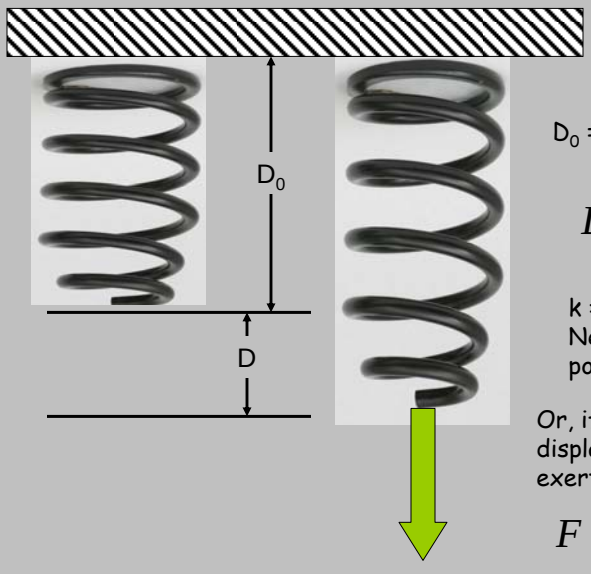
Modeling the World as a Mesh of Springs

Mike Bailey

Oregon State University

How do Springs Work?



D_0 = unloaded spring length

$$D = D_0 + \frac{F}{k}$$

k = spring stiffness in Newtons/meter or pounds/inch

Or, if you know the displacement, the force exerted by the spring must be:

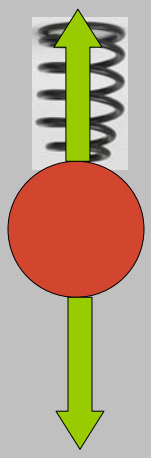
$$F = k(D - D_0)$$

Solving Motion where there is a Spring

$$F_{spring} = k(D - D_0)$$

$$v(y,t) = \int \frac{\sum F}{m} dt$$

$$v(y,t) = \int \frac{W - ky}{m} dt$$



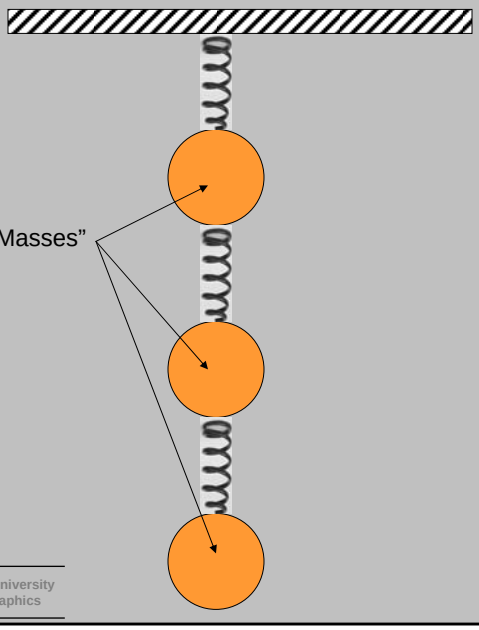
Weight

```
void
GetVelAcc( float t, float y, float vy, float *v, float *a )
{
        *v = vy;
        *a = ( W – K*y  ) /  MASS;
}
```

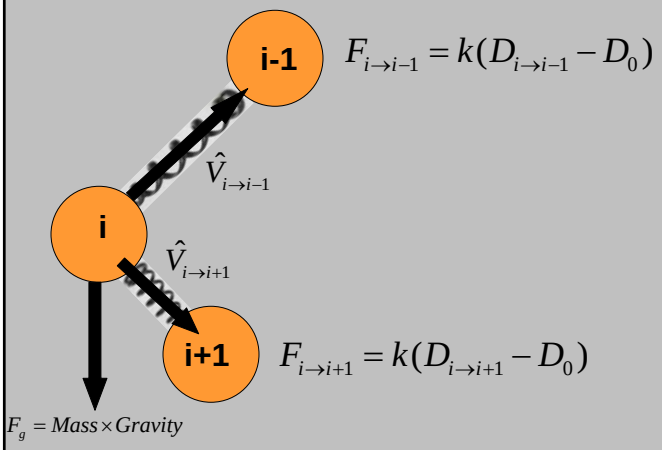
Note that, instead of using weight, we could write that last line with mass:

```
*a = ( MASS*GRAVITY – K*y  ) /  MASS;
```

