CS475 Parallel Programming

Differentiation and Integration

Wim Bohm
Colorado State University

Phenomena

- Physics: heat, flow, space, time
- Mathematics: continuous functions, (partial) differential equations
- Computer science: Discrete simulation of physical phenomena through

Finite Difference Methods

Differentials

- Physical phenomena like the flow of heat are modeled with differentials:

$$\frac{df}{dx} = \lim_{\Delta x \to o} \frac{f(x+\Delta x) - f(x)}{\Delta x}$$

- A differential describes **rate of change**, e.g. velocity is the rate of change of position, v = df/dx, and acceleration is the rate of change of velocity, a = dv/dx, which is the second derivative (the derivative of the derivative of position)

Partial Differential Equations

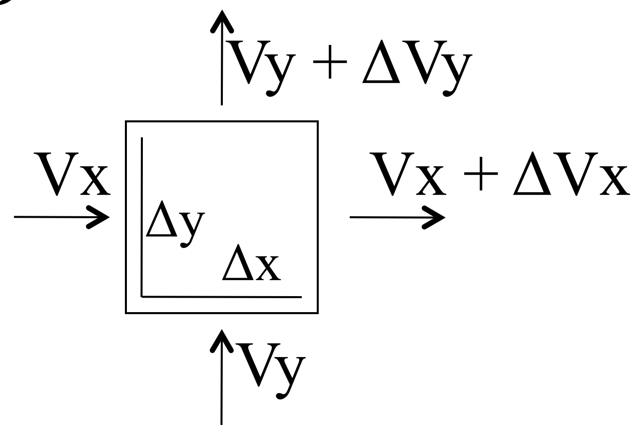
- Partial differential equations are differential equations in higher dimensions expressed in a coordinate system, e.g in 2D:

$$\frac{\partial u}{\partial x} \, and \, \frac{\partial u}{\partial y}$$

describe the change of u in the x and y direction.

Laplace

- Laplace described physical phenomena in 2 and 3D, e.g. heat in 2D



- In X direction: cell receives heat $Vx\Delta y$, loses heat $(Vx+\Delta Vx)\ \Delta y$, hence $\Delta Vx\ \Delta y$ heat removed
- Similarly, in Y direction: $\Delta Vy\ \Delta x$ heat removed

trick

$$\Delta Vx \Delta y = \frac{\Delta Vx}{\Delta x} \Delta x \Delta y \to \frac{\partial Vx}{\partial x} \Delta x \Delta y$$

$$\Delta Vy \Delta x = \frac{\Delta Vy}{\Delta y} \Delta x \Delta y \to \frac{\partial Vy}{\partial y} \Delta x \Delta y$$

$$Combined\ \ loss : (\frac{\partial Vx}{\partial x} + \frac{\partial Vy}{\partial y}) \Delta x \Delta y$$

More tricks

Heat conservation law: $$\frac{\partial Vx}{\partial x} + \frac{\partial Vy}{\partial y} = 0$$

Feynman: heat flows at a rate proportional to the temperature (u) gradient

$$Vx = -k\frac{\partial u}{\partial x}$$

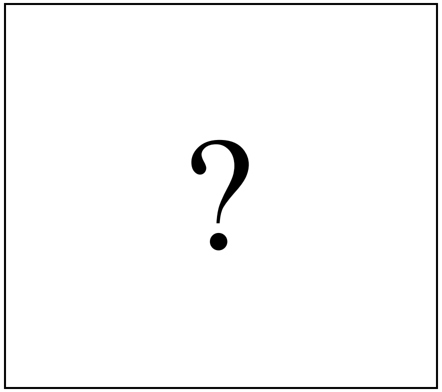
$$Vy = -k\frac{\partial u}{\partial y}$$

These two combined: $$\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$$

heat

- Heat at boundary known
- What is the heat inside?
- Discretize it

	n	
w	c	e
	s	



- $u_c = u(x,y)$, $u_n = u(x,y+h)$, $u_s = u(x,y-h)$,
  $u_e = u(x+h,y)$, $u_w = u(x-h,y)$

Taylor series: function approximation

We can express a function in terms of its derivatives,

The more derivatives the closer (at least that was the wisdom until chaos got discovered (Pointcare)).

