What “Damping” Means in a Spring-Mass Model

When you attach an ideal spring between two masses, the system oscillates forever because a Hookean spring stores but never dissipates mechanical energy.

Damping is any mechanism you add to remove that energy so the motion eventually settles instead of ringing indefinitely or even blowing up numerically.

1 Physical Picture

Think of a **dash-pot**: a piston moving through a viscous fluid.

The resisting force is proportional to the relative velocity of the two ends:

$$\boxed{f_{\mathrm{damp}} = -\,c\,\dot{x}} \tag{1}$$

c is the damping coefficient [N·s/m] (or N·m/rad for rotations).

$\dot{x}$ is the velocity (or angular-velocity) difference being damped.

The dash-pot converts kinetic energy into heat, so total mechanical energy monotonically decreases.

---

2 Mathematical Role in the Adams Formulation

Table 1: Examples of Damping in the Model

Where used in the code	**Term**	**Purpose**
`self.vel[i] *= self.damping`	Global velocity damping ($0 < \mathrm{damping} < 1$)	Cheap way to kill high-frequency noise each step.
`torsion_damp_*` in `_apply_torsion_springs`	Torsional dash-pot $f = -c_\tau(\hat{\zeta}_\ell \cdot \dot{\hat{\zeta}}_m + \dot{\hat{\zeta}}_\ell \cdot \hat{\zeta}_m)\,\hat{\zeta}_m$	Prevents the two axes from “whipping” past their rest angle.
`f_jdamp̂=-c(v_j-v_b)` in barycentric springs	Bulk-volume dash-pot	Stops spurious oscillation of the barycenter.

All of them follow the same linear law: $f \propto$ (relative velocity).

---

3 Classical Single-DOF Analogy

For one mass–spring–damper system, the equation of motion is:

$$m\ddot{x} + c\dot{x} + kx = 0 \tag{2}$$

The **damping ratio** is $\zeta = \frac{c}{2\sqrt{km}}$.

- $\zeta < 1$: **underdamped** (oscillatory decay)
- $\zeta = 1$: **critical damping** (fastest non-oscillatory return)
- $\zeta > 1$: **overdamped** (slow, aperiodic return)

In large systems you rarely compute a global ζ, but the intuition is the same: picking c larger than zero suppresses oscillations; too large and the system becomes sluggish or even unstable under explicit time integration (large damping forces create stiff terms).

---

4 How to Choose c or the Per-Step Factor

- **Empirical tuning** – start small (e.g., 0.01–0.1× critical) and increase until jitter disappears without visibly slowing motion.
- **Rayleigh (mass + stiffness) damping** – $C = \alpha M + \beta K$ gives frequency-independent control; often used in FEM but heavier to implement.
- **Time-step-scaled factor** – multiplying velocities by a constant < 1 (e.g., $v \leftarrow (1-\gamma\Delta t)v$) is equivalent to viscous damping with $c = \gamma m$.

In your code:

```
self.damping = 0.90  # multiply velocity each step
c_tau         = 0.05 # torsion dash-pot coefficient
```

These values are low enough to stabilize but high enough to damp out fast modes in 1–2 simulated seconds.

---

5 Take-away

Damping is the viscous, energy-dissipating counterpart to the elastic spring.
It is crucial in real-time or explicit simulations to:

- stabilise the integrator,
- prevent unrealistic ringing after sudden load changes,
- let the model settle into a steady shape.

Always document which damping terms you include, their coefficients, and whether they are global, per-spring, or frequency-selective so readers can reproduce or retune the behaviour.