Numerical Modelling in FORTRAN day 12

Paul Tackley, 2014

Today's Goals

1. **Fortran and numerical methods**: Review and discussion. New features of fortran 2003 & 2008.
2. **Projects**: Define goals, identify information needed from me.
3. **Exercises**: Solve any remaining problems.
4. Due date for all remaining things: Friday 13 February 2015

Fortran Review

- History
  - 53rd birthday (of first compiler)
  - versions 66, 77, 90, 95, 2003, 2008
- Variables
  - types: real, integer, logical, character, complex
  - Implicit types and implicit none
  - Initialisation
  - parameters
  - defined types
  - precision (32 vs 64 bit etc.)

Fortran Review (2)

- Arrays
  - fixed vs. automatic vs. allocatable vs. assumed shape
  - index ranges
  - data, reshape
  - built-in array algebra
- Input/Ouput
  - open, print, read, write, close
  - formats
  - iostat, rewind, status
  - namelist
  - binary vs. ascii, direct vs. sequential

Fortran Review (3)

- Flow control structures
  - do loops (simple, counting, while), exit, cycle
  - if…elseif…endif blocks
  - select case…
  - forall
  - where

Fortran Review (4)

- Functions and subroutines (procedures)
  - Intrinsic functions: mathematical, conversion, …
  - internal (contains), external (& interface blocks), in modules
  - array functions
  - recursive functions and result statement
  - named arguments
  - optional arguments
  - generic procedures
  - overloading
  - user-defined operators
  - save

Fortran Review (5)

- Modules
  - use
  - public and private variables and functions
  - =>
- Pointers
  - to variables, arrays, array sections, areas of memory, in defined types
- Optimization
  - maximize use of cache and pipelining
  - loops most important
  - 90/10 rule (focus on bottlenecks)
  - avoid branches, maximize data locality, unroll, tiling,…
- Libraries, makefiles

New features of Fortran 2003

- For a detailed description see: ftp://ftp.nag.co.uk/sc22wg5/N1601-N1650/N1648.pdf
- For a summary: http://www.fortran.bcs.org/2007/jubilee/newfeatures.pdf
- Enhancements to derived types
  - Parameterized derived types
  - Improved control of accessibility
  - Improved structure constructors
  - Finalizers

New features of Fortran 2003 (2)

- Object-oriented programming support
  - Type extension and inheritance
  - Polymorphism
  - Dynamic type allocation
  - Type-bound procedures
- Data manipulation enhancements
  - Allocatable components
  - Deferred type parameters
  - VOLATILE attribute
  - Explicit type specification in array constructors & allocate statements
  - Extended initialization expressions
  - Enhanced intrinsic procedures

New features of Fortran 2003 (3)

- Input/output enhancements
  - Asynchronous transfer
  - Stream access
  - User specified transfer ops for derived types
  - User specified control of rounding
  - Named constants for preconnected units
  - FLUSH
  - Regularization of keywords
  - Access to error messages
- Procedure pointers

New features of Fortran 2003 (4)

- Support for the exceptions of the IEEE Floating Point Standard
- Interoperability with the C programming language
- Support for international characters (ISO 10646 4-byte characters...)
- Enhanced integration with host operating system
  – Access to command line arguments,environment variables, processor error messages
- Plus numerous minor enhancements

Numerical methods review

- Approximations
  - Concept of discretization
  - Finite difference approximation
  - Treatment of boundary conditions
- Equations
  - diffusion equation
  - Poisson's equation
  - advection-diffusion equation
  - Stokes equation and Navier-Stokes equation
  - streamfunction and streamfunction-vorticity formulation
  - Pr, Ra, Ek
  - initial value vs. boundary value problems

Numerical methods Review (2)

- Timestepping
  - stability (timestep, advection scheme)
  - explicit vs. implicit, semi-implicit
  - upwind advection
- Solvers
  - direct
  - iterative (relaxation), multigrid method
  - Jacobi vs. Gauss-Seidel vs. red-black
- Parallelisation
  - Domain decomposition
  - Message-passing and MPI

Discussion

- We have focussed on one application (constant viscosity convection)
- Can apply same techniques to other applications / physical problems

Example: Darcy Equation

(groundwater or partial melt flow)

$$\vec{u} = -\frac{k}{\eta}\left(\vec{\nabla}P - g\rho\hat{\vec{y}}\right)$$

u=volume/area/time="Darcy velocity" (really a flux)

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

k=permeability (related to porosity and interconnectivity)
η=viscosity of fluid, ρ=density of fluid, P=pressure in fluid

Simplify and rearrange

Take divergence: velocity is eliminated

$$\vec{\nabla} \cdot \vec{u} = -\vec{\nabla} \cdot \left(\frac{k}{\eta} \vec{\nabla} P \right) + \vec{\nabla} \cdot \left(\frac{k}{\eta} g \rho \hat{\vec{y}} \right) = 0$$

2nd order equation for pressure

$$\vec{\nabla} \cdot \left(\frac{k}{\eta} \vec{\nabla} P \right) = g \frac{\partial}{\partial y} \left(\frac{\rho k}{\eta} \right)$$

