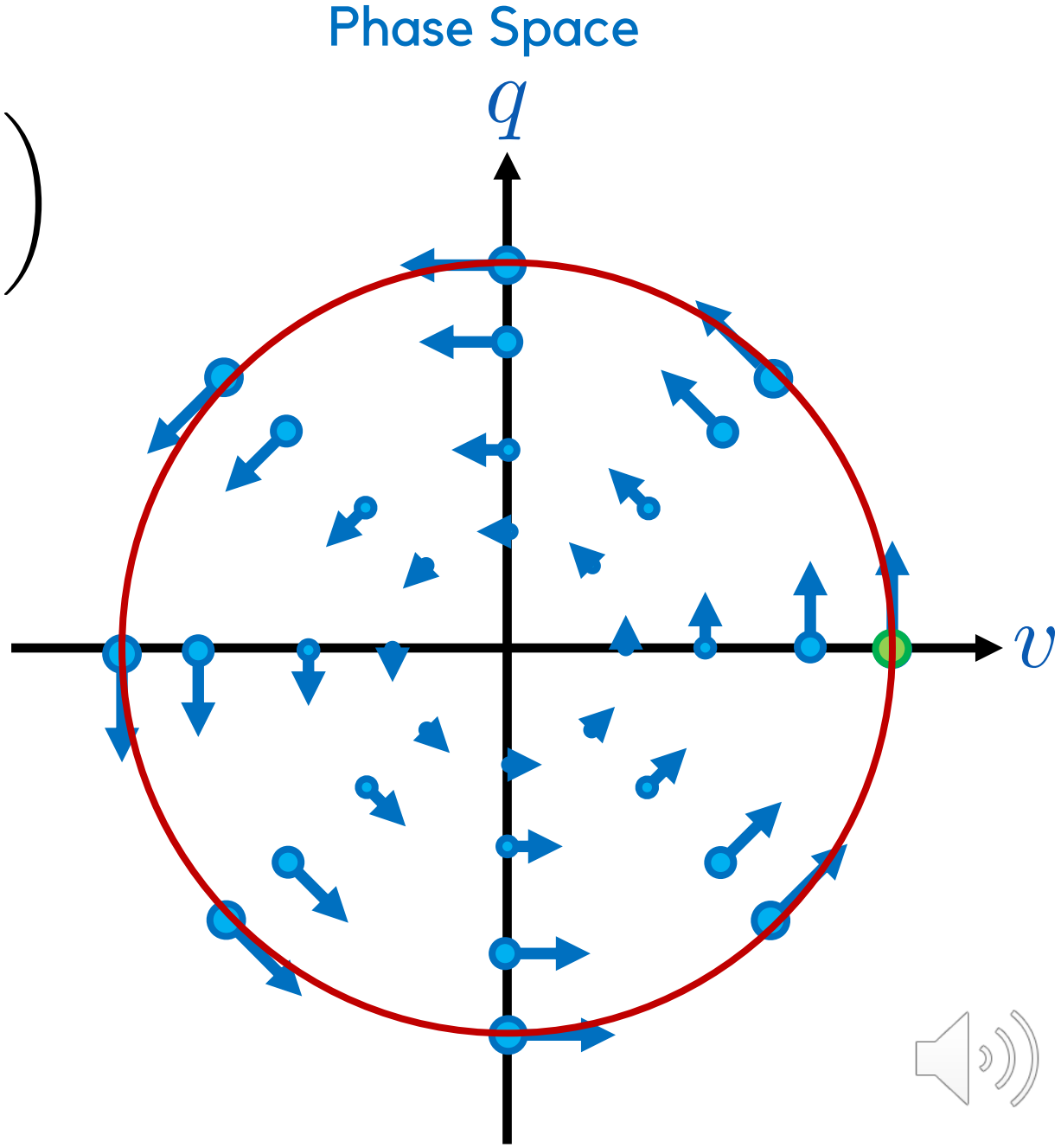


The Phase Space

$$\begin{pmatrix} m & 0 \\ 0 & 1 \end{pmatrix} \frac{d}{dt} \begin{pmatrix} v \\ q \end{pmatrix} = \begin{pmatrix} 0 & -k \\ 1 & 0 \end{pmatrix} \begin{pmatrix} v \\ q \end{pmatrix}$$







Types of Time Integration Algorithms

Explicit: Next time step can be computed entirely using values from the current time step or before

Implicit: Next time step is computed using values from the future!

