Computer Graphics & Scientific Visualization

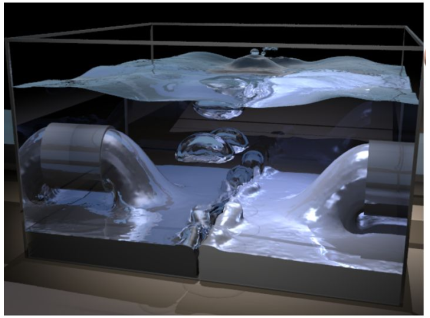
MPRI 2-39

Physically based animation

Why physically-based animation

1- **Key Framing**

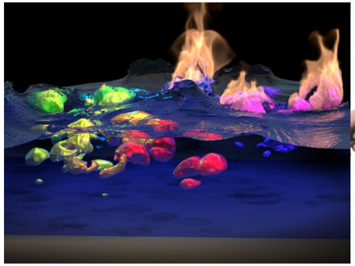
- + *Full control*
- - *Dynamic hard to control, tedious details*



2- **Physically-based animation**

- + *Dynamic handling*

Typically (fluids, gaz, cloth, hairs, …)



Note: Predictive simulation != CG simulation

- Model, solver accuracy != Speed, robustness, plausibility

General approach of physically-based simulation

1- Setup Initial Condition (position, speed, forces) $S(t=0) = S_0$

2- Describes constitutive relations
-> Set of differential equations

$$\mathcal{F}\left(t, S, \frac{\partial^n S}{\partial t^n}, \frac{\partial^m S}{\partial x^m}\right) = 0$$

3- Numerically solves equations incrementing through time

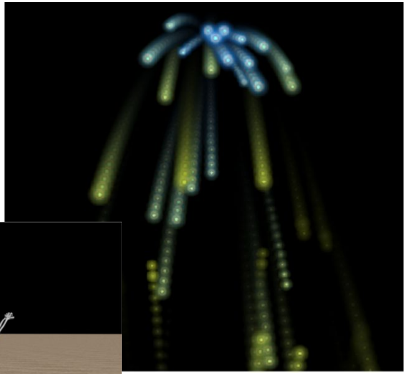
$$S(t + \Delta t) = S(t) + \Delta t \ldots$$

Seminal article: *[D. Terzopoulos, Deformable Models, 88]*

Physically-based simulation models

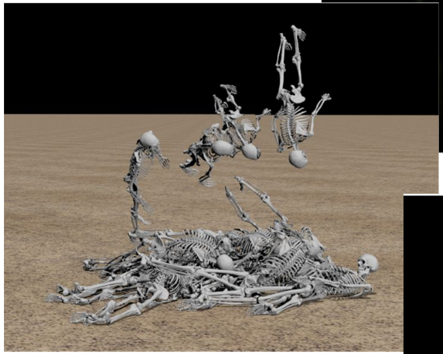
Particle based

- \+ Simple
- \- Accuracy



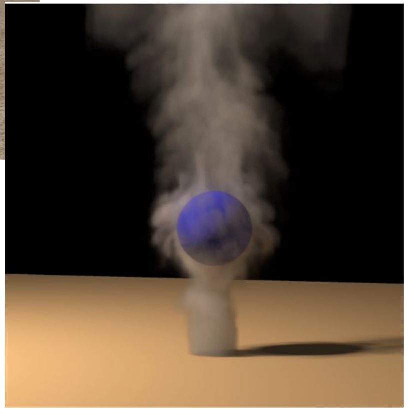
Rigid bodies

- \+ Inertia, orientation
- \- Collision



Continuum mechanics

- \+ Accurate
- \- Computationally intensive, complex



Particle based

Particles-based methods

Particle fully defined by its position p

Fundamental equation of dynamics

$$\mathrm{m\,p''(t) = F(t)}$$

$$\begin{cases} m\,v'(t) = F(t) \\ p'(t) = v(t) \end{cases}$$

ODE in t

ex. F =g

[LucasFilm, Star Trek II]

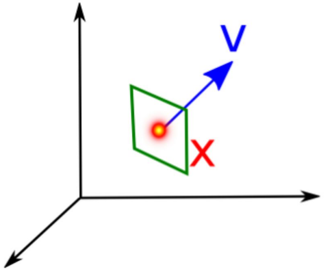
[Reeves, TOG83]

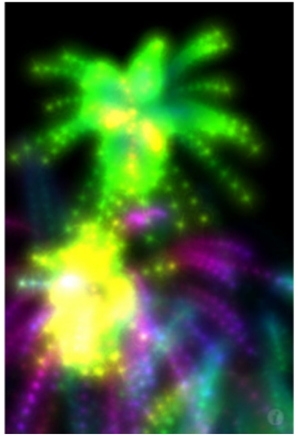
$$p(t) = 1/2g\,t^2 + v_0 t + p_0$$

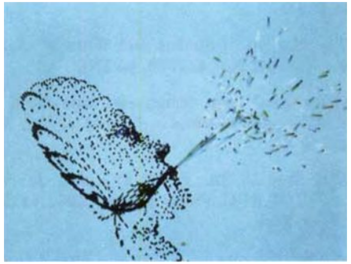
Very special case of analytical solution

