Cloud and Aerosol Research on Massively-parallel Architectures (CARMA)

Jon Gibson, NAG Manchester

HECToR

RESEARCH COUNCILS UK

What you're about to hear

- The dCSE project
- The LEM code
- Name that cloud!
- Some "interesting science"
  - Or how this project could save the world!
- The Parallel Iterative Solver
- Testing, optimisation and future work.

HECToR

RESEARCH COUNCILS UK

CARMA

- 12 month dCSE project with Dr Paul Connolly of Manchester University and Dr Alan Gadian of Leeds University.
  - Thanks also to Alan for the slides on the VOCALS project!
- Three main objectives
  - To implement a parallel iterative pressure solver within the LEM (Large Eddy Model) code
  - To update the existing LEM code to use FFT libraries
  - To parallelise the code ACPIM for use on HECToR

HECToR

RESEARCH COUNCILS UK

The Aerosol-Cloud-Precipitation Interaction Model (ACPIM)

- Developed and validated at the UoM and the Institute of Meteorology and Climate Reseach in Germany
- A detailed process-scale model for studying the growth and evolution of cloud from aerosols
  - Will be used to test our ability to simulate ice formation on aerosols in the atmosphere

HECToR

RESEARCH COUNCILS UK

New Science

- The parallelisation of ACPIM
  - will provide the research community with a model capable of simulating very detailed processes on cloud-scale systems, bridging the gap between laboratory work, cloud modelling and analysis of field data.

The Large Eddy Model (LEM)

- Code developed by The Met Office
- Widely used in academic research
- High resolution numerical model used to simulate cloud-scale and microscale atmospheric processes
  - e.g. aerosol-cloud interactions
  - Modified Navier-Stokes with parameterisations for turbulence, microphysical processes and radiation

HECToR

RESEARCH COUNCILS UK

Basic Equation Set

- The LEM solves the following basic equation set, shown using tensor notation:

$$\frac{Du_i}{Dt} = -\frac{\partial}{\partial x_i}\left(\frac{p'}{\rho_s}\right) + \delta_{i3}B' + \frac{1}{\rho_s}\frac{\partial \tau_{ij}}{\partial x_j} - 2\epsilon_{ijk}\Omega_j u_k \quad (1)$$

$$\frac{\partial}{\partial x_i}(\rho_s u_i) = 0 \quad (2)$$

$$\frac{D\theta}{Dt} = \frac{1}{\rho_s}\frac{\partial h_i^\theta}{\partial x_i} + \left(\frac{\partial \theta}{\partial t}\right)_{mphys} + \left(\frac{\partial \theta}{\partial t}\right)_{rad} \quad (3)$$

$$\frac{Dq_n}{Dt} = \frac{1}{\rho_s}\frac{\partial h_i^{q_n}}{\partial x_i} - \left(\frac{\partial q_n}{\partial t}\right)_{mphys} \quad (4)$$

Parallel Iterative Pressure Solver

- Implementation of an iterative method for solving Poisson's Equation
  - Removes the need for periodic boundary conditions
  - and so extends the science that the LEM can be used for.