$$f(x+h) = f(x) + \sum_{i=1}^{k} \frac{1}{i!} f^{i}(x)$$

Taylor approximation

$$u_e = u(x+h,y) = u_c + h\frac{\partial u}{\partial x} + \frac{1}{2}h^2\frac{\partial^2 u}{\partial x^2}$$

$$u_w = u(x-h,y) = u_c - h\frac{\partial u}{\partial x} + \frac{1}{2}h^2\frac{\partial^2 u}{\partial x^2}$$

$$u_e + u_w = 2u_c + h^2\frac{\partial^2 u}{\partial x^2}$$

$$u_e + u_w + u_s + u_n = 4u_c + h^2\frac{\partial^2 u}{\partial x^2} + h^2\frac{\partial^2 u}{\partial y^2}$$

Taylor + Heat conservation

Taylor: $u_e + u_w + u_s + u_n = 4u_c + h^2 \frac{\partial^2 u}{\partial x^2} + h^2 \frac{\partial^2 u}{\partial y^2}$

Heat conservation: $\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} = 0$

therefore: $u_c = (u_n + u_s + u_e + u_w) / 4$

Thermal equilibrium: **temperature at (x,y) is average of surrounding temperatures**

Solving the heat equation

- nxn grid: we could have a direct solution
  - nxn equations with nxn unknowns
  - **Too Complex!**
- iterative solution: relaxation
  - Keep doing $u_c = (u_n + u_s + u_e + u_w)/4$ at every point until equilibrium reached
- Jacobi version: ping pong with two arrays
  - Nice parallelism, slow convergence
- Gauss-Seidel: one array, use latest version
  - More complex data dependence, faster convergence

CS view

- Nearest neighbor computation, checkerboard or block row partitioning
- Exchange of data along borders
- Trick: overlapping areas (see e.g. Quinn Ch. 13)
  - Re-computation
  - Reduced communication frequency
  - Potentially more complicated communication pattern

Integration

- Differentiation: finding rate of change in f(x)

$$y = x^n, \qquad \frac{dy}{dx} = nx^{n-1}$$

$$y = z + w, \qquad \frac{dy}{dx} = \frac{dz}{dx} + \frac{dw}{dx}$$

$$y = z.w, \qquad \frac{dy}{dx} = w\frac{dz}{dx} + z\frac{dw}{dx}$$

$$y = u / v, \qquad \frac{dy}{dx} = (v\frac{du}{dx} - u\frac{dv}{dx}) / v^2$$

- Integration: finding surface under f(x)

Integration

$$\int_a^b f(x)dx = F(b) - F(a) \quad where \quad F'(x) = f(x)$$

$$\int_a^b x^n dx = \frac{b^{n+1} - a^{n+1}}{n+1}$$

Numerical integration

- Approximate f(x) and derive simple formula for integral
  - Linear: two points, quadratic: three, etc.
  - Two approaches: open vs, closed:

    open: points don't include a and b

    closed: points include a and b

    different math
- Approximate in a number of intervals
  - Applying any form of above approximation methods

Trapezoidal rule

- $I \sim (b-a).((f(a)+f(b))/2)$
- Intervals: x_0, $x_1 = x_0+h$, $x_2 = x_0+2h$, …. $X_{n,}$ $h = (b-a)/n$

$$I \sim h(\,(f(x_0)+f(x_1))/2 \;+\; \ldots + ((f(x_{n-1})+f(x_n)/2))$$

$$= \qquad (b-a)\frac{f(x_0)+2\sum_{i=1}^{n-1} f(x_i)+f(x_n)}{2n}$$

Better approximations

- Either: more points (increase n)
- or higher order polynomials
  - E.g. Simpsons rule uses quadratic approximation over 3 points

$$I = \frac{h}{3}(f(x_0) + 4f(x_1) + f(x_2))$$

  - Intervals:

$$I = \frac{b-a}{3n}(f(x_0) + 4\sum_{i=1,3,5..}^{n-1} f(x_i) + 2\sum_{i=2,4,6..}^{n-1} f(x_i) + f(x_n))$$

Iterative / adaptive approach

- Iterate with smaller and smaller segments

  until $I_i \sim I_{i+1}$

  $h_1=(b-a)/n$ $\quad$ $h_{2etc.} = (h_1)/2$ etc.
- Error: use relative error

$$\varepsilon_r = \frac{present \quad approx - previous \quad approx}{present \quad approx} .100\%$$

$$\leq (0.5 * 10^{2-n})\%$$

n: number of significant digits

Recursive approach: adaptive quadrature

```
trap(left,right) = { return (right-left)*(f(left)+f(right))/2;}
tol = (0.5*exp(10,2-n));
area(left,right,est) ={
   mid=(left+right)/2;
   a1=trap(left,mid); a2=trap(mid,right);
   newest = a1+a2;
   if(abs((newest-est)/newest)<tol)
      return newest;
   else return area(left,mid,a1) +area(mid,right,a2)
}
```