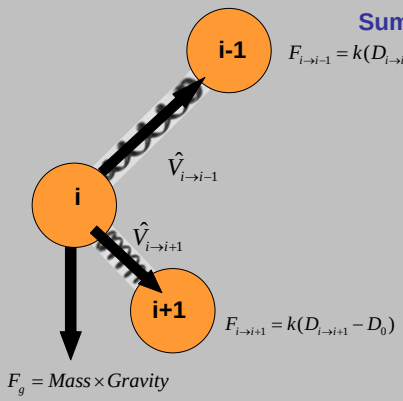

Modeling a String as a Group of Masses Connected by Springs



Computing Forces



Summing Forces



$$\sum F_{i,x} = \hat{V}_{i\to i-1,x} \cdot F_{i\to i-1} + \hat{V}_{i\to i+1,x} \cdot F_{i\to i+1} = Mass \cdot \ddot{x}$$

$$\sum F_{i,y} = \hat{V}_{i\to i-1,y} \cdot F_{i\to i-1} + \hat{V}_{i\to i+1,y} \cdot F_{i\to i+1} + F_g = Mass \cdot \ddot{y}$$

Solve for Each State as a Whole, not as Individual Links: Do it this Way

Second Order solution:

```
for( int i = 0; i < NUMLINKS; i++ )
{
        GetVelAccel( i,  &vx1[i], &vy1[i], &ax1[i], &ay1[i] );
}
for( int i = 0; i < NUMLINKS; i++ )
{
        vxtmp[i] = Links[i].vx + DT * ax1[i];
        vytmp[i] = Links[i].vy + DT * ay1[i];
        xtmp[i]  = Links[i].x  + DT * vx1[i];
        ytmp[i]  = Links[i].y  + DT * vy1[i];
}

for( int i = 0; i < NUMLINKS; i++ )
{
        GetVelAccel( i,  &vx2[i], &vy2[i], &ax2[i], &ay2[i] );
}
for( int i = 0; i < NUMLINKS; i++ )
{
        Links[i].vx = Links[i].vx  + DT * ( ax1[i] + ax2[i] ) / 2.;
        Links[i].vy = Links[i].vy  + DT * ( ay1[i] + ay2[i] ) / 2.;
        Links[i].x  = Links[i].x   + DT * ( vx1[i] + vx2[i] ) / 2.;
        Links[i].y  = Links[i].y   + DT * ( vy1[i] + vy2[i] ) / 2.;
}
```

Get *all* the velocities and accelerations

Apply *all* the velocities and accelerations

Get *all* the velocities and accelerations

Apply *all* the velocities and accelerations

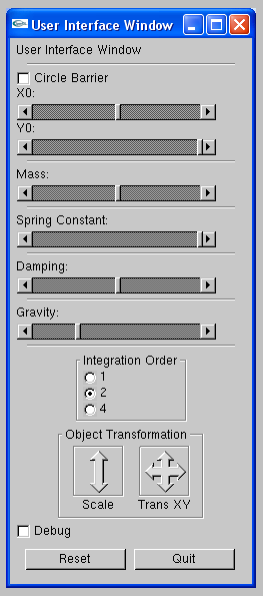
Don't do it this Way!