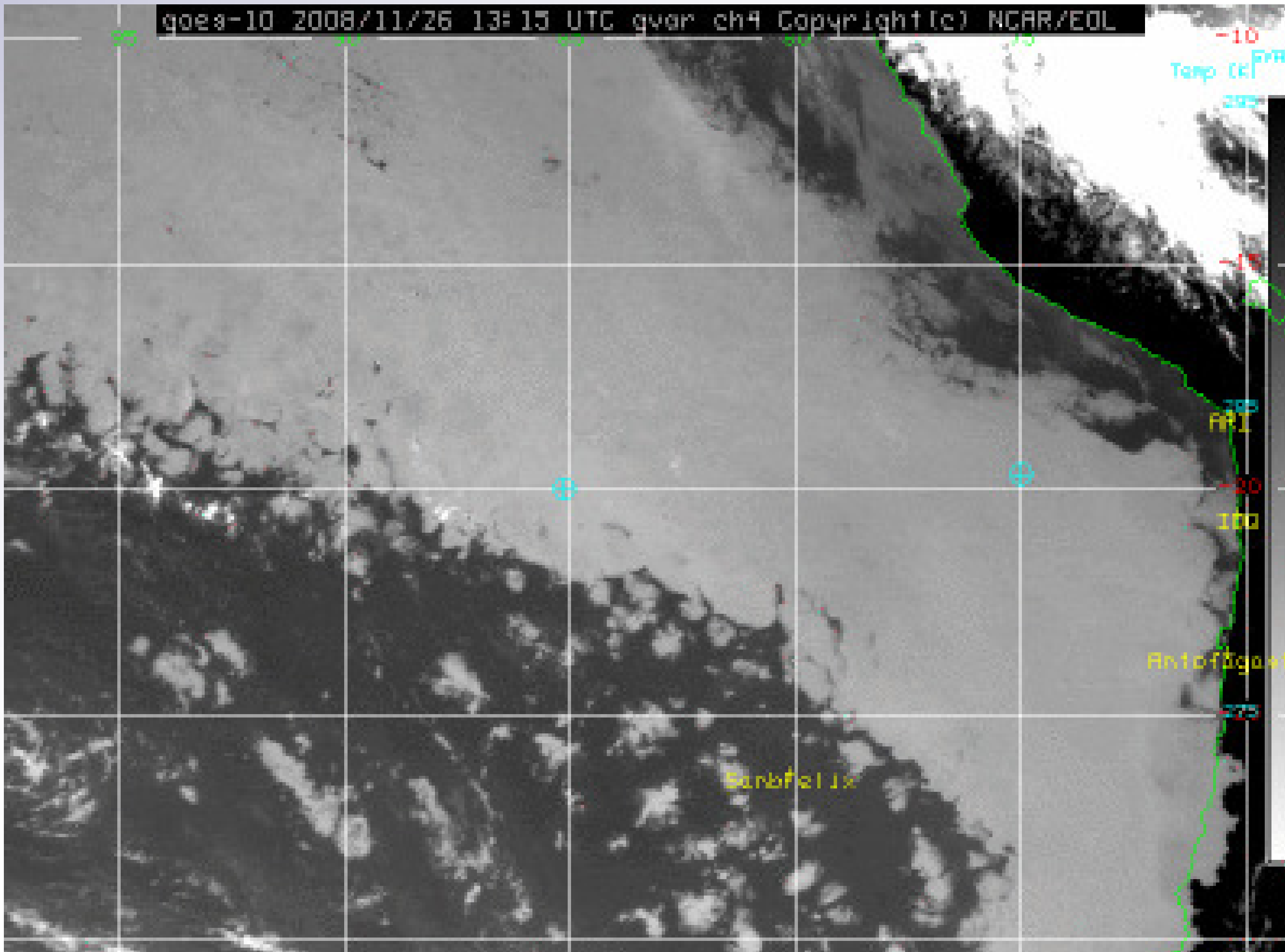
New Science

- With the new parallel iterative solver
  - The code could be used to study lightning generation and the electrification of clouds
    - Leading to a lightning predictor tool
  - Non-periodic solutions are needed for the multi-national VOCALS (Vamos Ocean Cloud Atmosphere Land Study) project, an investigation of stratocumulus clouds in the south-east Pacific and their effect on climate.

HECToR

RESEARCH COUNCILS UK

Name That Cloud!