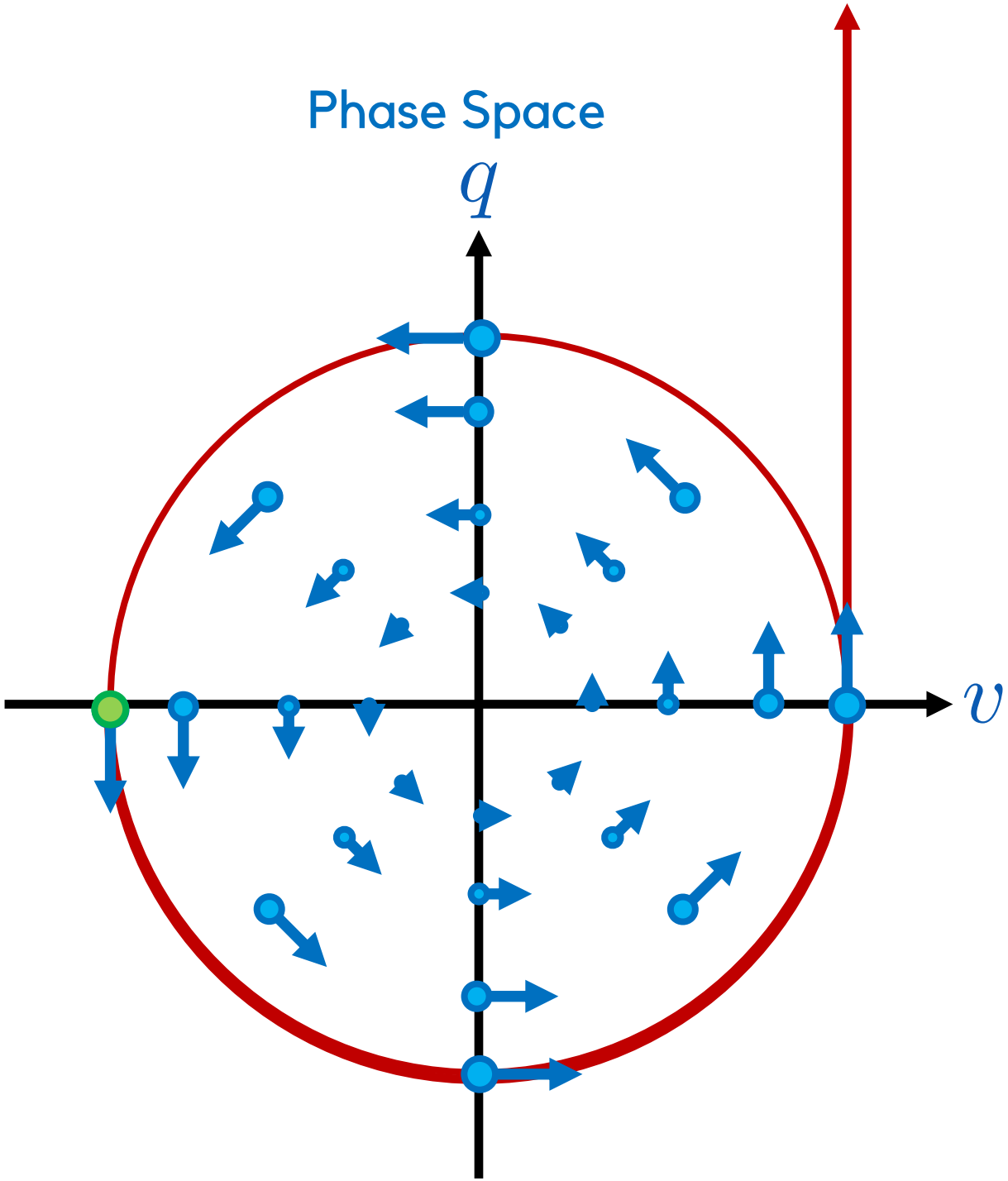
Concerns

1. Performance

2. Stability

3. Accuracy

Stability

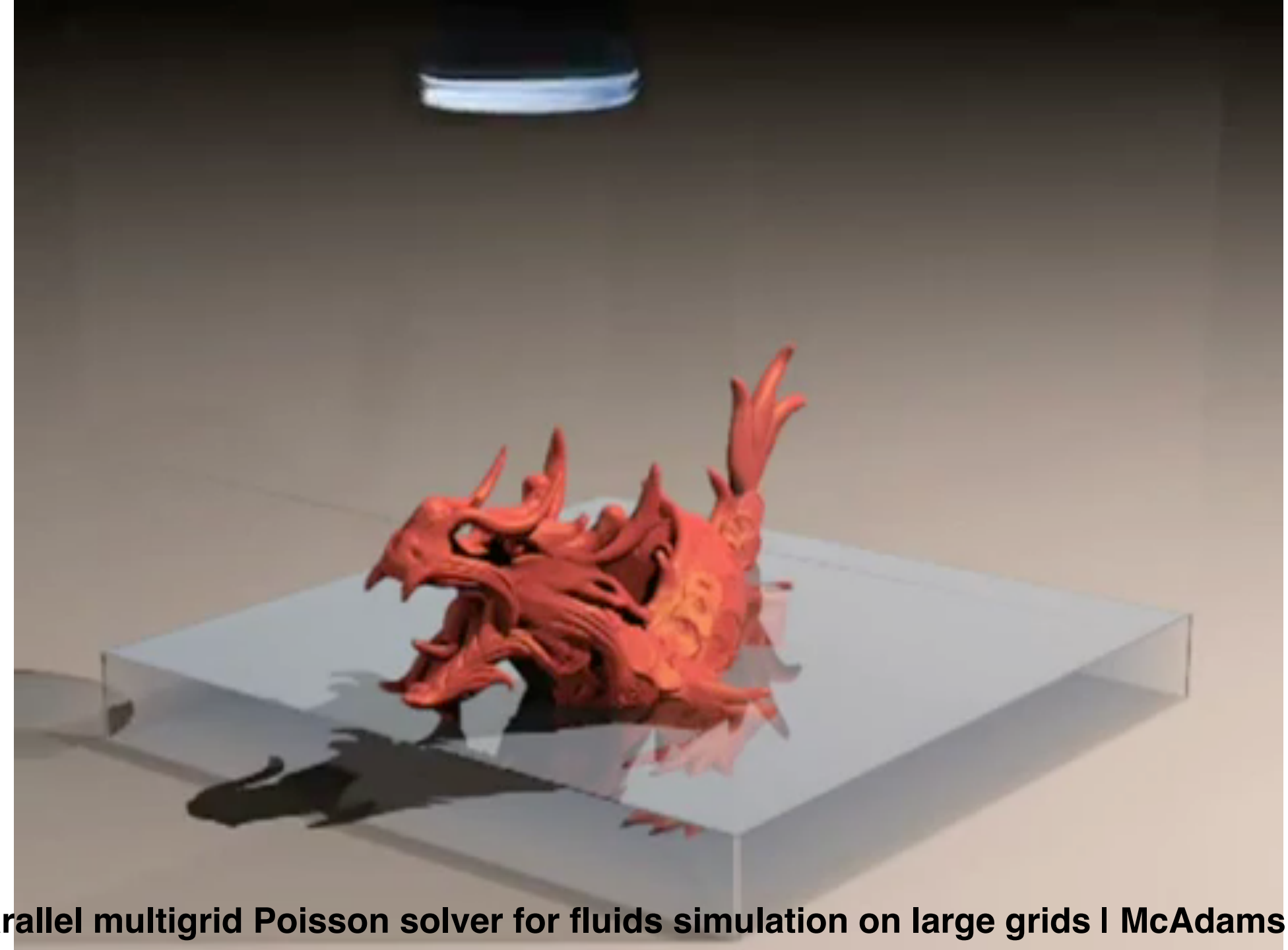


Concerns

1. Performance

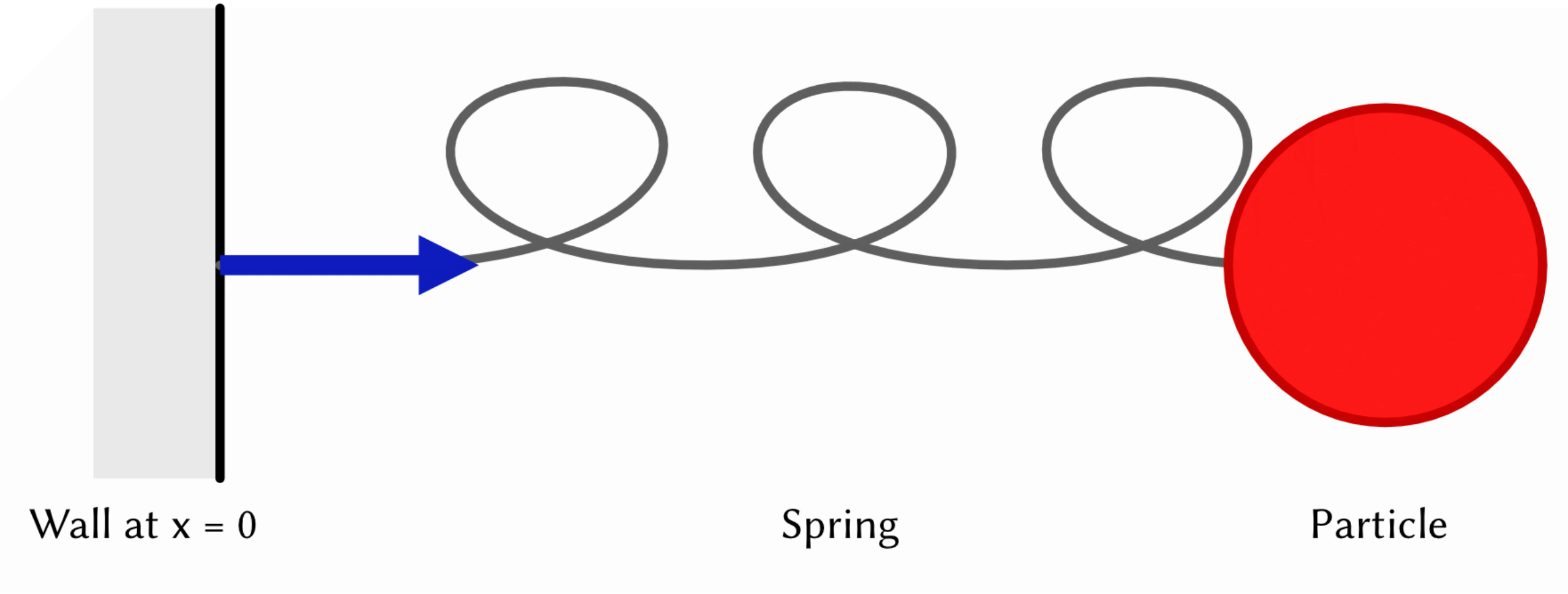
2. Stability

3. Accuracy



A parallel multigrid Poisson solver for fluids simulation on large grids | McAdams et al

Numerical Integration for our Mass-Spring System



Forward Euler Time Integration

$$\mathrm{A}\dot{\mathbf{y}} = \mathbf{f}\left(\mathbf{y}\right)$$

Replace time derivative with finite difference: $\dot{\mathbf{y}} \approx \frac{1}{\Delta t}\left(\mathbf{y}^{t+1} - \mathbf{y}^{t}\right)$

$$\mathrm{A}\frac{1}{\Delta t}\left(\mathbf{y}^{t+1} - \mathbf{y}^{t}\right) = \underbrace{\mathbf{f}\left(\mathbf{y}^{t}\right)}_{\text{Evaluated at current time step}}$$

$$\mathbf{y}^{t+1} = \mathbf{y}^{t} + \Delta t \mathrm{A}^{-1}\mathbf{f}\left(\mathbf{y}^{t}\right)$$

Forward Euler Time Integration

$$\mathrm{A}\dot{\mathbf{y}} = \mathbf{f}\left(\mathbf{y}\right)$$

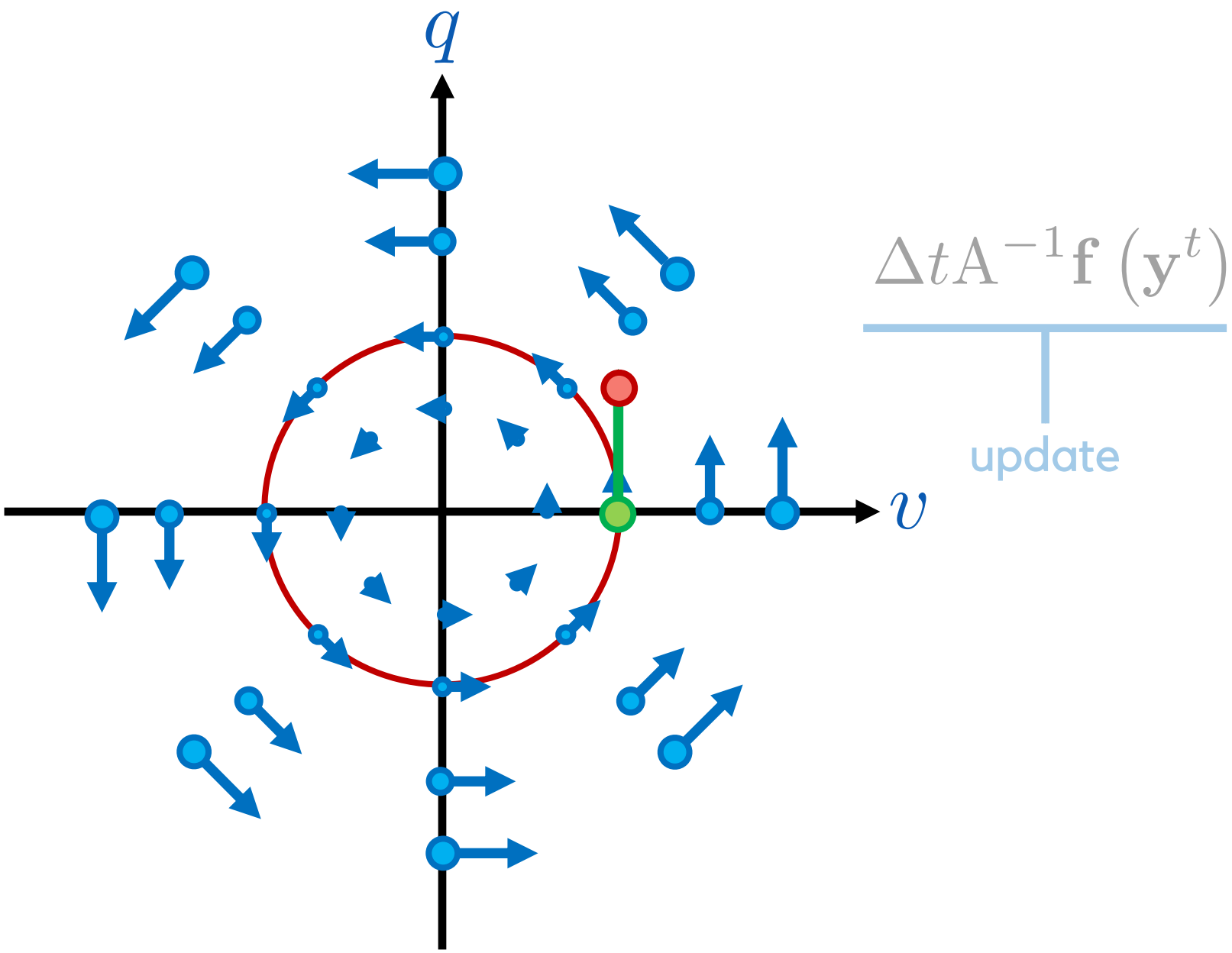
Replace time derivative with finite difference: $\dot{\mathbf{y}} \approx \frac{1}{\Delta t}\left(\mathbf{y}^{t+1} - \mathbf{y}^{t}\right)$

$$\mathrm{A}\frac{1}{\Delta t}\left(\mathbf{y}^{t+1} - \mathbf{y}^{t}\right) = \underbrace{\mathbf{f}\left(\mathbf{y}^{t}\right)}_{\text{Evaluated at current time step}}$$