Second Order solution:

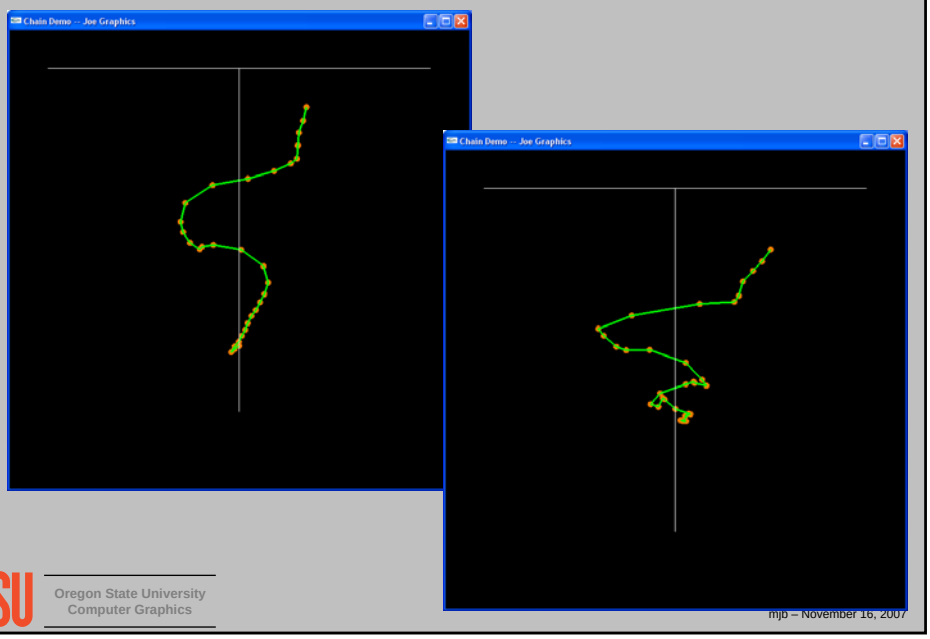
```
for( int i = 0; i < NUMLINKS; i++ )
{
        GetVelAccel( i,  &vx1[i], &vy1[i], &ax1[i], &ay1[i] );
        vxtmp[i] = Links[i].vx + DT * ax1[i];
        vytmp[i] = Links[i].vy + DT * ay1[i];
        xtmp[i]  = Links[i].x  + DT * vx1[i];
        ytmp[i]  = Links[i].y  + DT * vy1[i];
}

for( int i = 0; i < NUMLINKS; i++ )
{
        GetVelAccel( i,  &vx2[i], &vy2[i], &ax2[i], &ay2[i] );
        Links[i].vx = Links[i].vx  + DT * ( ax1[i] + ax2[i] ) / 2.;
        Links[i].vy = Links[i].vy  + DT * ( ay1[i] + ay2[i] ) / 2.;
        Links[i].x  = Links[i].x   + DT * ( vx1[i] + vx2[i] ) / 2.;
        Links[i].y  = Links[i].y   + DT * ( vy1[i] + vy2[i] ) / 2.;
}
```

