

Mixed Implicit/Explicit

- For some problems, that square root can mean velocity limit much stricter
- Or, we know we want to properly resolve the position-based oscillations, but don't care about exact damping rate
- Go explicit on position, implicit on velocity
 - Often, a(x,v) is linear in v, though nonlinear in x; this way we avoid Newton iteration

Newmark Methods

A general class of methods

 $x_{n+1} = x_n + \Delta t v_n + \frac{1}{2} \Delta t^2 [(1 - 2\beta)a_n + 2\beta a_{n+1}]$ $v_{n+1} = v_n + \Delta t [(1 - \gamma)a_n + \gamma a_{n+1}]$

- Includes Trapezoidal Rule for example (β=1/4, γ=1/2)
- υ The other major member of the family is Central Differencing (β =0, γ =1/2)
 - This is mixed Implicit/Explicit

Central Differencing

• Rewrite it with intermediate velocity:

$$v_{n+\frac{1}{2}} = v_n + \frac{1}{2}\Delta t a(x_n, v_n)$$

$$x_{n+1} = x_n + \Delta t v_{n+\frac{1}{2}}$$

$$v_{n+1} = v_{n+\frac{1}{2}} + \frac{1}{2}\Delta t a(x_{n+1}, v_{n+1})$$

- Looks like a hybrid of:
 - Midpoint (for position), and
 - Trapezoidal Rule (for velocity split into Forward and Backward Euler half steps)

cs533d-term1-2005

Central: Performance

- Constant acceleration: great
 - 2nd order accurate
- Position dependence: good
 - Conditionally stable, no damping
- Velocity dependence: good
 - Stable, but only conditionally monotone
- Can we change the Trapezoidal Rule to Backward Euler and get unconditional monotonicity?

cs533d-term1-2005

Staggered Implicit/Explicit

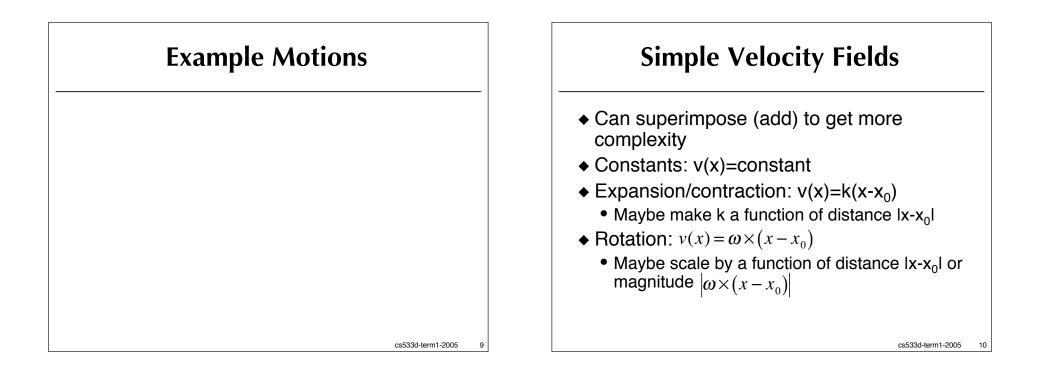
 Like the staggered Symplectic Euler, but use B.E. in velocity instead of F.E.:

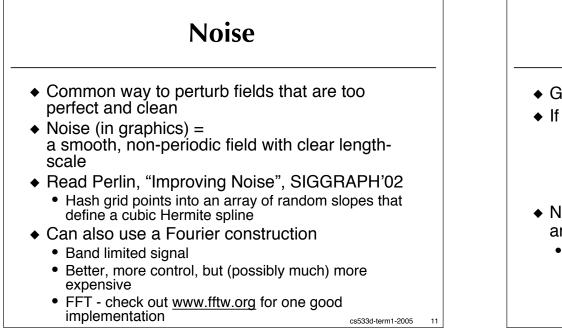
$$v_{n+\frac{1}{2}} = v_{n-\frac{1}{2}} + \frac{1}{2}(t_{n+1} - t_{n-1})a(x_n, v_{n+\frac{1}{2}})$$
$$x_{n+1} = x_n + \Delta t v_{n+\frac{1}{2}}$$

- Constant acceleration: great
- Position dependence: good (conditionally stable, no damping)
- Velocity dependence: great (unconditionally monotone)

Summary (2nd order)

- Depends a lot on the problem
 - What's important: gravity, position, velocity?
- Explicit methods from last class are probably bad
- Symplectic Euler is a great fully explicit method (particularly with staggering)
 - Switch to implicit velocity step for more stability, if damping time step limit is the bottleneck
- Implicit Compromise method
 - Fully stable, nice behaviour





Example Forces

- ◆ Gravity: F_{gravity}=mg (a=g)
- If you want to do orbits

$$F_{gravity} = -GmM_0 \frac{x - x_0}{\left|x - x_0\right|^3}$$

- Note x₀ could be a fixed point (e.g. the Sun) or another particle
 - But make sure to add the opposite and equal force to the other particle if so!