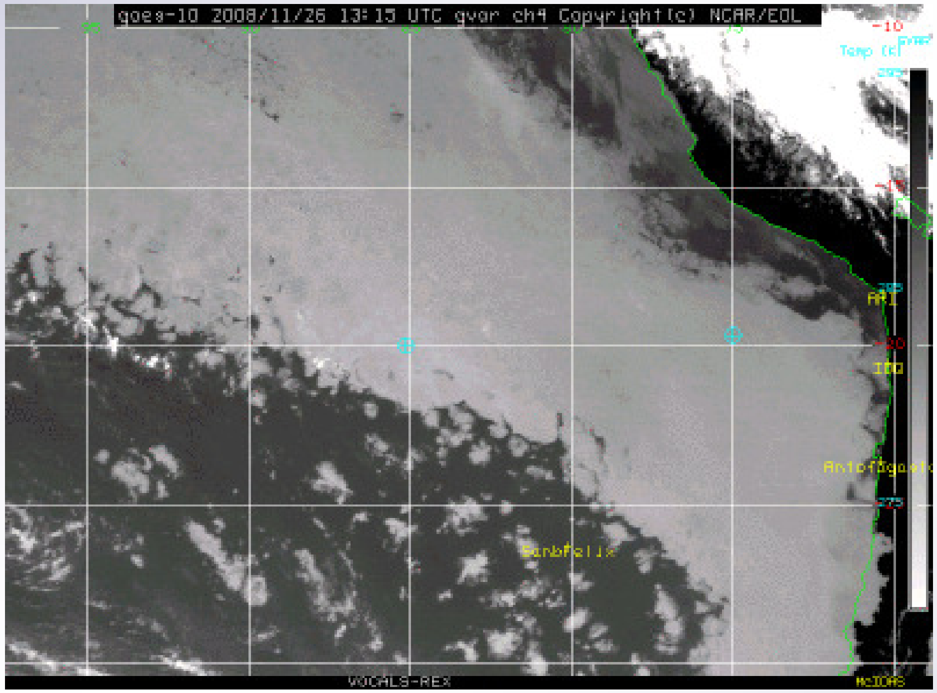
HECToR

RESEARCH COUNCILS UK

- Stratocumulus Clouds cover more than 30% of Ocean Surface
- Stratocumulus Clouds have a high reflectance, which depends on droplet number and mean droplet size.

<u>Twomey Effect</u>.

Lots of small drops, produce whiter clouds.



Smaller glass beads are whiter.

November 2008, during VOCALS. Stratocumulus clouds off the coasts of Chile and Peru (US and UK scientists)

HECToR

RESEARCH COUNCILS UK

Technique *(Latham, Nature, 1990)*

1. Latham et al. Phil Trans of
2. Salter et al. Nov 2008.

To disseminate “natural” sea-water 0.8 micron droplets at ocean surface, into the boundary layer.

These ascend via thermals and turbulence to cloud-base levels in sufficient quantities and with sufficient salt-mass to dominate as CCN, thereby increasing N, in a quantitatively controllable manner, and mono dispersed size distribution.

Switching off the sprayers, returns to the status quo within weeks

Stephen Salter’s design, picture courtesy John McNeill

Support for idea

- Natural droplet creation at ocean surface.
- Bubble-bursting, white-capping
- NaCl droplets are effective CCN
- Ship tracks, fires, industrial pollution etc.

HECToR

Cloud Whitening

RESEARCH COUNCILS UK

Some LEM code details

- Mainly Fortran77 but with some Fortran90
  - All memory statically allocated
- Communication written using GCOM
  - A wrapper around a wrapper around a subset of MPI
- “Version control” using nupdate
  - Decks, comdecks and nupdate

HECToR

RESEARCH COUNCILS UK

Running the LEM

- Jobs compiled and submitted via scripts
  - lemsub and runXT4
- Set-up jobs and chain jobs
  - Jobs can resubmit themselves, reading the input from the previous job

Decomposition of the Domain

- The domain is made up of IIP×JJP×KKP grid-points.
- Integration of the equations only requires information from up to 2 grid-points away so the domain can be decomposed into sub-domains with overlapping halo regions.
- A centred-difference time integration scheme is used.