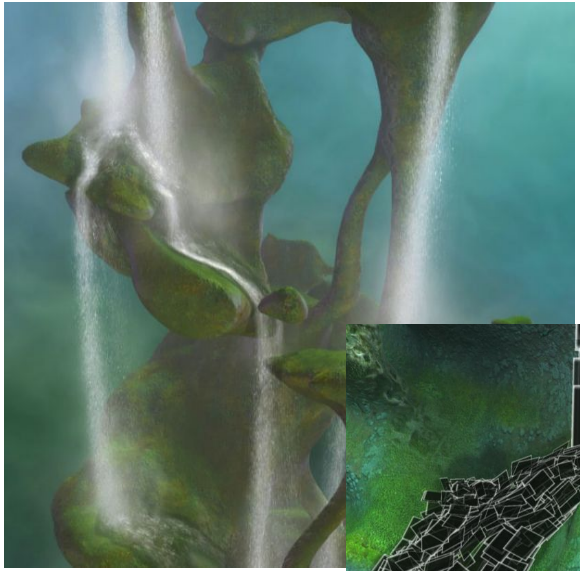
Particles: Sprites

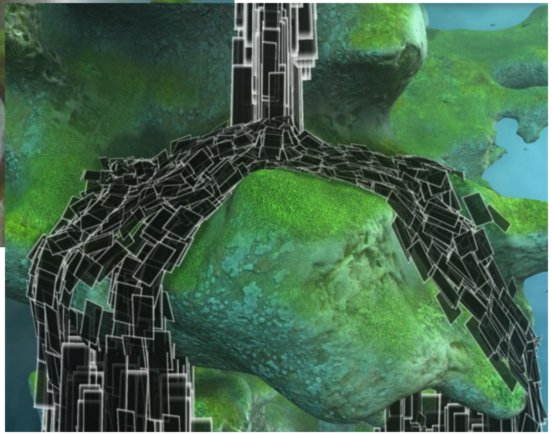
Emit, suppress particles
Transparent texture
Fall under gravity, follow field











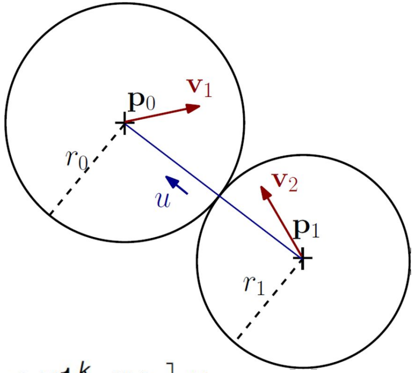
NVIDIA

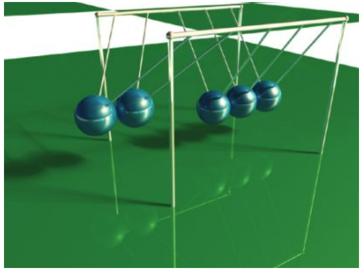
[Karl Sims, Particle animation and rendering using data parallel computation, SIGGRAPH90]

Particles: Hard sphere

Sphere of mass m, center x, radius r

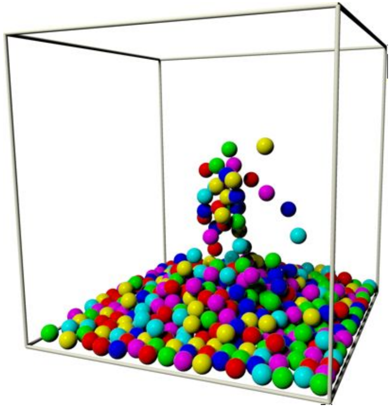
Collision if $\|x_1 - x_2\| < r_1 + r_2$





Speed after collision

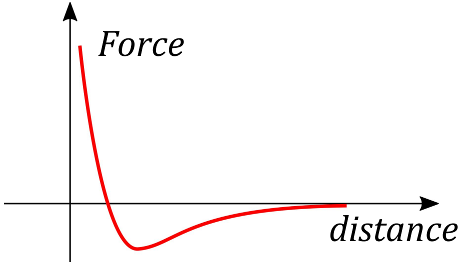
$$\begin{cases} v_1^{k+1} = v_1^k + \frac{1}{m_1+m_2}\left[m2 < v2^k, u > -\frac{1}{2}(m_1+3m_2) < v1^k, u >\right] u \\ v_2^{k+1} = v_2^k + \frac{1}{m_1+m_2}\left[m1 < v1^k, u > -\frac{1}{2}(m_2+3m_1) < v2^k, u >\right] u \\ u = x1 - x0 \end{cases}$$



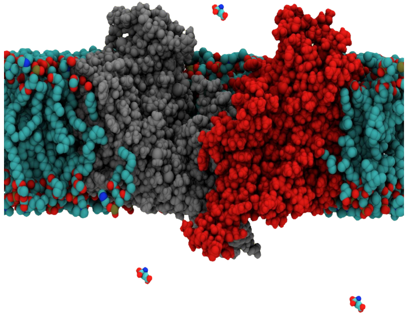
+ *Project back the sphere to the collision surface*

Particles: Interaction and force fields

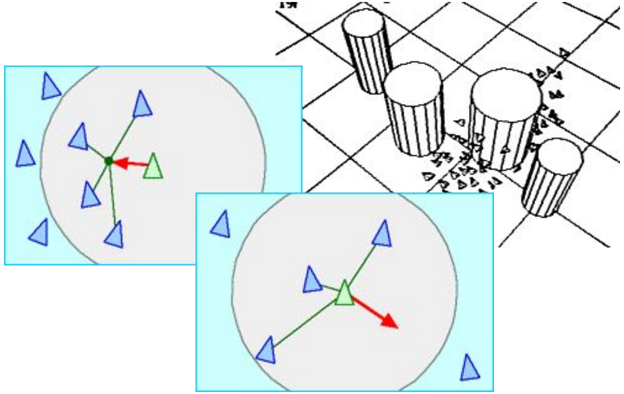
Molecular dynamic
inspired



ex. Lennard-Jones



Boids model
[Reynolds, SIGGRAPH 87]