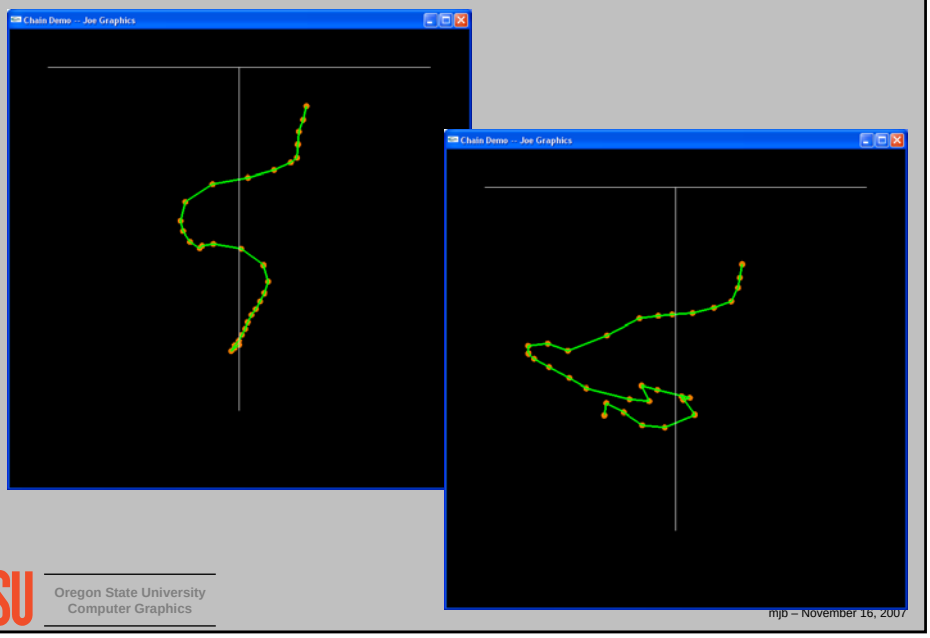
Changing Variables on-the-fly



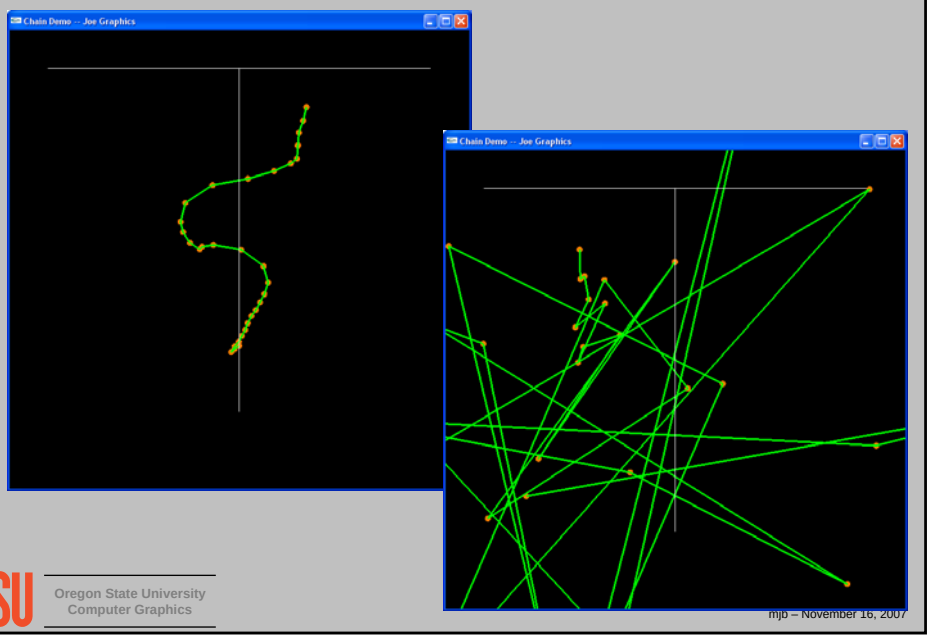
Simulating a String



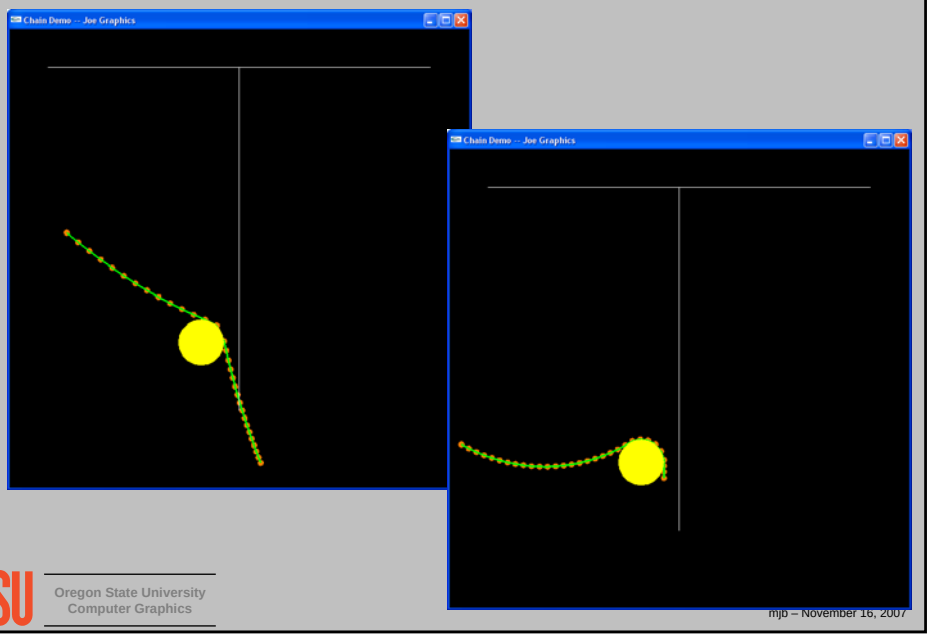
Less Damping



First Order Instability



Placing a Physical Barrier in the Scene



Placing a Physical Barrier in the Scene

```
if( DoCircle )
{
	for( int i = 0; i < NUMLINKS; i++ )
	{
		float dx = Links[i].x - CIRCX;
