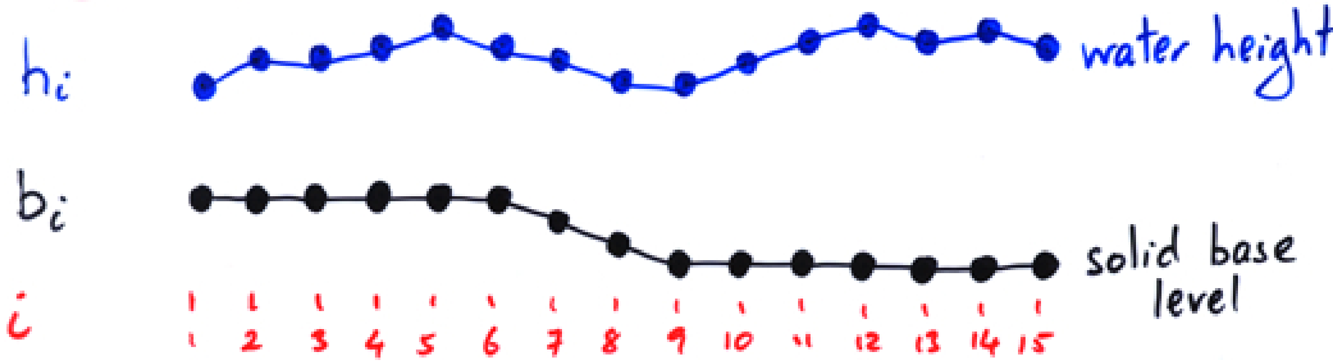
Modelling Water

- Real water has lots of interesting properties
  - smooth (laminar) and turbulent flow
  - waves
  - splashes
  - droplets
  - surface tension
  - wetting (surfaces)
  - mixing (with other liquids)
  - cavitation (when you drop a solid in)

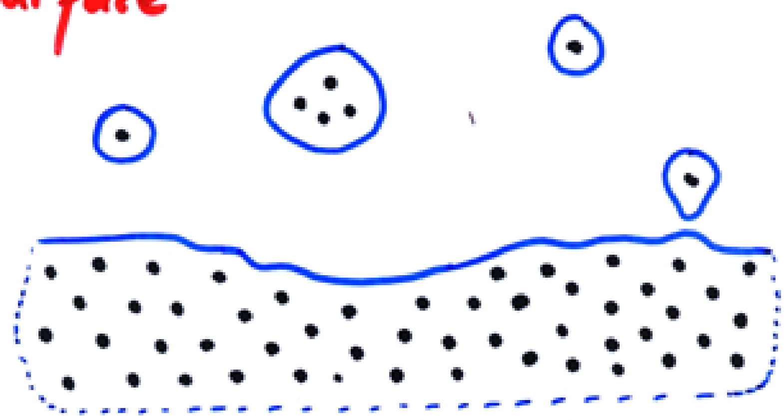
- DAMTP has a research group studying how to model water
  - ∴ graphics people shouldn't expect to find an easy and accurate model
  - → but might find an "OK" model with "reasonable" computational requirements
  - → "OK" means that it <u>looks</u> right (or close enough to right...)

THREE WAYS YOU COULD MODEL WATER

height field



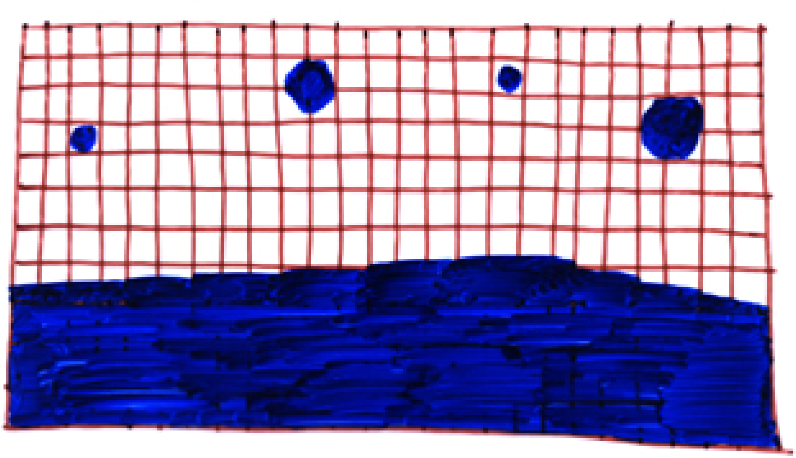
implicit surface



voxel space

a voxel can be:

- full
- empty
- surface



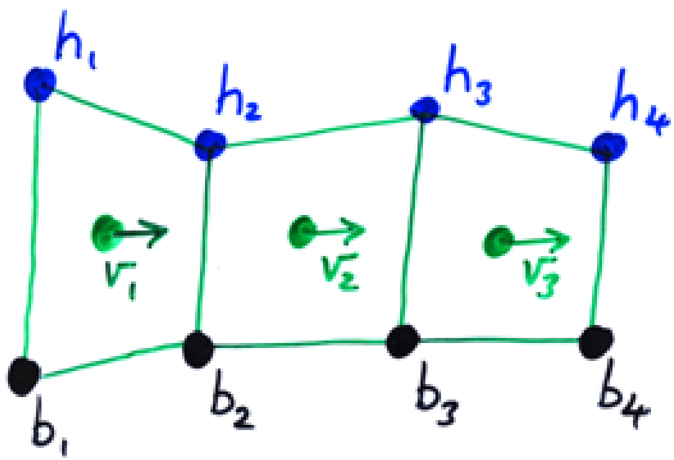
Height Field Model

- cannot model splashes, breaking waves, droplets, ..
- based on Kass & Miller "Rapid, stable fluid dynamics for CG" SIGGRAPH 90
- simple implementation in Java at:

http://www.cl.cam.ac.uk/~nad/applts/AGwater.html

---

each column of water has a velocity



$d_i = h_i - b_i$

depth

Basic Equations

① $$\underbrace{\frac{\partial v}{\partial t}}_{\text{acceleration}} = \underbrace{-v\frac{\partial v}{\partial x}}_{\text{velocity difference} \to \text{force}} \underbrace{- g\frac{\partial h}{\partial x}}_{\text{pressure difference} \to \text{force}} \qquad (F=ma)$$

② $$\underbrace{\frac{\partial d}{\partial t}}_{\text{change in depth}} = \underbrace{-\frac{\partial}{\partial x}(vd)}_{\text{amount of water flowing}} \qquad \text{(volume conservation)}$$

Kass & Miller assume:

- adjacent columns have (roughly) the same velocity

  ∴ approximate 1 by

  $$\boxed{1'} \quad \frac{\partial v}{\partial t} = -g \frac{\partial h}{\partial x}$$

- depths do not change rapidly (?)

  ∴ approximate 2 by

  $$\boxed{2'} \quad \frac{\partial h}{\partial t} = -d \frac{\partial v}{\partial x}$$

Combine $\boxed{1'}$ & $\boxed{2'}$ to get

$$\boxed{3} \quad \frac{\partial^2 h}{\partial t^2} = gd \frac{\partial^2 h}{\partial x^2}$$

which (it transpires) is the
"shallow water Navier-Stokes simplification"

Kass & Miller go on to produce a discrete approximation and solve a tridiagonal system of linear equations to advance from one timestep to the next because "... the wave equation $\boxed{3}$ is a notoriously bad example for explicit differential equation methods such as Euler"

NEVERTHELESS...

we can implement it using Euler evaluation

① $$v_i' = v_i + \Delta t \left(v_i \frac{v_{i+1} - v_i}{\Delta x} - g \frac{h_{i+1} - h_i}{\Delta x} \right)$$

② $$h_i' = h_i + \Delta t \left([v_{i-1} - v_i] \frac{h_i - b_i}{\Delta x} \right)$$

- [my implementation includes several tweaks to these equations].

What goes wrong?

if $$v_i > \frac{\Delta x}{\Delta t}$$

then the algorithm takes more water from column i than is actually present

- as v_i approaches $\frac{\Delta x}{\Delta t}$ the Euler solution diverges from the accurate solution

- one solution: decrease Δt
- another: use a better solver (e.g. Runge-Kutta)
- & another: solve a system of equations (e.g. Kass & Miller)

Solving the full* Navier-Stokes equation

Variables		constants	
position	(x, y, z)		
velocity	(u, v, w)		
time	t	dynamic viscosity	η
pressure	p	density	ρ
body force	(F_x, F_y, F_z)		

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z} = 0$$

$$\rho\left(\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} + w\frac{\partial u}{\partial z}\right) = -\frac{\partial p}{\partial x} + \eta\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 u}{\partial z^2}\right) + F_x$$

$$\rho\left(\frac{\partial v}{\partial t} + u\frac{\partial v}{\partial x} + v\frac{\partial v}{\partial y} + w\frac{\partial v}{\partial z}\right) = -\frac{\partial p}{\partial y} + \eta\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2}\right) + F_y$$

$$\rho\left(\frac{\partial w}{\partial t} + u\frac{\partial w}{\partial x} + v\frac{\partial w}{\partial y} + w\frac{\partial w}{\partial z}\right) = -\frac{\partial p}{\partial z} + \eta\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2}\right) + F_y$$

- Simplify as much as possible (e.g. assume no body force) & discretise
- Solve on a voxel grid for each time step

* this is actually Navier-Stokes for an irrotational, incompressible liquid — it gets worse if you remove these 2 constraints