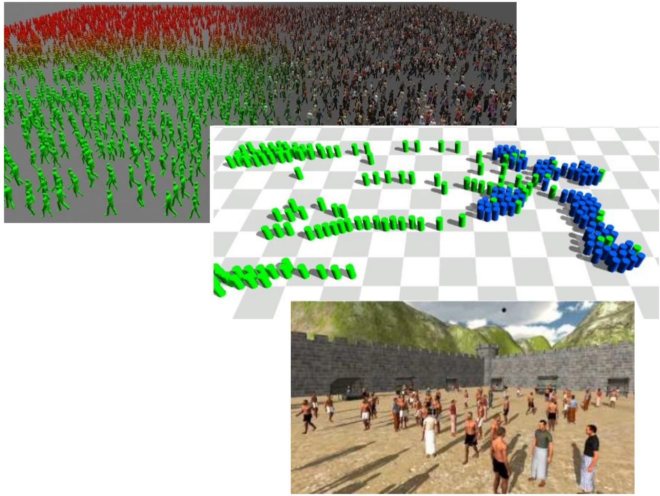


Steering forces:
- Attraction/repulsion
- Following, alignment, …

-> emerging behaviors

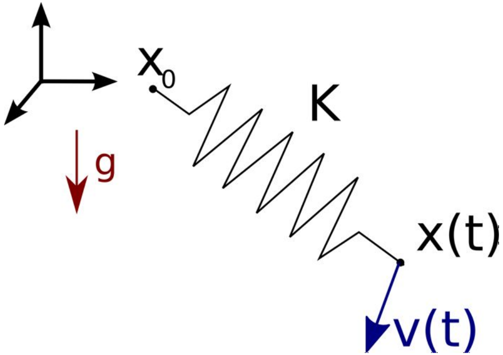
Crowd models
[Pettre et al, 15]

OpenSteer

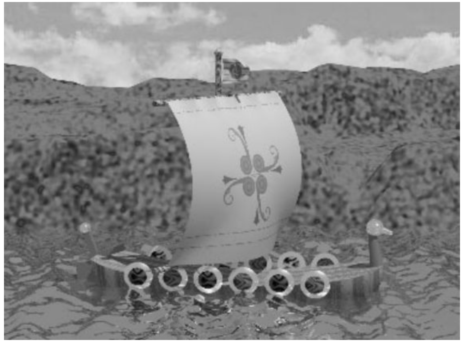


[Galvane et al, 13]

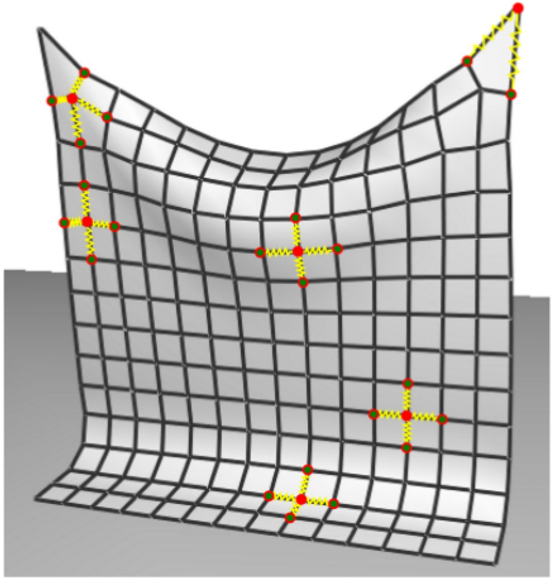
Particles: Spring mass



$$F(t) = K\ \left(L_0 - \|\mathbf{x}(t) - \mathbf{x}_0\|\right) \frac{\mathbf{x}(t) - \mathbf{x}_0}{\|\mathbf{x}(t) - \mathbf{x}_0\|}$$



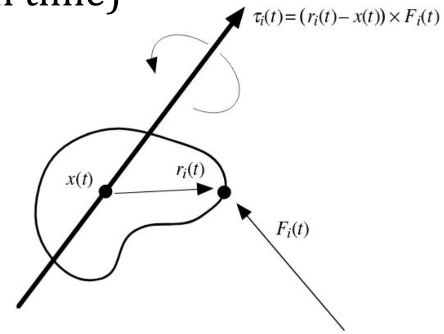
[Provot, 95]



Rigid Bodies

Rigid bodies

Rigid body fully defined by the position of its **center of gravity p**, and its **orientation R**
Dynamic takes into account: **inertia** (change of orientation through time)



[Baraff, Witkin, SIGGRAPH Course 01]

Fundamental equation of dynamics

$$\begin{cases} Mv'(t) = F(t) \\ I\omega'(t) + \omega(t) \times (I\,\omega(t)) = \tau(t) \end{cases}$$

$$v(t) = x'(t) \qquad R'(t) = \omega(t)R(t)$$

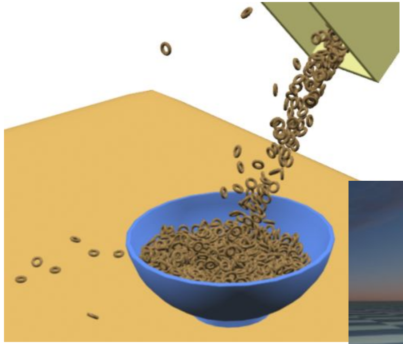
$$F(t) = \int_q F(q,t)\,\mathrm{d}q \qquad \tau(t) = \int_q (q(t) - p(t)) \times F(q,t)\,\mathrm{d}q$$

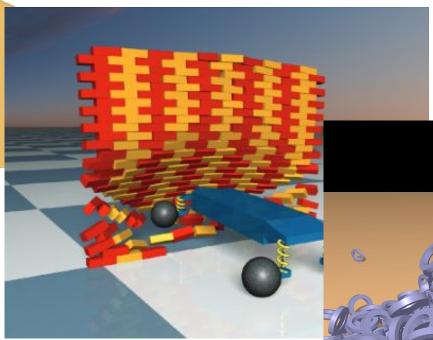
$$I = \int_q (q-p)^2\,\mathrm{Id} - (q-p)(q-p)^t\,\mathrm{d}q$$

Rigid bodies examples

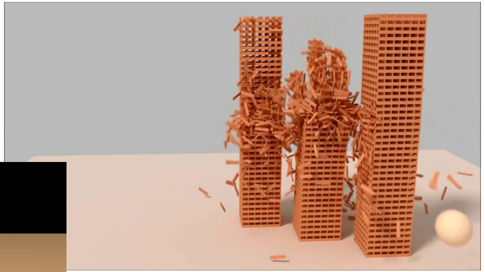
Widely used in VFX and video games:
Stacks of elements, Fracture, Explosions, ...