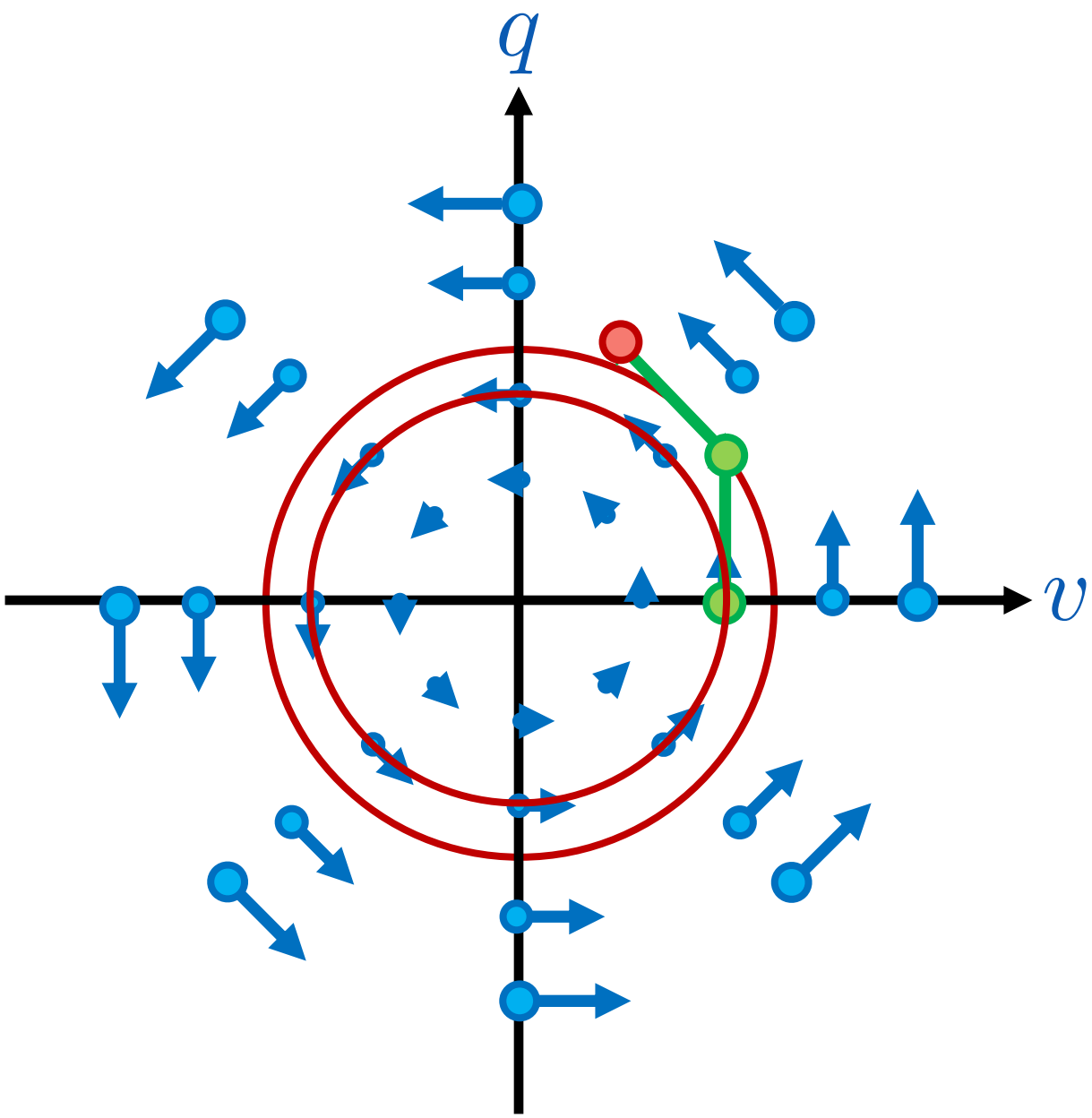
$$v^{t+1} = v^{t} - \Delta t \frac{k}{m} q^{t}$$

$$q^{t+1} = q^{t} + \Delta t v^{t}$$

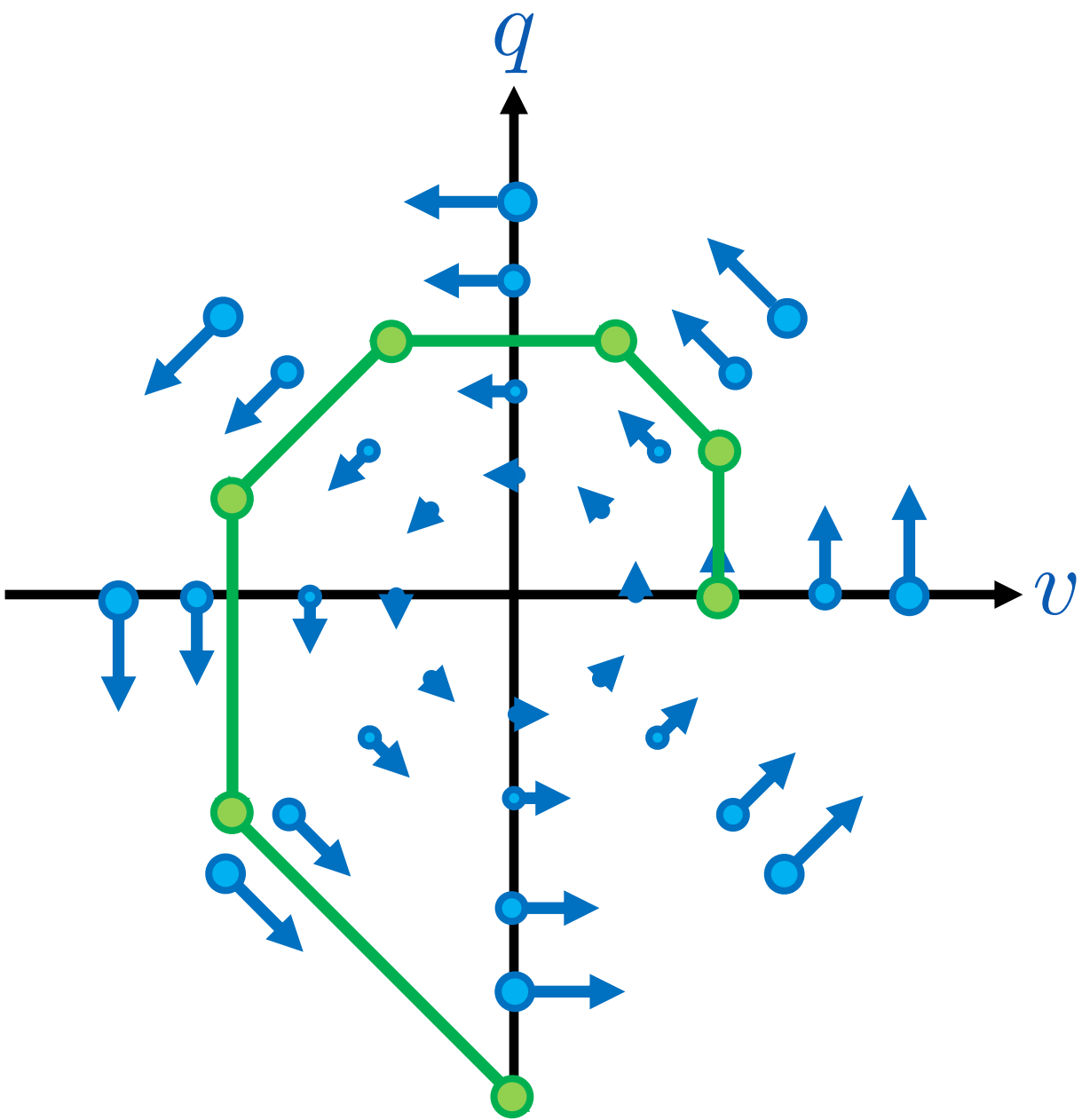
Phase Space Trajectory



Phase Space Trajectory



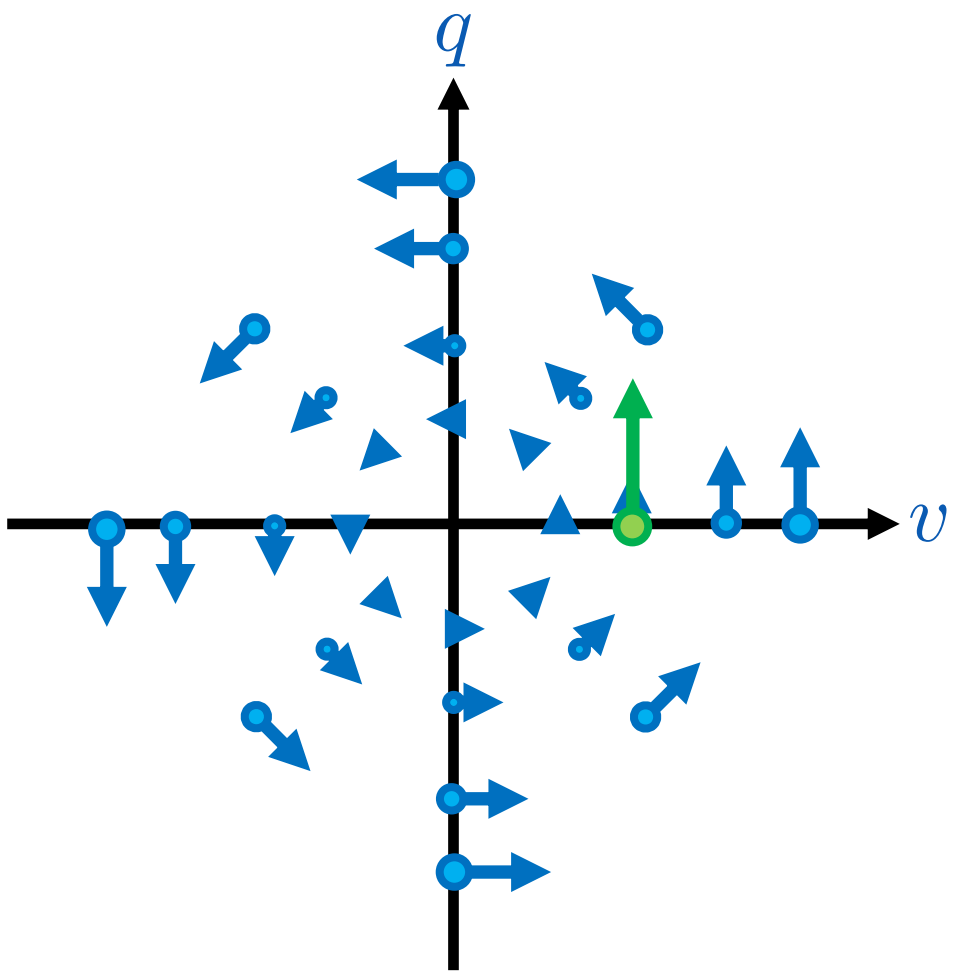
Phase Space Trajectory



Forward Euler

Integrate using current ”slope”

$$\mathbf{y}^{t+1} = \mathbf{y}^{t} + \Delta t \underbrace{\mathrm{A}^{-1}\mathbf{f}\left(\mathbf{y}^{t}\right)}_{\text{slope}}$$



