

BCS Fortran SG Meeting
11th June 2009

Meet the Competition
Matlab and other animals

Dr John Pelan PGDip Laws CPhys CITP

J.Pelan@ucl.ac.uk

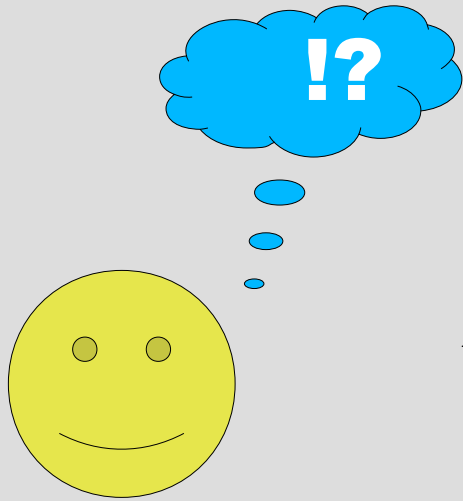
Overview

- Obvious?
- Not a sales pitch!
- Getting your “problem” into a computer!
- The choices we have
- Making the selection
- Quick look at Matlab (license server permitting!)
- Conclusions

Computational “X”

- Where $X = \{ \text{Physics, Chemistry, Engin...} \}$
- Theory - “Model” to be tested or queried
- Step #1 -- expressed mathematically
- Step #2 -- expressed computationally

The Mapping Processes



$$A(i,j) = A(i+1,j) +$$

$$\rho \left(\frac{\partial \mathbf{v}}{\partial t} + \mathbf{v} \cdot \nabla \mathbf{v} \right) = -\nabla p + \nabla \cdot \mathbb{T} + \mathbf{f},$$



The Menu

- Fortran
- C, C++, Java
- Basic, Perl
- Python
- Mathematica, Maple, Sage
- Matlab, Octave, R, SPSS, AVS, IDL
- Spreadsheets (!!)

Selecting? #1

- Existing corpora / legacy code
- Your existing skill set
- Local expertise
- Resources – tools at hand
- Time / Turnaround
- My boss/supervisor told me to!

Selecting? #2

- Good match to the problem at hand
- Ease of coding, maintenance, debugging
- Robust – well tested, good tools
- Run Portable
- Data Portable
- Enduring – lifetime of project
- Uses the available hardware optimally

Matlab History

- MATLAB ("matrix laboratory")
- Created in the late 1970s by Cleve Moler, (CS Professor, University of New Mexico)
- Access to LINPACK and EISPACK without having to learn Fortran.
- Commercial product sold by Mathworks
- Runs on Windows, Linux, Mac OS X...

Matlab - Basics

- Interactive environment
- Instant gratification!

```
>> A = [ 1 2 3 ; 4 5 6]
```

```
A =
```

```
1 2 3
```

```
4 5 6
```

```
>> A(2,3)
```

```
ans =
```

```
6
```

Matlab – More Basics

```
>> B = A(1:2,2:3)
```

```
B =
```

```
    2    3  
    5    6
```

```
>> B^2
```

```
ans =
```

```
    19    24  
    40    51
```

Matlab – Some inbuilt demos

- Switch to Matlab!

Matlab - Features

- Lots of toolboxes ('libraries')
- NAG
- Parallelism (threads and message passing)
- Can plug-in optimised libraries at the backend – e.g. BLAS, LAPACK etc.
- Has a compiler 'mex'
- Has published API so can interface your own code & exchange data.

Matlab - Disadvantages

- It costs – lots, although student version ~£90
- Lots of toolboxes – each one at a price!
- Portable?
- Not a good fit for the largest problems
- Open source equivalent - octave

Conclusions

- It is quick and easy to learn
- Fast-turnaround, good for development
- May not use hardware efficiently out-of-box or scale to the largest problems
- Not too dissimilar to Fortran
- Expensive
- Requires everyone in project uses it
- Don't dismiss it but Fortran not dead yet.