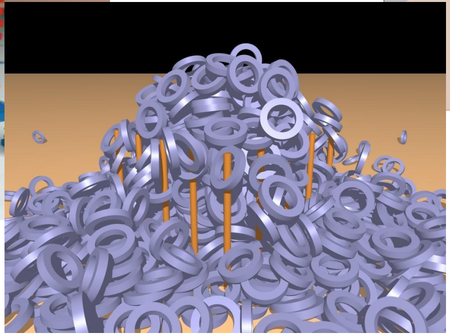
Main Open Softwares : **ODE, Bullets, PhysX**





[Bridson 03]





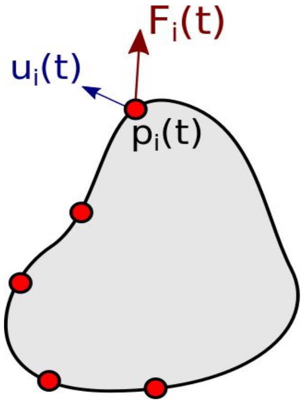
[Bender et al.
Interactive Simulation of Rigid Bodies in Computer Graphics, CGF 13]

Continuous mechanics

Lagrangian / Eulerian

Characterizing a deforming object: 2 point of views

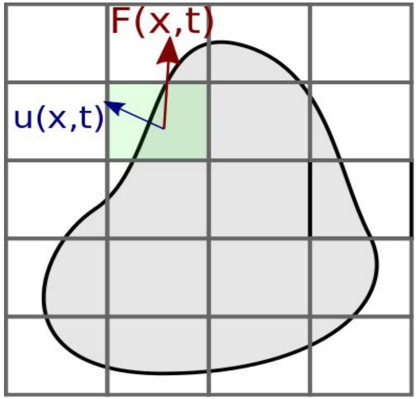
Lagrangian:
forces, speed attached to moving particle



\+ *advection*

$$a_i(t) = \frac{\partial u_i}{\partial t}(t)$$

Eulerian:
forces, speed at fixed position in space



\+ *Spatial operator on field (ex. divergence, laplacian)*

$$a(x,t) = \frac{\mathrm{d}u}{\mathrm{d}t}(x,t) = \frac{\partial u}{\partial t}(x,t) + (u \cdot \nabla u)(x,t)$$

Convective/material derivative

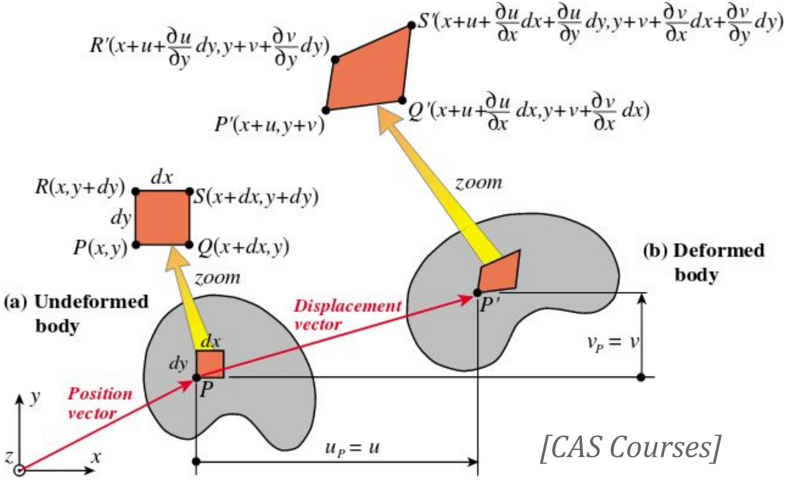
Continuous mechanics

Fundamental equation of dynamics

Case of solids (ex. elasticity, plasticity)

$$\rho \frac{\mathrm{d}u_i}{\mathrm{d}t}(t,x) = \rho(x) F_i(t) + \frac{\partial \sigma_{ij}}{\partial x_j}$$

Deformation w/r rest state (structured)



[CAS Courses]

Case of fluids : Navier Stokes

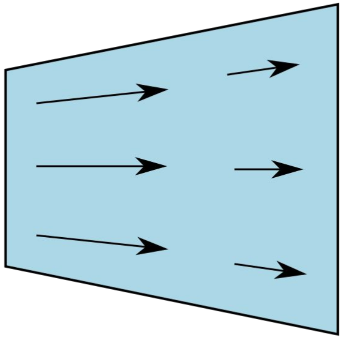
$$\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\,\mathbf{u} - \eta \nabla^2 \mathbf{u} = g - \nabla p / \rho$$