Combine k and η into hydraulic resistance R

$$\vec{\nabla} \cdot \left(\frac{1}{R} \vec{\nabla} P \right) = g \frac{\partial}{\partial y} \left(\frac{\rho}{R} \right)$$

A Poisson-like equation for pressure => easy to solve using existing methods

Example 2: Full elastic wave equation

Most convenient to solve as coupled first-order PDEs

F=ma for a continuum: $$\rho \frac{\partial \vec{v}}{\partial t} = \nabla \cdot \underline{\underline{\sigma}}$$

Hooke's law for an **isotropic** material $$\sigma_{ij} = \lambda \delta_{ij} \varepsilon_{kk} + 2\mu \varepsilon_{ij}$$

Where $$\varepsilon_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)$$ u=displacement of point from equilibrium position

μ =shear modulus. λ, μ are known as Lamé constants

Sometimes written
Using K=bulk modulus $$\sigma_{ij} = K\varepsilon_{kk}\delta_{ij} + 2\mu(\varepsilon_{ij} - \tfrac{1}{3}\varepsilon_{kk}\delta_{ij})$$

Resulting equations to solve using finite differences

Variables: 3 velocity components and 6 stress components

On staggered grid (same as viscous flow modelling)

$$\frac{\partial}{\partial t} v_x = \frac{1}{\rho}\left(\frac{\partial}{\partial x}\sigma_{xx} + \frac{\partial}{\partial y}\sigma_{xy} + \frac{\partial}{\partial z}\sigma_{xz}\right)$$

$$\frac{\partial}{\partial t}\sigma_{xx} = (\lambda + 2\mu)\frac{\partial v_x}{\partial x} + \lambda\left(\frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z}\right)$$

$$\frac{\partial}{\partial t} v_y = \frac{1}{\rho}\left(\frac{\partial}{\partial x}\sigma_{xy} + \frac{\partial}{\partial y}\sigma_{yy} + \frac{\partial}{\partial z}\sigma_{zy}\right)$$

$$\frac{\partial}{\partial t}\sigma_{yy} = (\lambda + 2\mu)\frac{\partial v_y}{\partial y} + \lambda\left(\frac{\partial v_x}{\partial x} + \frac{\partial v_z}{\partial z}\right)$$

$$\frac{\partial}{\partial t} v_z = \frac{1}{\rho}\left(\frac{\partial}{\partial x}\sigma_{zx} + \frac{\partial}{\partial y}\sigma_{zy} + \frac{\partial}{\partial z}\sigma_{zz}\right)$$

$$\frac{\partial}{\partial t}\sigma_{zz} = (\lambda + 2\mu)\frac{\partial v_z}{\partial z} + \lambda\left(\frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y}\right)$$

$$\frac{\partial}{\partial t}\sigma_{xy} = \mu\left(\frac{\partial v_x}{\partial y} + \frac{\partial v_y}{\partial x}\right)$$

$$\frac{\partial}{\partial t}\sigma_{xz} = \mu\left(\frac{\partial v_x}{\partial z} + \frac{\partial v_z}{\partial x}\right)$$

$$\frac{\partial}{\partial t}\sigma_{yz} = \mu\left(\frac{\partial v_y}{\partial z} + \frac{\partial v_z}{\partial y}\right)$$

Mathematical classification of 2nd-order PDEs: Elliptical, hyperbolic, parabolic

Elliptical	$\nabla^2 \phi = f$	e.g., Poisson
Parabolic	$\dfrac{\partial \phi}{\partial t} \propto \nabla^2 \phi$	e.g., diffusion
Hyperbolic	$\nabla^2 \phi \propto \dfrac{1}{V^2} \dfrac{\partial^2 \phi}{\partial t^2}$	e.g., wave eqn.

We' ve done them all!

Waves: $$\frac{\partial^2 P}{\partial t^2} = v^2 \nabla^2 P$$

Fluids:

$$\frac{1}{\Pr}\left(\frac{\partial \vec{v}}{\partial t} + \vec{v}\cdot\vec{\nabla}\vec{v}\right) = -\vec{\nabla}P + \vec{\nabla}\cdot\left(\eta\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right)\right) + \frac{1}{Ek}\vec{\Omega}\times\vec{v} + Ra.T\hat{g}$$

$$\frac{\partial T}{\partial t} + \vec{v}\cdot\nabla T = \nabla\cdot(k\nabla T) \qquad \nabla\cdot\vec{v} = 0$$

Percolation: $$\vec{\nabla}\cdot\left(\frac{k}{\eta}\vec{\nabla}P\right) = g\frac{\partial}{\partial y}\left(\frac{\rho k}{\eta}\right)$$

note the many similar terms!

N-body: $$\frac{d^2\vec{x}_i}{dt^2} = \vec{a}_i \qquad \vec{a}_i = G\sum_{j=1}^{n,j\neq i}\frac{m_j}{\left|\vec{x}_j - \vec{x}_i\right|^2}\cdot\frac{(\vec{x}_j - \vec{x}_i)}{\left|\vec{x}_j - \vec{x}_i\right|}$$

Projects

- Make a list of what everyone is doing
- Define goals (e.g., test cases) that determine when project is finished
- For some projects people need more information from me: make a list of this & I will send this week
- Available for consulation in most of January & February

END!

FORTRAN