Runge-Kutta Time Integration

Integrate using average ”slope”

Example using two slopes

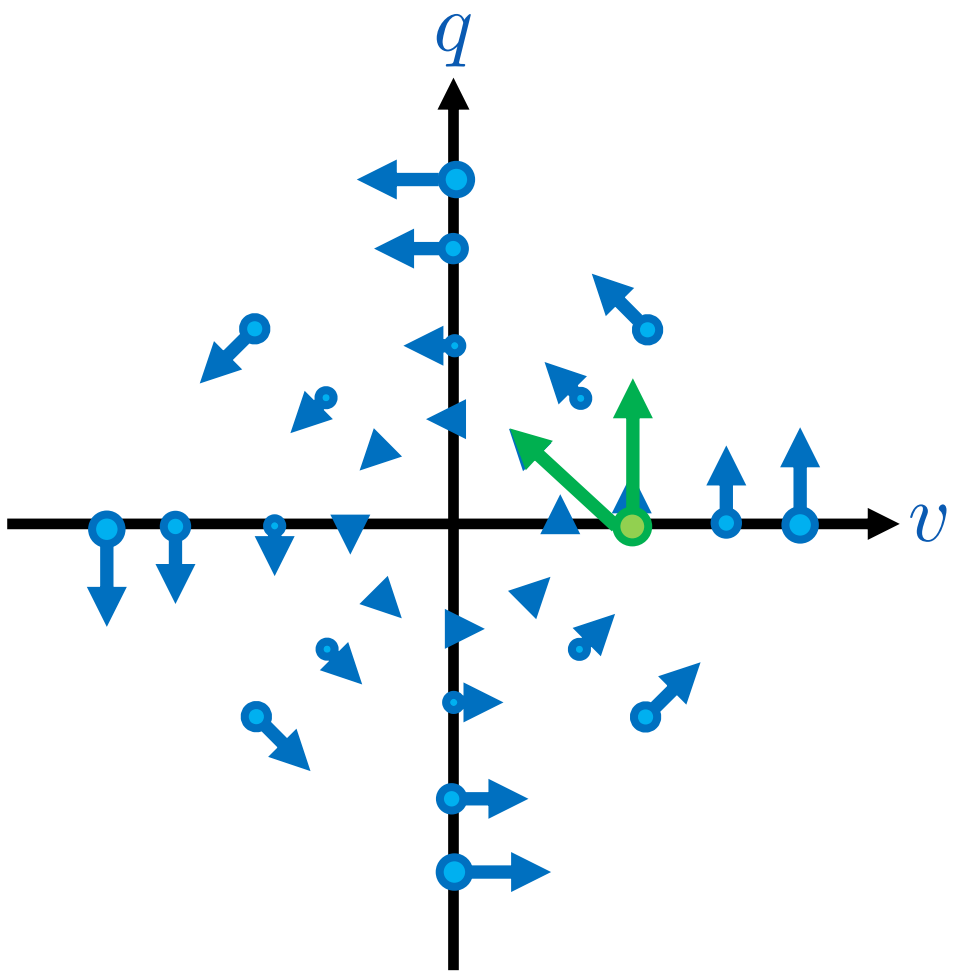
time coefficients

$$\mathbf{y}^{t+1} = \mathbf{y}^{t} + \Delta t \mathrm{A}^{-1}\left(\alpha \mathbf{f}\left(\mathbf{y}^{t+a}\right) + \beta \mathbf{f}\left(\tilde{\mathbf{y}}^{t+b}\right)\right)$$

slope 1 slope 2

Use Forward Euler to estimate $\mathbf{y}^{t+a}$

$$\tilde{\mathbf{y}}^{t+a} = \mathbf{y}^{t} + a\Delta t \mathrm{A}^{-1}\mathbf{f}\left(\mathbf{y}^{t}\right)$$



Runge-Kutta Time Integration

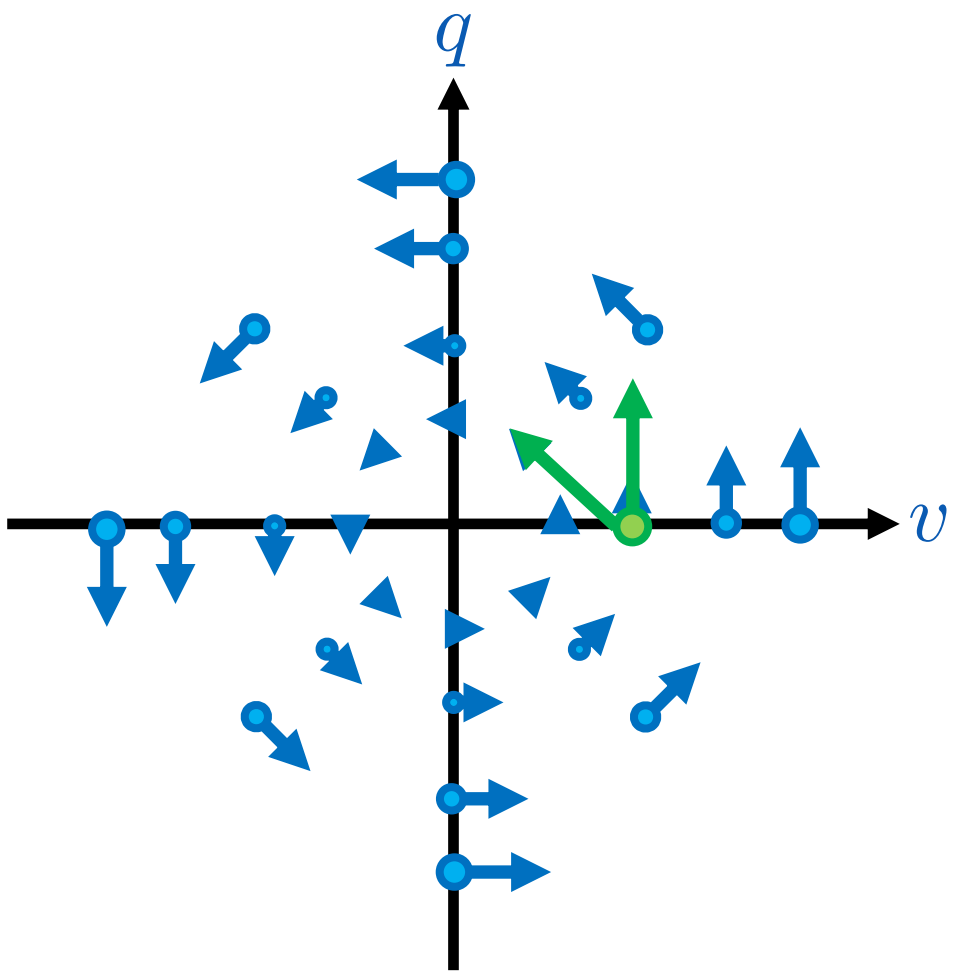
Integrate using average ”slope”

Example using two slopes

time coefficients

$$\mathbf{y}^{t+1} = \mathbf{y}^{t} + \Delta t \mathrm{A}^{-1}\left(\alpha \mathbf{f}\left(\mathbf{y}^{t+a}\right) + \beta \mathbf{f}\left(\tilde{\mathbf{y}}^{t+b}\right)\right)$$

averaging coefficients



Use Forward Euler to estimate $\mathbf{y}^{t+a}$

$$\tilde{\mathbf{y}}^{t+a} = \mathbf{y}^{t} + a\Delta t \mathrm{A}^{-1}\mathbf{f}\left(\mathbf{y}^{t}\right)$$

Runge-Kutta Time Integration

time coefficients

$$\mathbf{y}^{t+1} = \mathbf{y}^{t} + \Delta t \mathrm{A}^{-1} \left(\alpha \mathbf{f} \left(\mathbf{y}^{t+a} \right) + \beta \mathbf{f} \left(\tilde{\mathbf{y}}^{t+b} \right) \right)$$

averaging coefficients

with $a = 0 \;\; b = 1 \;\; \alpha = \beta = \dfrac{1}{2}$

Becomes Heun's Method

$$\mathbf{y}^{t+1} = \mathbf{y}^{t} + \frac{\Delta t}{2} \mathrm{A}^{-1} \left(\mathbf{f} \left(\mathbf{y}^{t} \right) + \mathbf{f} \left(\tilde{\mathbf{y}}^{t+1} \right) \right)$$

using $\tilde{\mathbf{y}}^{t+1} = \mathbf{y}^{t} + \Delta t \mathrm{A}^{-1} \mathbf{f} \left(\mathbf{y}^{t} \right)$

Runge-Kutta Time Integration

Rewrite in more compact form

$$\kappa_1 = \mathrm{A}^{-1}\mathbf{f}\left(\mathbf{y}^t\right)$$

$$\kappa_2 = \mathrm{A}^{-1}\mathbf{f}\left(\mathbf{y}^t + \Delta t \cdot \kappa_1\right)$$

$$\mathbf{y}^{t+1} = \mathbf{y}^t + \frac{\Delta t}{2}\left(\kappa_1 + \kappa_2\right)$$