viscosity *forces*

$$\nabla \cdot \mathbf{u} = 0$$

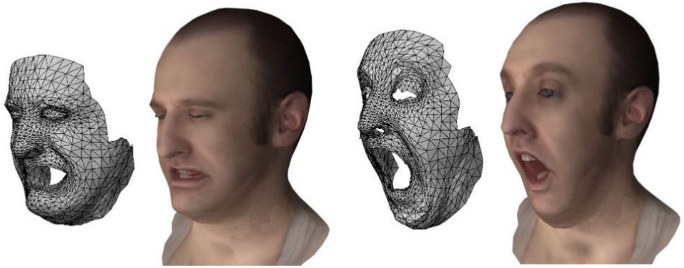
incompressibility

Unstructured deformation

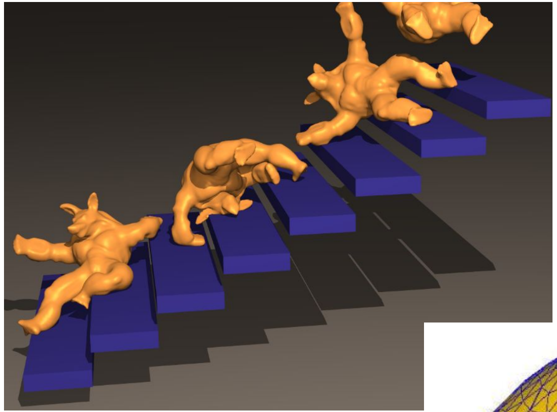


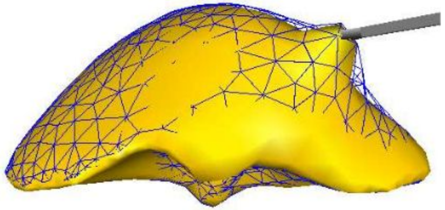
Continuous mechanics examples

Model reduction



[Barbic et al, 16]

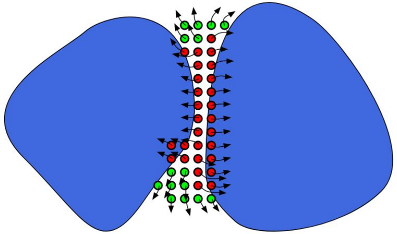




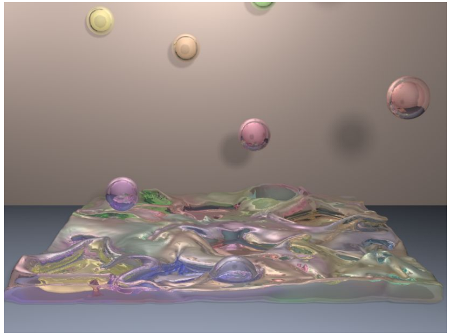
[Sifakis, Fedkiw, et al 05]

[Delingette, Ayache et al, 00]

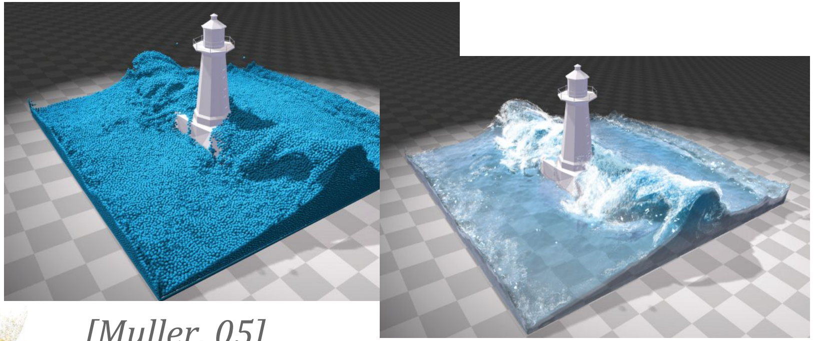
Continous mechanics: FLuids: Grid + particles



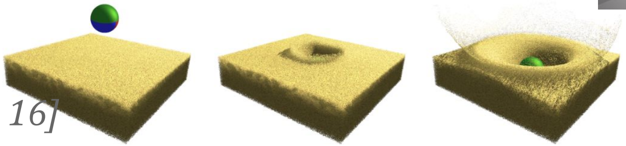
[Barteil, Adam, 06]



SPH method



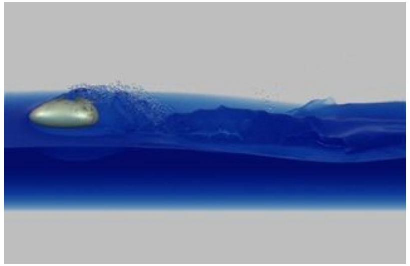
[Muller, 05]



[Davier, Desboubes, 16]

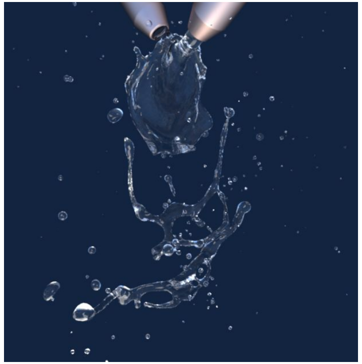
PIC/FLIP method

[Foster, Fedkiw, 01]



Fluid mesh

[Wojtan et al., 17]



Numerically solving differential equation

Numerical solution of ODE

$$U'(t) = \mathcal{F}(t, U(t))$$ (arbitrary dimension, non linear)

Explicit Euler

$$U(t + \Delta t) = U(t) + \Delta t\ \mathcal{F}(t, U(t))$$

1st Order
Conditionally stable

Implicit Euler

$$U(t + \Delta t) = U(t) + \Delta t\ \mathcal{F}(t + \Delta t, U(t + \Delta t))$$

Unknown

1st Order
Unconditionally stable

Numerical solution of ODE

Implicit method: stable, but impractical for non linear problems

Special case of linear problem: $U'(t) = \mathrm{A}\, U(t) + b$

Implicit Euler: $$U(t + \Delta t) = U(t) + \Delta t\ (\mathrm{A}\, U(t + \Delta t) + b)$$

$$U(t + \Delta t) = (\mathrm{Id} - \Delta t\ \mathrm{A})^{-1}\, (U(t) + \Delta t\ b)$$