		float dy = Links[i].y - CIRCY;
		float rsqd = dx*dx + dy*dy;
		if( rsqd < CIRCR*CIRCR )
		{
			float r = sqrt( rsqd );
			dx /= r;
			dy /= r;
			Links[i].x = CIRCX + CIRCR * dx;
			Links[i].y = CIRCY + CIRCR * dy;
			Links[i].vx *=  dy;
			Links[i].vy *= -dx;
		}
	}
}
```

Vector from circle center to the lumped mass

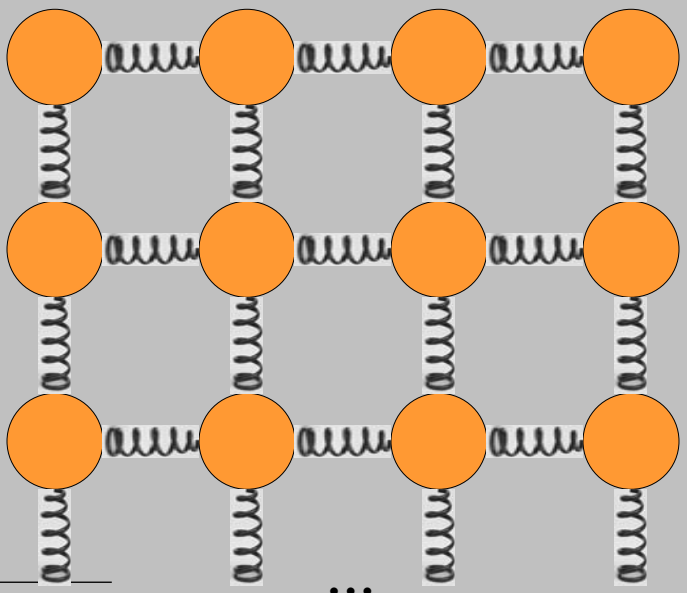
If the lumped mass is inside the circle …