Runge-Kutta Time Integration

Fourth-Order Runge Kutta

$$\kappa_1 = \mathrm{A}^{-1}\mathbf{f}\left(\mathbf{y}^t\right)$$

$$\kappa_2 = \mathrm{A}^{-1}\mathbf{f}\left(\mathbf{y}^t + \frac{\Delta t}{2} \cdot \kappa_1\right)$$

$$\kappa_3 = \mathrm{A}^{-1}\mathbf{f}\left(\mathbf{y}^t + \frac{\Delta t}{2} \cdot \kappa_2\right)$$

$$\kappa_4 = \mathrm{A}^{-1}\mathbf{f}\left(\mathbf{y}^t + \Delta t \cdot \kappa_3\right)$$

$$\mathbf{y}^{t+1} = \mathbf{y}^t + \frac{\Delta t}{6}\left(\kappa_1 + 2 \cdot \kappa_2 + 2 \cdot \kappa_3 + \kappa_4\right)$$

Backward Euler Time Integration

$$\mathrm{A}\dot{\mathbf{y}} = \mathbf{f}\left(\mathbf{y}\right)$$

Replace time derivative with finite difference:

$$\mathrm{A}\frac{1}{\Delta t}\left(\mathbf{y}^{t+1} - \mathbf{y}^{t}\right) = \underbrace{\mathbf{f}\left(\mathbf{y}^{t+1}\right)}_{\text{Evaluated at next time step}}$$

The Coupled First Order System

Rewrite in matrix form

$$\underbrace{\begin{pmatrix} m & 0 \\ 0 & 1 \end{pmatrix}}_{\mathrm{A}} \underbrace{\frac{d}{dt}\begin{pmatrix} v \\ q \end{pmatrix}}_{\dot{\mathbf{y}}} = \overbrace{\underbrace{\begin{pmatrix} 0 & -k \\ 1 & 0 \end{pmatrix}}_{\mathrm{B}} \underbrace{\begin{pmatrix} v \\ q \end{pmatrix}}_{\mathbf{y}}}^{\mathbf{f}(\mathbf{y})}$$

1st Order ODE

$$\mathrm{A}\dot{\mathbf{y}} = \mathrm{B}\mathbf{y}$$

Backward Euler Time Integration

$$\mathrm{A}\dot{\mathbf{y}} = \mathbf{f}\left(\mathbf{y}\right)$$

Replace time derivative with finite difference:

$$\mathrm{A}\frac{1}{\Delta t}\left(\mathbf{y}^{t+1} - \mathbf{y}^{t}\right) = \underbrace{\mathbf{f}\left(\mathbf{y}^{t+1}\right)}_{\text{Evaluated at next time step}}$$

Backward Euler Time Integration

$$\mathrm{A}\dot{\mathbf{y}} = \mathbf{f}\left(\mathbf{y}\right)$$

Replace time derivative with finite difference:

$$\mathrm{A}\frac{1}{\Delta t}\left(\mathbf{y}^{t+1} - \mathbf{y}^{t}\right) = \underbrace{\mathrm{B}\mathbf{y}^{t+1}}_{\text{Evaluated at next time step}}$$

$$\mathbf{y}^{t+1} = \mathbf{y}^{t} + \Delta t \mathrm{A}^{-1}\mathrm{B}\mathbf{y}^{t+1}$$

Backward Euler Time Integration

$$\mathrm{A}\dot{\mathbf{y}} = \mathbf{f}\left(\mathbf{y}\right)$$

Replace time derivative with finite difference:

$$\mathrm{A}\frac{1}{\Delta t}\left(\mathbf{y}^{t+1} - \mathbf{y}^{t}\right) = \underbrace{\mathrm{B}\mathbf{y}^{t+1}}_{\text{Evaluated at next time step}}$$

$$\left(\underbrace{\mathrm{I}}_{\text{identity matrix}} - \Delta t \mathrm{A}^{-1}\mathrm{B}\right)\mathbf{y}^{t+1} = \mathbf{y}^{t}$$

Backward Euler Time Integration

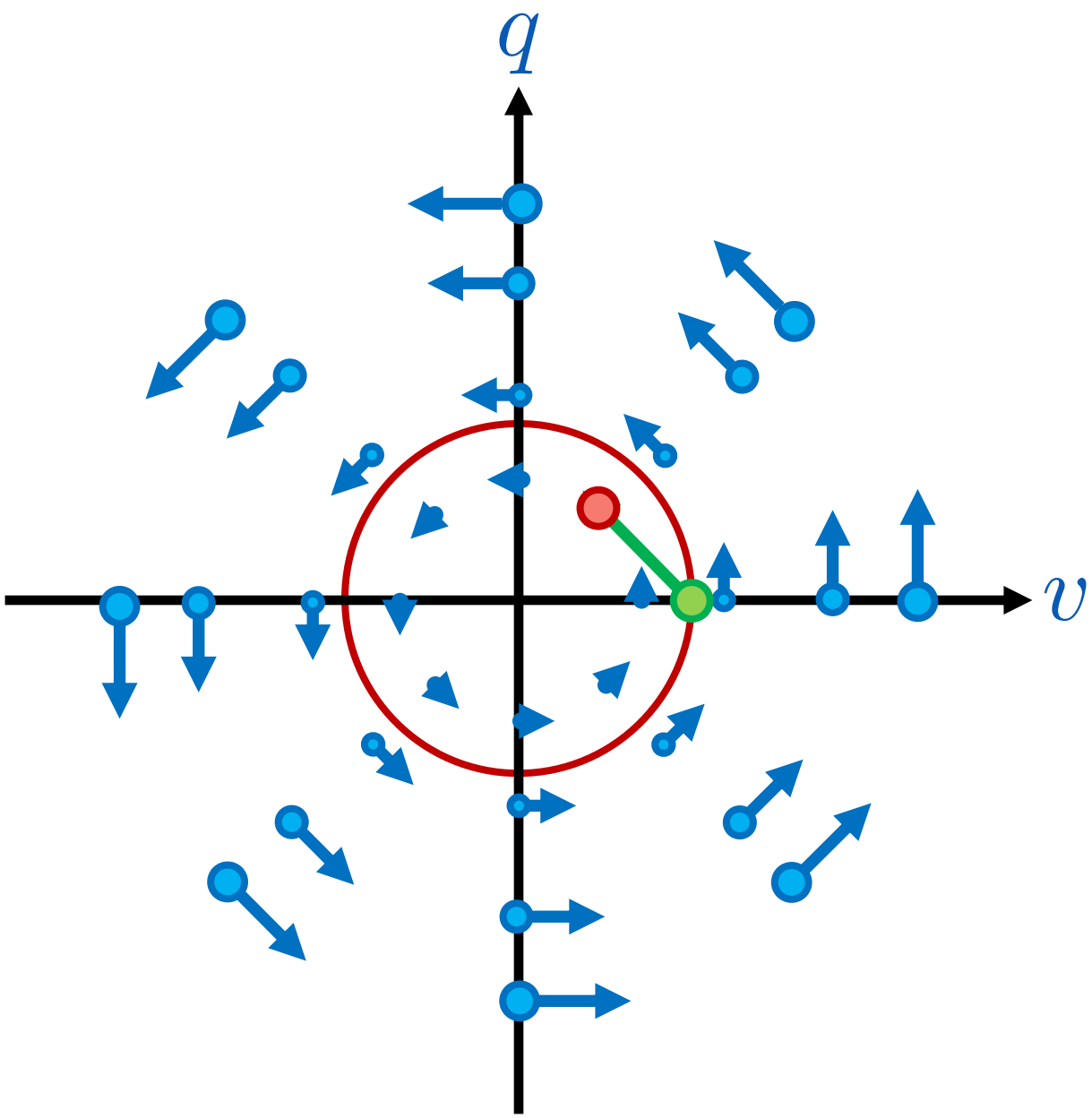
$$\mathrm{A}\dot{\mathbf{y}} = \mathbf{f}\left(\mathbf{y}\right)$$

Replace time derivative with finite difference:

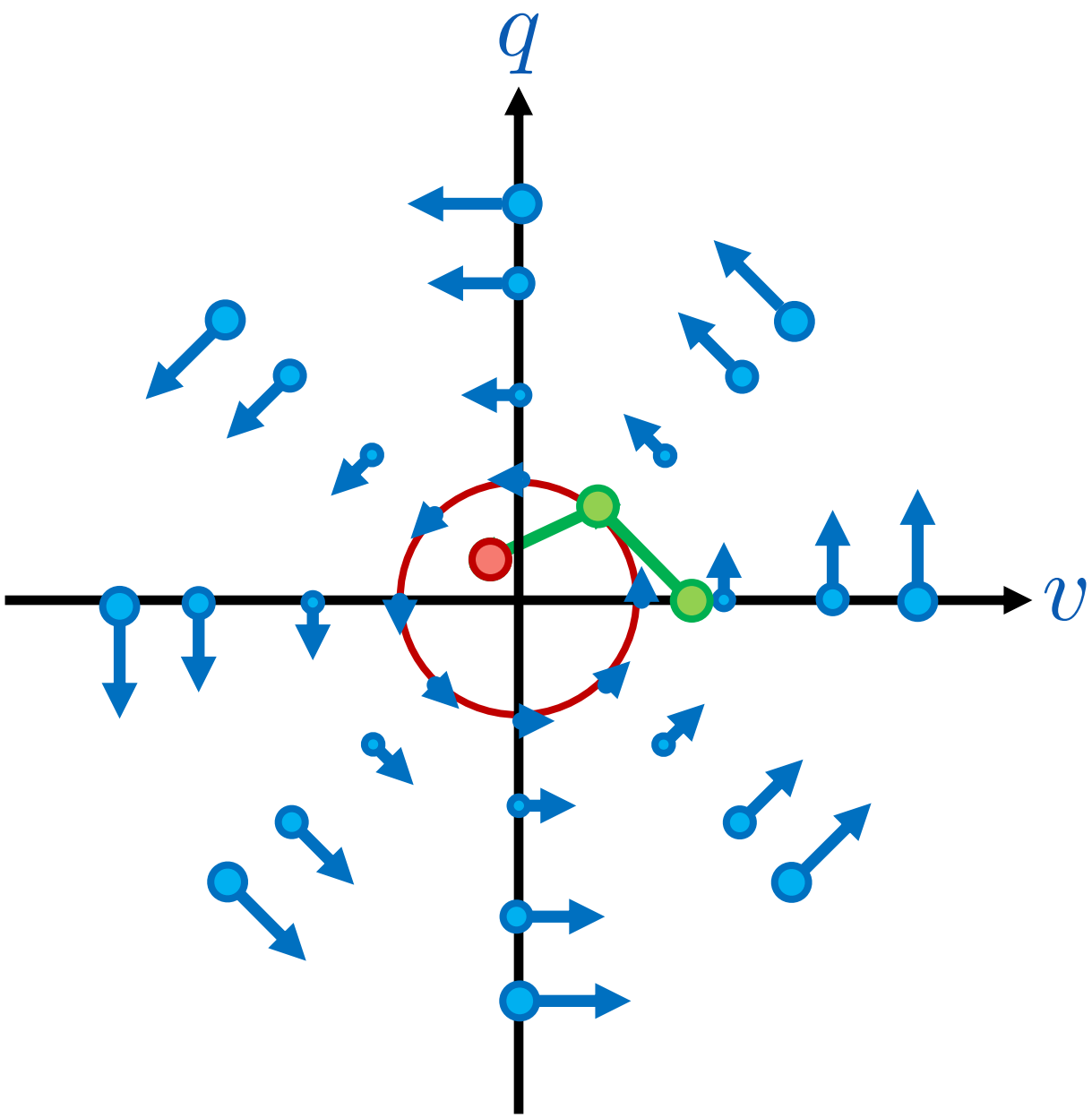
$$\mathrm{A}\frac{1}{\Delta t}\left(\mathbf{y}^{t+1} - \mathbf{y}^{t}\right) = \underbrace{\mathrm{B}\mathbf{y}^{t+1}}_{\text{Evaluated at next time step}}$$