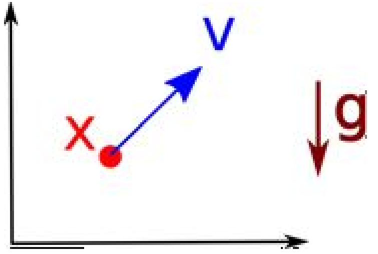
Solve a linear system at each step

For non linear system: Linearize with respect to Δt, then use implicit Euler

Exemple: Free fall

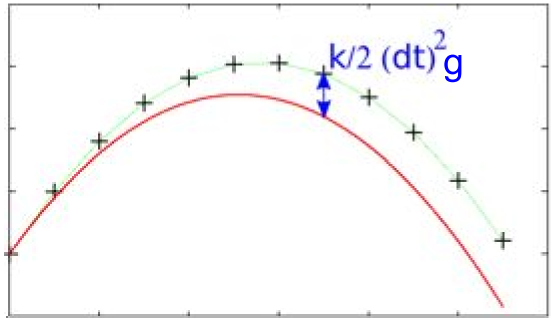
Motion: $x'(t)=v(t)$
$v'(t)=g$

IC: $x(t=0)=x_0$
$v(t=0)=v_0$



Explicit Euler $u'(t) = \mathrm{A}u(t) + \mathrm{b}(t)$ $\quad u(t) = \begin{pmatrix} x(t) \\ v(t) \end{pmatrix}$ $\quad \mathrm{A} = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}$

$\mathrm{b}(t) = \begin{pmatrix} 0 \\ g \end{pmatrix}$



$$\begin{cases} x^{k+2} = 2x^{k+1} - x^k + (\Delta t)^2 g \\ x^0 = x_0 \\ x^1 = x_0 + \Delta t v_0 \end{cases}$$

$$x(t = k\Delta t) = x_0 + (k\Delta t)\, v_0 + \frac{k(k-1)}{2}(\Delta t)^2 g$$

Real solution:

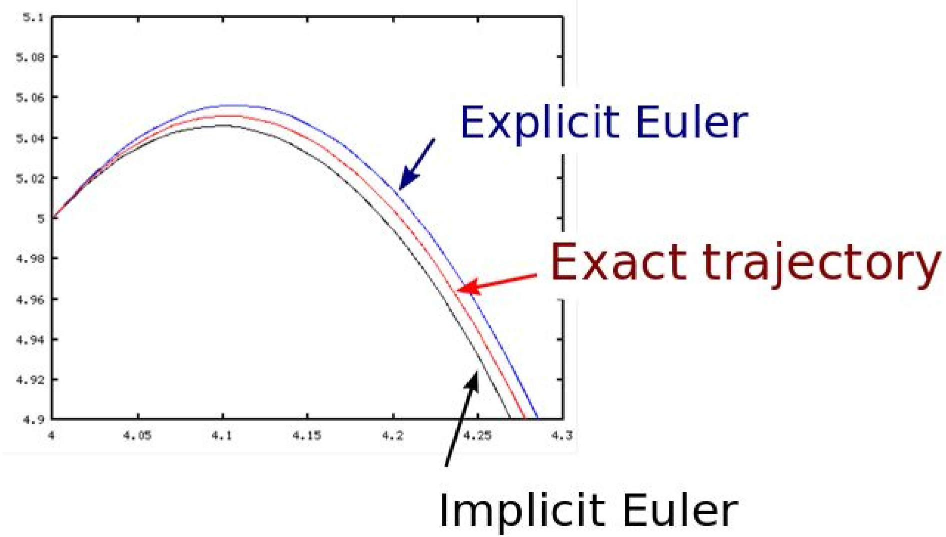
$$\tilde{x}(t = k\Delta t) = x_0 + (k\Delta t) v_0 + \frac{1}{2}(k\Delta t)^2 g$$

Exemple: Free fall

$$u^{k+1} = (\mathrm{I} - \Delta t \mathrm{A})^{-1} \left(u^k + \Delta t \,\mathrm{b}\right)$$

$$\begin{cases} x^{k+2} = 2x^{k+1} - x^k + (\Delta t)^2 g \\ x^0 = x_0 \\ x^1 = x_0 + \Delta t \, v_0 + (\Delta t)^2 g \end{cases}$$

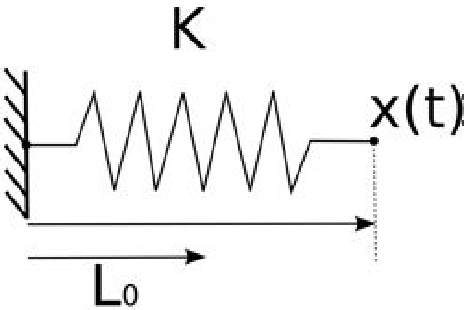
$$x(k\Delta t) = x_0 + (k\Delta t)\, v_0 + \frac{k(k+1)}{2}(\Delta t)^2 g$$



Exemple: 1D Spring mass system

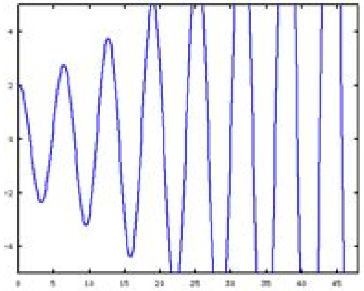
Spring force

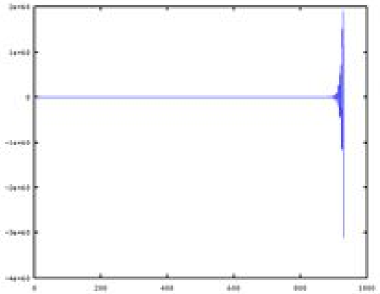
$$F(t) = K(L_0 - x(t))$$



Explicit Euler:

$$x^{k+2} = 2x^{k+1} - \left(1 + (\Delta t)^2 \frac{K}{m}\right) x^k + (\Delta t)^2 \frac{K}{m} L_0$$