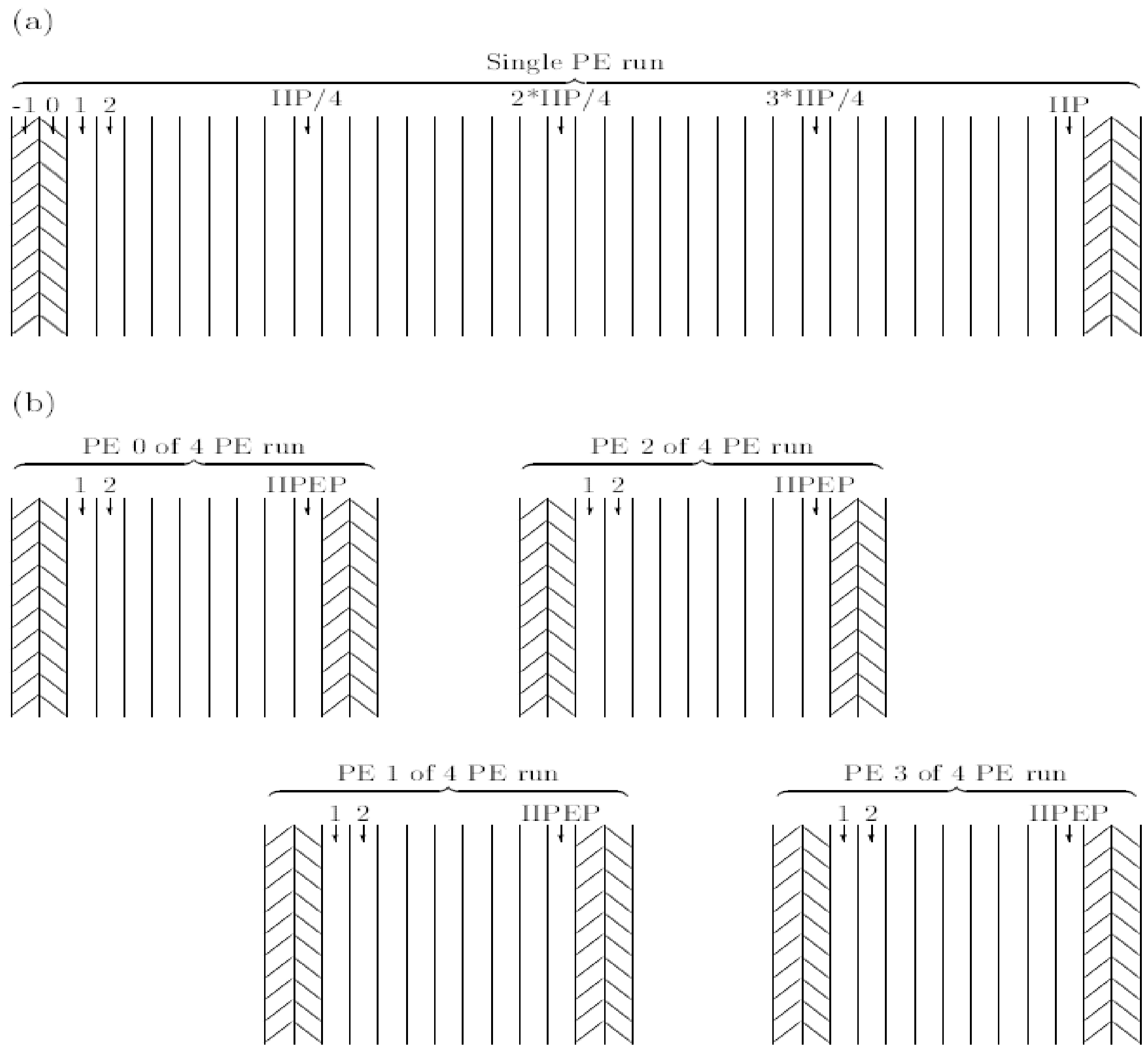
Decomposition of the Domain

- The LEM integrates the field in a succession of 2D slices in the y-z plane, moving along the x-direction.
- The domain is decomposed in the x-direction so that each PE holds IIP/NPES slices, along with a two-slice halo region for each neighbouring sub-domain.