$$\underbrace{\mathbf{y}^{t+1} - \mathbf{y}^{t}}_{\text{vector from between states}} = \overbrace{\Delta t \mathrm{A}^{-1}\mathrm{B}\mathbf{y}^{t+1}}^{\text{slope at } t+1}$$

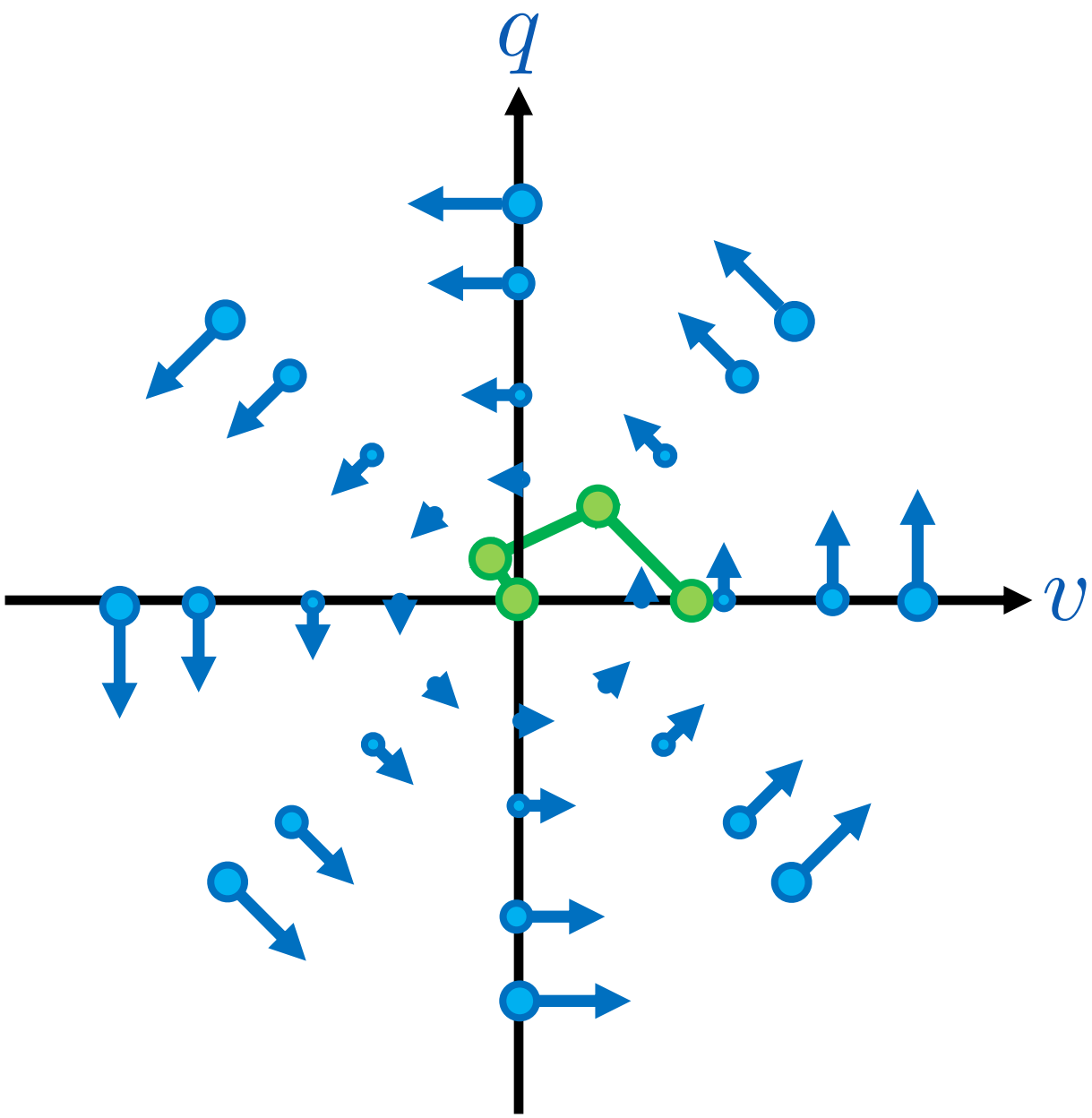
Phase Space Trajectory



Phase Space Trajectory



Phase Space Trajectory



Backward Euler Time Integration

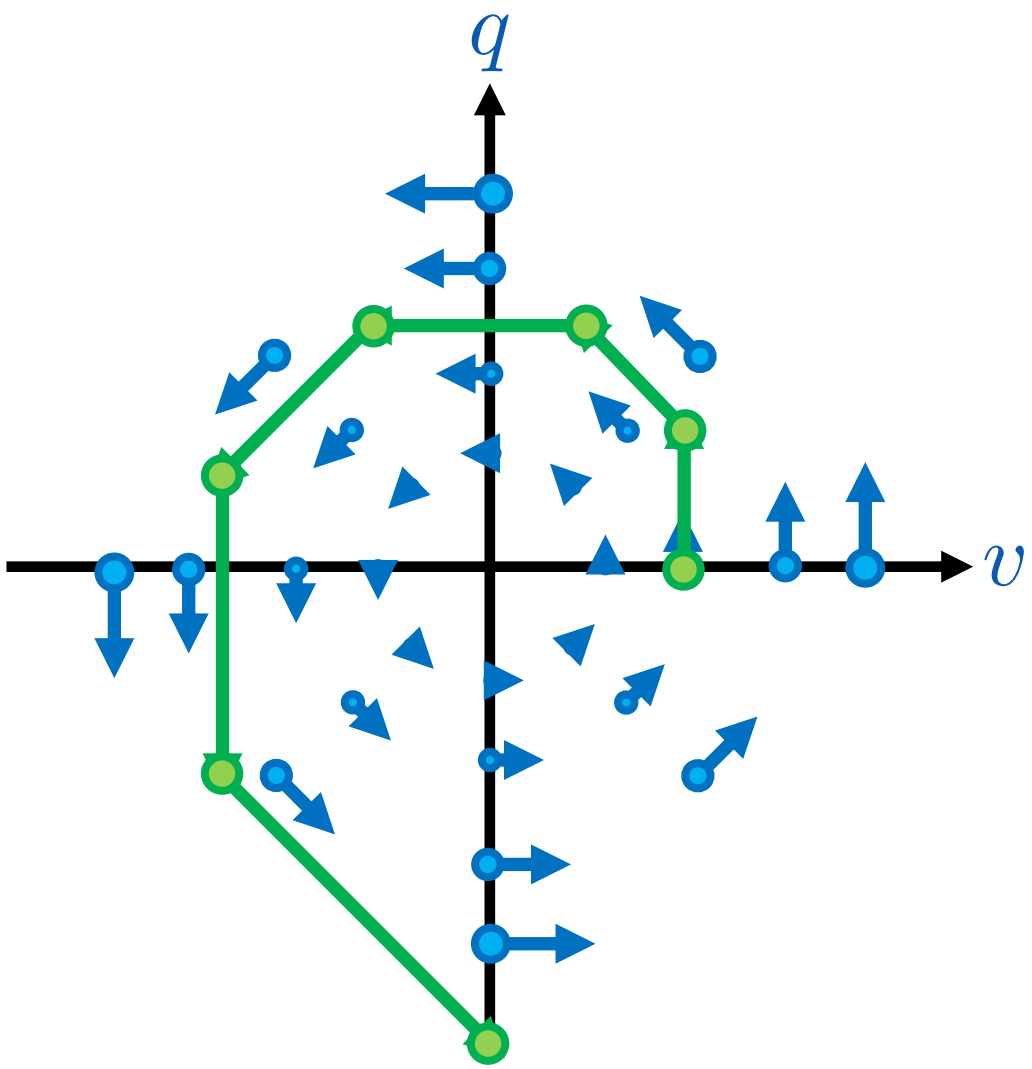
$$\left(\mathrm{I} - \Delta t \mathrm{A}^{-1}\mathrm{B}\right)\mathbf{y}^{t+1} = \mathbf{y}^{t}$$

$$v^{t+1} + \Delta t \frac{k}{m} q^{t+1} = v^{t} \qquad q^{t+1} - \Delta t v^{t+1} = q^{t}$$