Expect:

$$x(t) = A \sin(\omega t + \varphi)$$

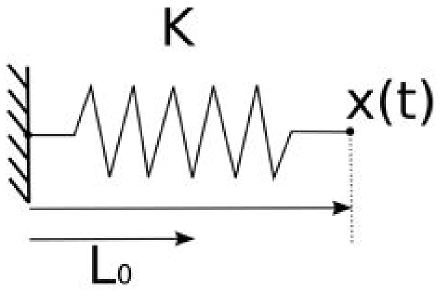
Exemple: 1D Spring mass system

Spring force

$$F(t) = K(L_0 - x(t))$$

Add fluid damping

$$F_d(t) = -\mu\, v(t)$$



New equation for explicit Euler:

$$x^{k+2} - \left(2 - \frac{\mu}{m}\Delta t\right) x^{k+1} + \left(1 + (\Delta t)^2 \frac{K}{m} - \frac{\mu}{m}\Delta t\right) x^k = (\Delta t)^2 \frac{K}{m} L_0$$

Conditionnaly stable

Large K => Stiff springs

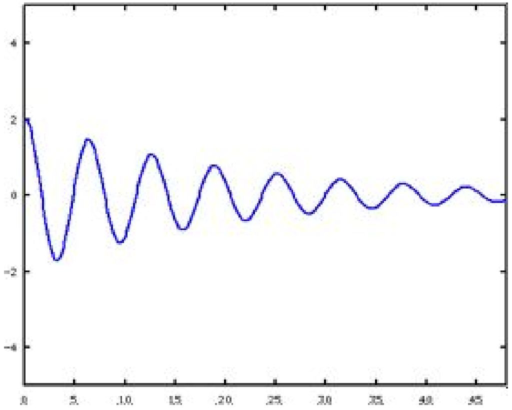
$\boxed{\text{Stiff ODE}}$

Exemple: 1D Spring mass system

$$\underbrace{\begin{pmatrix} 1 & -\Delta t \\ -\Delta t \frac{K}{m} & 1 \end{pmatrix}}_{\text{M}} u^{k+1} = u^k + \Delta t \begin{pmatrix} 0 \\ K \frac{L_0}{m} \end{pmatrix}$$

Eigenvalues of M^{-1} $\quad 1 - \Delta t \sqrt{\frac{K}{m}} < 1$

Unconditionally stable



Numerical solution of ODE

Runge-Kutta

$$U(t+\Delta t) = U(t) + \Delta t \sum_i b_i k_i$$

$$\begin{aligned} k_1 &= \mathcal{F}(t, U(t)) \\ k_2 &= \mathcal{F}(t + \alpha_2 \Delta t, U(t) + \Delta t(\beta_{21} k_1)) \\ k_3 &= \mathcal{F}(t + \alpha_3 \Delta t, U(t) + \Delta t(\beta_{31} k_1 + \beta_{32} k_2)) \\ &\ldots \end{aligned}$$

+ *Explicit formulation, better stability (still conditionally stable)*

ex. 2nd Order $\alpha_2 = 0.5, \beta_{21} = 0.5$

$$U(t+\Delta t) = U(t) + \Delta t \left[\mathcal{F}\left(t + \frac{\Delta t}{2}, U(t) + \frac{\Delta t}{2} \mathcal{F}(t, U(t)) \right) \right]$$

Commonly used: Dormand-Prince (aka ode45) - 5th order

Exemple: Free fall

$$F : u^k = (x^k, v^k) \mapsto \begin{pmatrix} v^k \\ g \end{pmatrix}$$

$$u^{k+1/2} = \begin{pmatrix} x^k + \Delta t/2\, v^k \\ v^k + \Delta t/2\, g \end{pmatrix}$$

$$F(u^{k+1/2}) = \begin{pmatrix} v^k + \frac{\Delta t}{2} g \\ g \end{pmatrix}$$

$$x^{k+1} = x^k + \Delta t\ v^k + \frac{(\Delta t)^2}{2} g$$
$$v^{k+1} = v^k + \Delta t g$$

$$\begin{cases} x^{k+2} = 2x^{k+1} - x^k + (\Delta t)^2 g \\ x^0 = x_0 \\ x^1 = x_0 + \Delta t\ v_0 + \frac{(\Delta t)^2}{2} g \end{cases} \Rightarrow \boxed{x^k = x_0 + (k\,\Delta t)\ v_0 + \frac{(k\Delta t)^2}{2} g}$$

Exact solution

Numerical solution of PDE

Category: Linear equation of degree 2

In analogy with:

Parabolic

$$\frac{\partial f}{\partial t} = a\frac{\partial^2 f}{\partial x^2}$$

$$y = ax^2$$

Diffusion equation, smooth solution

Hyperbolic

$$a\frac{\partial^2 f}{\partial t^2} - b\frac{\partial^2 f}{\partial x^2} = 0$$

$$\frac{y^2}{a^2} - \frac{x^2}{b^2} = 0$$

Wave equation, propagation with finite speed

Elliptique

$$a\frac{\partial^2 f}{\partial x^2} + b\frac{\partial^2 f}{\partial y^2} = 0$$

$$\frac{y^2}{a^2} + \frac{x^2}{b^2} = 0$$

Poisson, Laplace equation

Numerical solution of PDE: general approaches

Finite difference (FDM)

Values only defined at the nodes

\+ Easy to implement

\- Only for structured grid

Finite elements (FEM)