Drag Forces

- ◆ Air drag: F_{drag}=-Dv
 - If there's a wind blowing with velocity v_w then F_{drag} =-D(v-v_w)
- D should be a function of the cross-section exposed to wind
 - Think paper, leaves, different sized objects,
- Depends in a difficult way on shape too
 - Hack away!

cs533d-term1-2005

Spring Forces

- Springs: $F_{spring} = -K(x-x_0)$
 - x₀ is the attachment point of the spring
 - Could be a fixed point in the scene
 - ...or somewhere on a character's body
 - ...or the mouse cursor
 - ...or another particle (but please add equal and oppposite force!)

cs533d-term1-2005

Nonzero Rest Length Spring

 Need to measure the "strain": the fraction the spring has stretched from its rest length L

$$F_{spring} = -K \left(\frac{|x - x_0|}{L} - 1 \right) \frac{|x - x_0|}{|x - x_0|}$$

Spring Damping

- ♦ Simple damping: F_{damp}=-D(v-v₀)
 - But this damps rotation too!
- ♦ Better spring damping: F_{damp}=-D(v-v₀)•u/L u
 - Here u is $(x-x_0)/lx-x_0l$, the spring direction
- ◆ [work out 1d case]
- Critical damping: fastest damping possible
 - For individual springs, gives a good typical damping force you can multiply by a factor

<section-header><section-header><section-header><section-header><section-header><section-header><section-header><section-header><section-header><section-header><section-header><section-header><section-header><section-header><section-header><section-header>

Collision and Contact

- We can integrate particles forward in time, have some ideas for velocity or force fields
- But what do we do when a particle hits an object?
- No simple answer, depends on problem as always
- General breakdown:
 - Interference vs. collision detection
 - What sort of collision response: (in)elastic, friction
 - Robustness: do we allow particles to actually be inside an object?

cs533d-term1-2005 18

Interference vs. Collision

- Interference (=penetration)
 - Simply detect if particle has ended up inside object, push it out if so
 - Works fine if $v\Delta t < \frac{1}{2}w$ [w=object width]
 - Otherwise could miss interaction, or push dramatically the wrong way
 - The ground, thick objects and slow particles
- Collision
 - Check if particle trajectory intersects object
 - Can be more complicated, especially if object is moving too...
- For now, let's stick with the ground (y=0)

Repulsion Forces

- Simplest idea (conceptually)
 - Add a force repelling particles from objects when they get close (or when they penetrate)
 - Then just integrate: business as usual
 - Related to penalty method: instead of directly enforcing constraint (particles stay outside of objects), add forces to encourage constraint
- For the ground:
 - F_{repulsion}=-Ky when y<0 [think about gravity!]
 - ...or -K(y-y0)-Dv when y<y0 [still not robust]
 - ...or K(1/y-1/y0)-Dv when y<y0

19

Repulsion forces

- Difficult to tune:
 - Too large extent: visible artifact
 - Too small extent: particles jump straight through, not robust (or time step restriction)
 - Too strong: stiff time step restriction, or have to go with implicit method - but Newton will not converge if we guess past a singular repulsion force
 - Too weak: won't stop particles
- Rule-of-thumb: don't use them unless they really are part of physics
 - Magnetic field, aerodynamic effects, ...

cs533d-term1-2005

21

Collision and Contact

- Collision is when a particle hits an object
 - Instantaneous change of velocity (discontinuous)
- Contact is when particle stays on object surface for positive time
 - Velocity is continuous
 - Force is only discontinuous at start

cs533d-term1-2005 22

• At point of contact, find normal n • For ground, n=(0,1,0)• Decompose velocity into • normal component $v_N=(v\cdot n)n$ and • tangential component $v_T=v\cdot v_N$ • Normal response: $v_N^{after} = -\varepsilon v_N^{before}$, $\varepsilon \in [0,1]$ • $\varepsilon=0$ is fully inelastic • $\varepsilon=1$ is elastic • Tangential response • Frictionless: $v_T^{after} = v_T^{before}$ • Then reassemble velocity $v=v_N+v_T$

cs533d-term1-2005 23