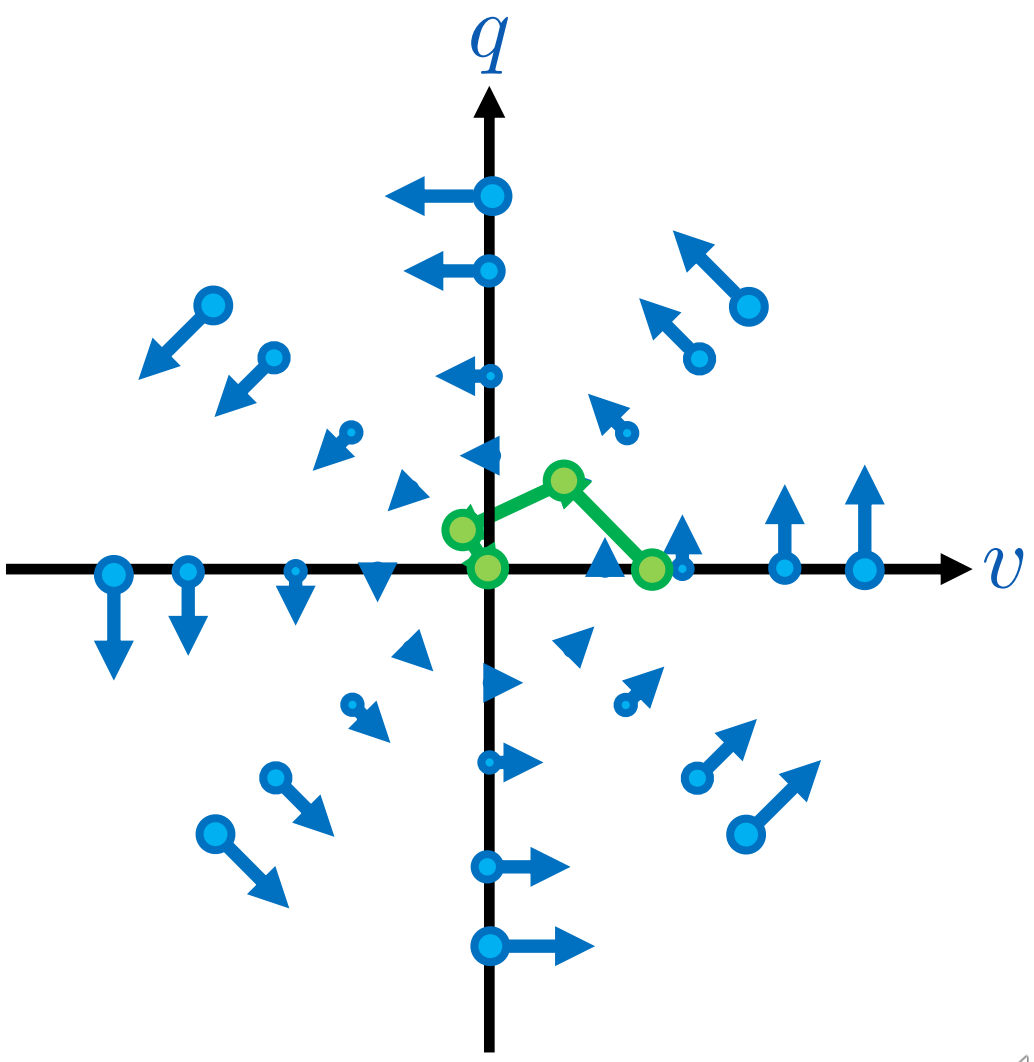
$$v^{t+1} + \Delta t \frac{k}{m}\left(q^{t} + \Delta t v^{t+1}\right) = v^{t}$$

$$\left(1 + \Delta t^{2}\frac{k}{m}\right) v^{t+1} = v^{t} - \Delta t \frac{k}{m} q^{t} \qquad q^{t+1} = q^{t} + \Delta t v^{t+1}$$

Phase Space Trajectory



Explicit (exploding!!!)



Implicit (damping)

Symplectic Euler Time Integration

First take an explicit step

Next take an implicit step

If we do this just right, we can ”cancel out” long term exploding and damping

Symplectic Euler Time Integration

First take an explicit velocity step

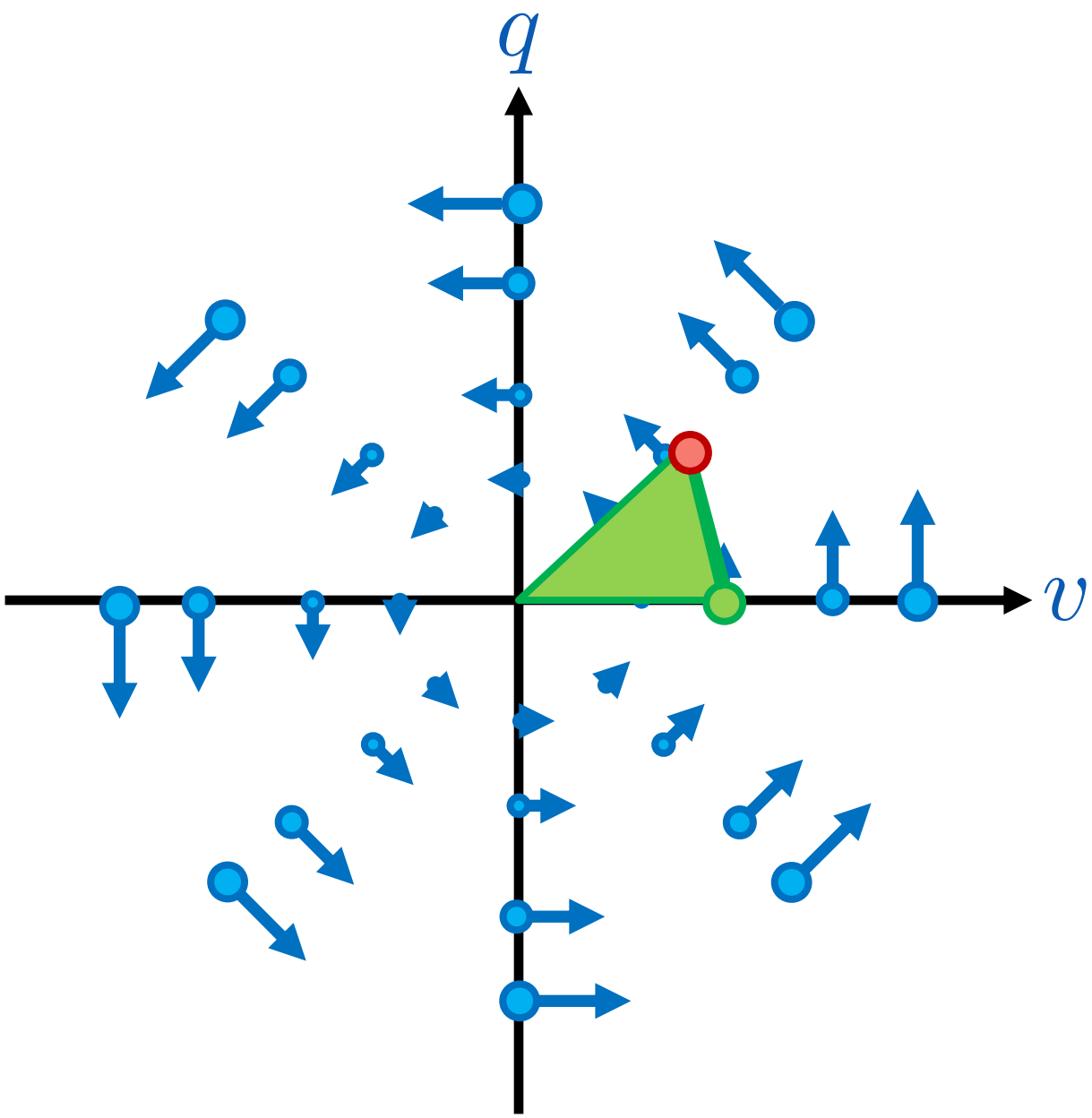
$$v^{t+1} = v^t - \Delta t \frac{k}{m} q^t$$