Continuous basis functions (shape functions)

\+ General shape, accuracy

\- Computationally expensive

Finite volume (FV)

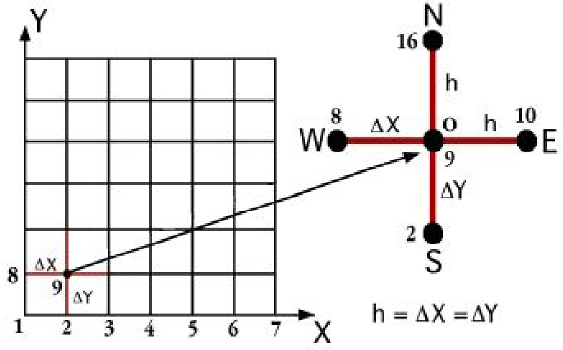
Piecewise constant values

\+ Adapted to advection, non smooth solution

Finite difference

Discretize the derivatives in space on a grid
=> ODE

Solve the ODE using explicit/implicit approaches



ex. Diffusion equation

$$\frac{\partial U}{\partial t}(x,t) = \triangle U(x,t)$$

$$\frac{\partial U}{\partial t}(x,t) = \frac{1}{(\Delta x)^2}\Big(U(x+\Delta x,t) - 2\,U(x,t) + U(x-\Delta x,t)\Big)$$

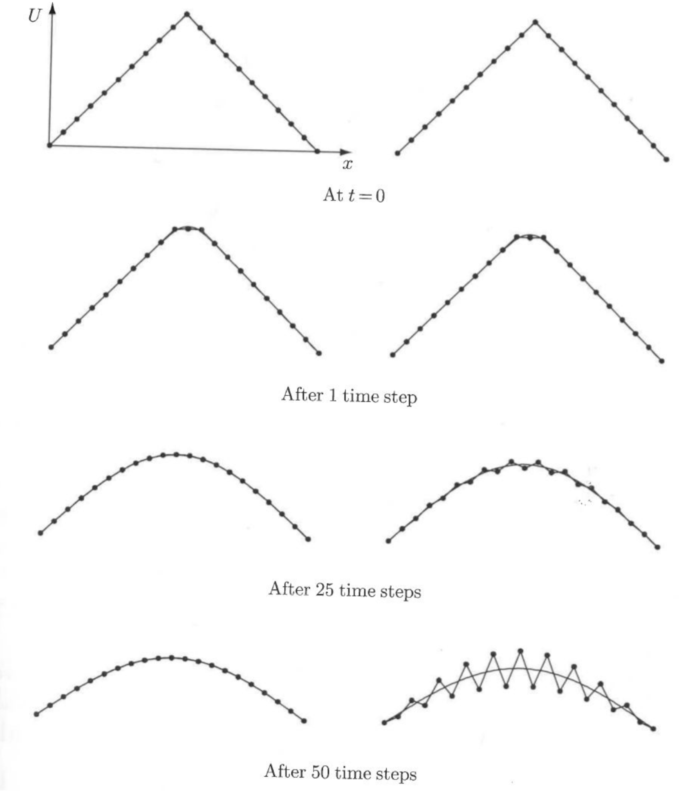
Finite difference

Explicit Euler

Conditionally stable

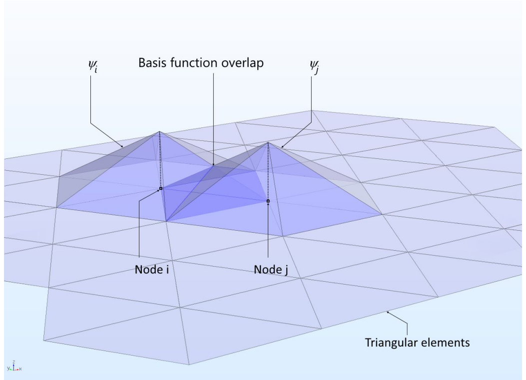
$$\frac{\Delta t}{(\Delta x)^2} \leq \frac{1}{2}$$

t=0.0012 t=0.0013



Finite elements method

- Convert the equation into a weak form
- Discretize the domain, set the basis shape functions
- Express the linear system on the weights of the shape functions
- Solve a sparse linear system



Finite Element Method

ex. Poisson equation $\triangle u(x,y) + f(x,y) = 0$

$u = g_D \text{ on } \partial\Omega_D$ *Dirichlet BC*

$\frac{\partial u}{\partial n} = g_N \text{ on } \partial\Omega_N$ *Neumann BC*

Weak formulation: $\int_\Omega \triangle u\, v + f v = 0$ v: smooth "test" function

$$\int_\Omega \nabla u \cdot \nabla v - \int_{\partial\Omega} v \frac{\partial u}{\partial n}$$

Integration by parts, divergence theorem

=> $$\int_\Omega \nabla u \cdot \nabla v = \int_\Omega v\, f + \int_{\partial\Omega_N} v\, g_N$$

Discretization: Look for solution written as $u_h = \sum_{j=1}^{n} \mathbf{u}_j \phi_j + \sum_{j=n+1}^{n+\partial n} \mathbf{u}_j \phi_j$ (Galerkin FEM)

$(\phi_i)_i$ Shape functions, basis of the "test" functions v

Finite Element Method

Sparse linear system: $\mathrm{K}\,\mathbf{u} = \mathbf{f}$

$$k_{ij} = \int_{\Omega} \nabla\phi_j \cdot \nabla\phi_i$$

$$f_i = \int_{\Omega} \phi_i\, f + \int_{\partial\Omega_N} \phi_i\, g_N - \sum_{j=n+1}^{n+n_\partial} \mathbf{u}_j \int_{\Omega} \nabla\phi_j \cdot \nabla\phi_i$$

K: Stiffness matrix