Unit vector from circle center to the lumped mass

Push the lumped mass from inside the circle to the circle's surface

Keep just the tangential velocity

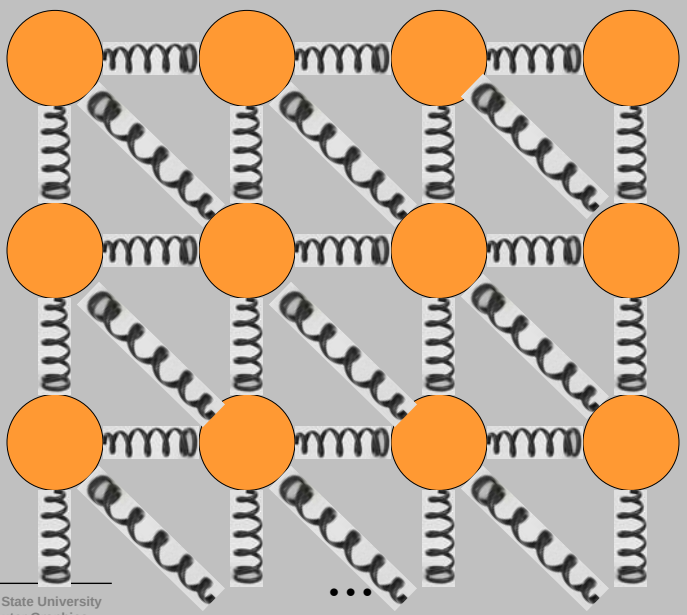
Modeling Cloth



Oregon State University
Computer Graphics

mjb – November 16, 2007

Modeling Cloth



Oregon State University
Computer Graphics

mjb – November 16, 2007

Examples





David E. Breen Donald H. House, Michael J. Wozny: *Predicting the Drape of Woven Cloth Using Interacting Particles Using Interacting Particles*

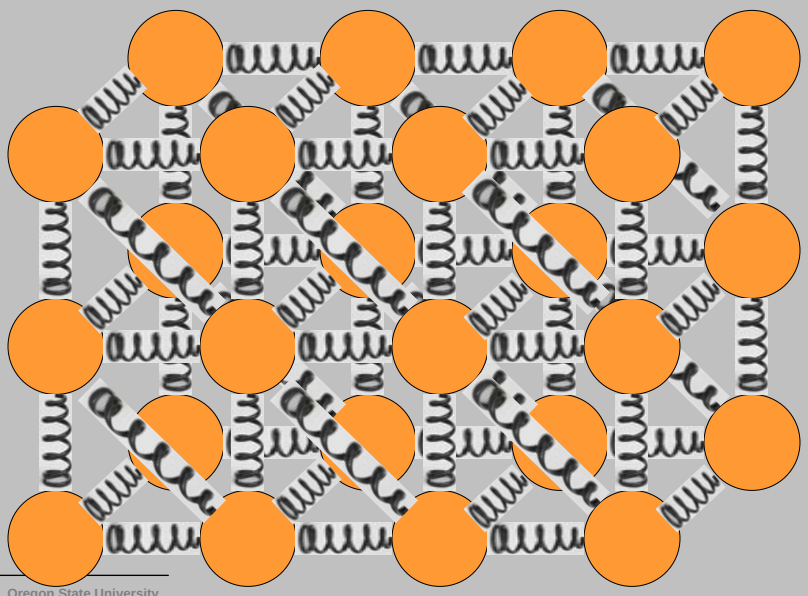
Examples

MiraLab, University of Geneva

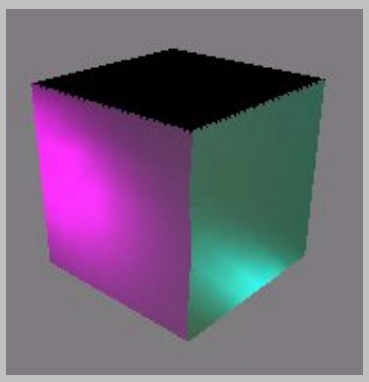
Examples

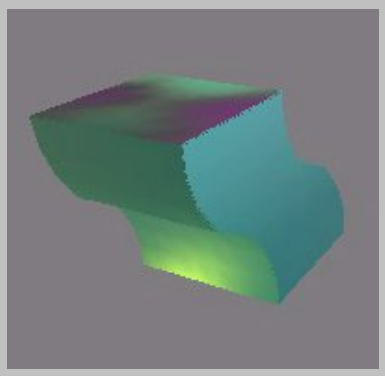
Pixar

Modeling Jello



Examples





MIT

We can also use this same Method to Model Rigid Objects

California Department of Transportation

We can also use this same Method to Model Rigid Objects