Next take an implicit position step

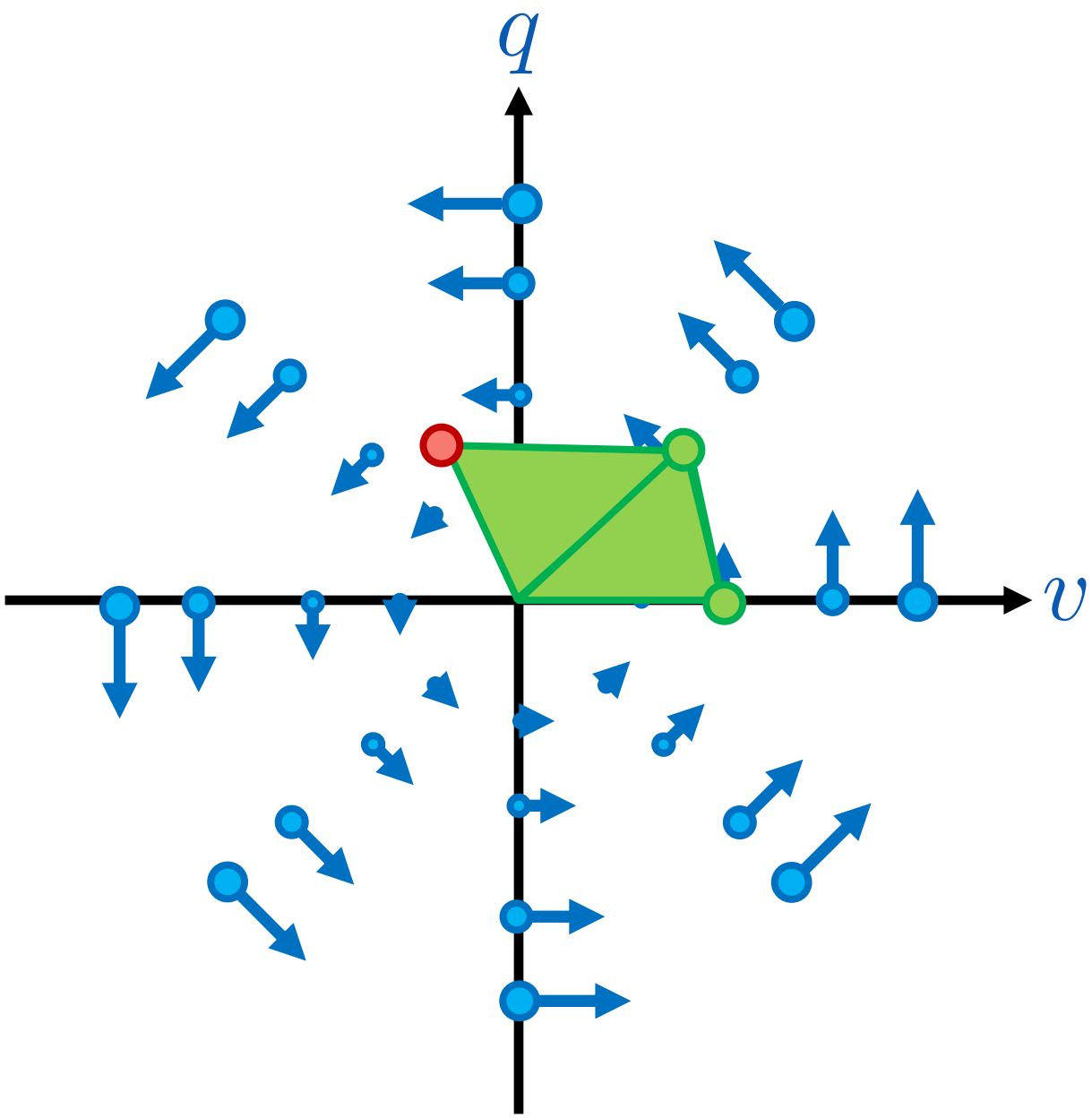
$$q^{t+1} = q^t + \Delta t v^{t+1}$$

If we do this just right, we can ”cancel out” exploding and damping

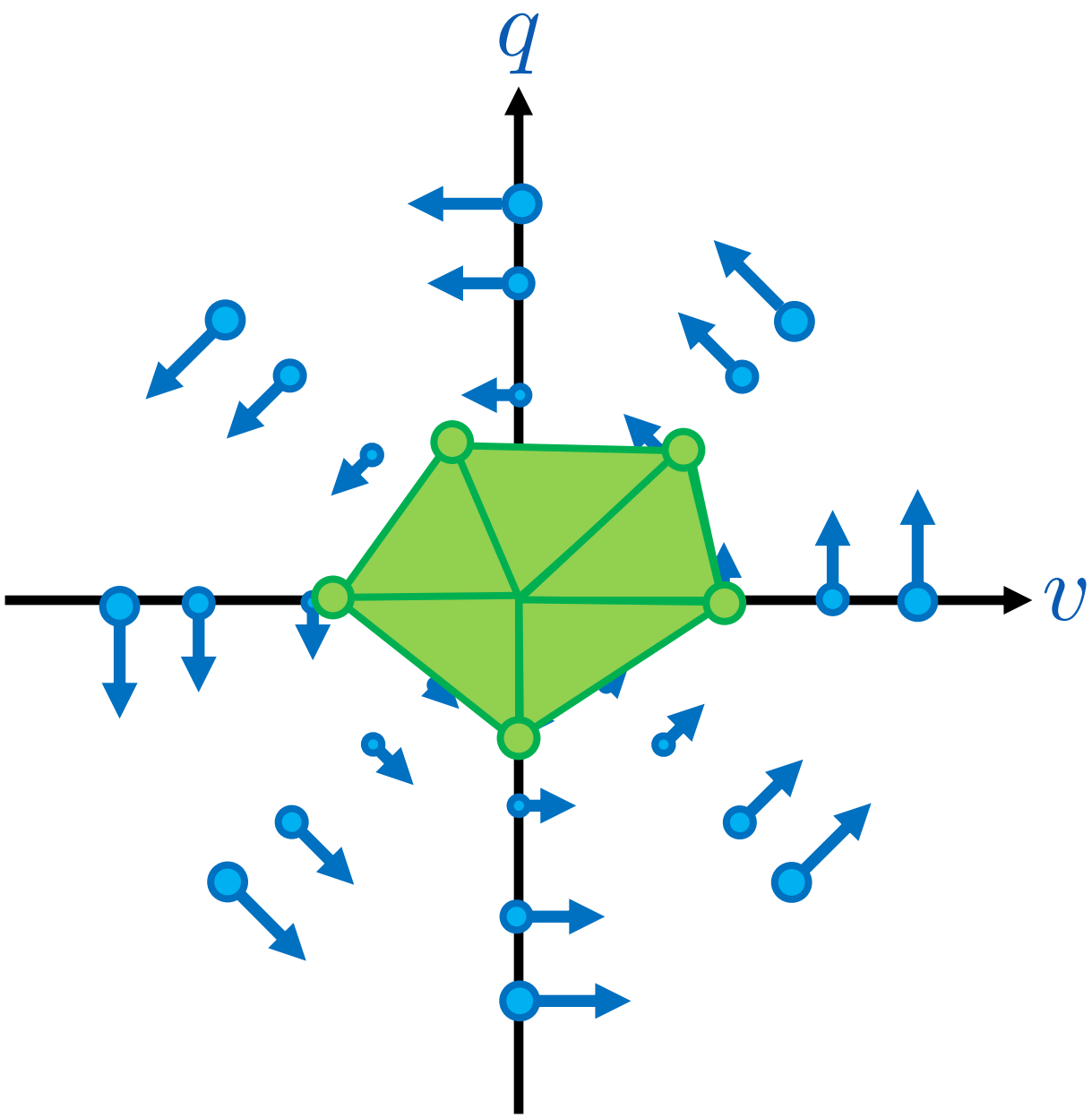
Phase Space Trajectory



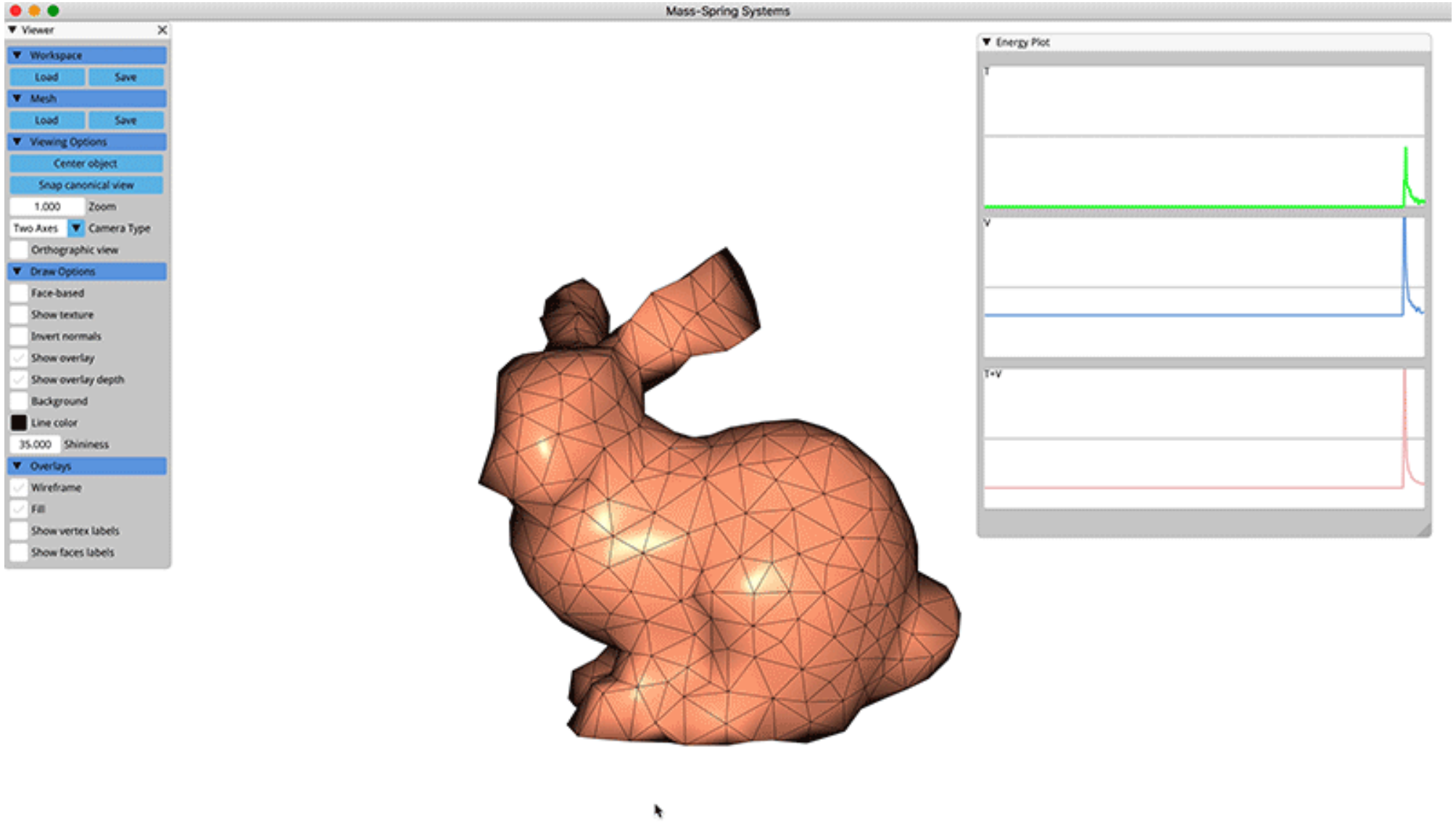
Phase Space Trajectory



Phase Space Trajectory



Next Week:


Mass-Spring Systems
Viewer
Workspace
Load
Save
Mesh
Load
Save
Viewing Options
Center object
Snap canonical view
1.000
Zoom
Two Axes
Camera Type
Orthographic view
Draw Options
Face-based
Show texture
Invert normals
Show overlay
Show overlay depth
Background
Line color
35.000
Shininess
Overlays
Wireframe
Fill
Show vertex labels
Show faces labels
Energy Plot
T
V
T+V