HECToR

RESEARCH COUNCILS UK

The Decomposed Domain



The Pressure Solver

- Calculation of the pressure term in the system equations requires the solution of a Poisson-like elliptic equation

$$\frac{\partial}{\partial x_i}\left[\rho_s \frac{\partial}{\partial x_i}(p'/\rho_s)\right] = \frac{\partial}{\partial x_i}(\rho_s s_i)$$

where

$$s_i = \delta_{i3}B' - u_j \frac{\partial u_i}{\partial u_j} + \frac{1}{\rho_s}\frac{\partial \tau_{ij}}{\partial x_j} - 2\varepsilon_{ijk}\Omega_j u_k$$

HECToR

RESEARCH COUNCILS UK

The Pressure Solver

- This equation had so far been solved via the use of Fourier transforms.
  - Required the use of periodic boundary conditions.
- Using an iterative solver in its place would remove the need for periodic boundary conditions and so enable new science.

A Parallel Iterative Solver

- A good choice of algorithm was found to be the BiConjugate Gradient Stabilized method, or BiCGStab.
- A quick look at the algorithm will highlight what is required to implement it.
- Note that we are solving an equation of the form

$$\mathbf{Ax} = \nabla^2 \mathbf{x} = \mathbf{b}$$

HECToR

RESEARCH COUNCILS UK

The BiCGStab Algorithm

1. Compute $\mathbf{r}_0 = \mathbf{b} - \mathbf{A}\mathbf{x}_0$ for an initial guess $\mathbf{x}_0$
2. Initialise $\bar{\mathbf{r}} = \mathbf{r}_0$; $\mathbf{p}_0 = \mathbf{v}_0 = 0$ and $\rho_{-1} = \alpha_0 = \omega_0 = 1$
3. DO i = 1, ITMAX

$$\rho_{i-1} = \bar{\mathbf{r}}^T \bullet \mathbf{r}_{i-1}$$

$$\beta_{i-1} = \left(\frac{\rho_{i-1}}{\rho_{i-2}}\right)\left(\frac{\alpha_{i-1}}{\omega_{i-1}}\right)$$

$$\mathbf{p}_i = \mathbf{r}_{i-1} + \beta_{i-1}\left[\mathbf{p}_{i-1} - \omega_{i-1}\mathbf{v}_{i-1}\right]$$

$$\text{solve}\, \mathbf{M}\hat{\mathbf{p}} = \mathbf{p}_i$$

$$\mathbf{v}_i = \mathbf{A}\hat{\mathbf{p}}$$

The BiCGStab Algorithm

$$\alpha_i = \frac{\rho_{i-1}}{\bar{\mathbf{r}}^T \bullet \mathbf{v}_i}$$

$$\mathbf{s} = \mathbf{r}_{i-1} - \alpha_i \mathbf{v}_i$$

$$\text{solve } \mathbf{M}\hat{\mathbf{s}} = \mathbf{s}$$

$$\mathbf{t} = \mathbf{A}\hat{\mathbf{s}}$$

$$\omega_i = \frac{\mathbf{t}^T \bullet \hat{\mathbf{s}}}{\mathbf{t}^T \bullet \mathbf{t}}$$

$$\mathbf{x}_i = \mathbf{x}_{i-1} + \alpha_i \hat{\mathbf{p}} + \omega_i \hat{\mathbf{s}}$$

$$\mathbf{r}_i = \hat{\mathbf{s}} - \omega_i \mathbf{t}$$

Check for convergence; continue if necessary.

END DO

Calculating Ax=∇^2x

- Assume a 7 point stencil and use the following finite difference relation

$$\nabla^2 \mathbf{f} = \frac{1}{\Delta x^2} f(x+\Delta x, y, z) + \frac{1}{\Delta x^2} f(x-\Delta x, y, z)$$

$$+ \frac{1}{\Delta y^2} f(x, y+\Delta y, z) + \frac{1}{\Delta y^2} f(x, y-\Delta y, z)$$

$$+ \frac{1}{\Delta z^2} f(x, y, z+\Delta z) + \frac{1}{\Delta z^2} f(x, y, z-\Delta z)$$

$$-2\left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} + \frac{1}{\Delta z^2}\right) f(x, y, z)$$

HECToR

RESEARCH COUNCILS UK

Preconditioner

- Use Jacobi method
  - The element-based formula being:

$$x_i^{(k+1)} = \frac{1}{a_{ii}}\left(b_i - \sum_{j \neq i} a_{ij} x_j^{(k)}\right)$$

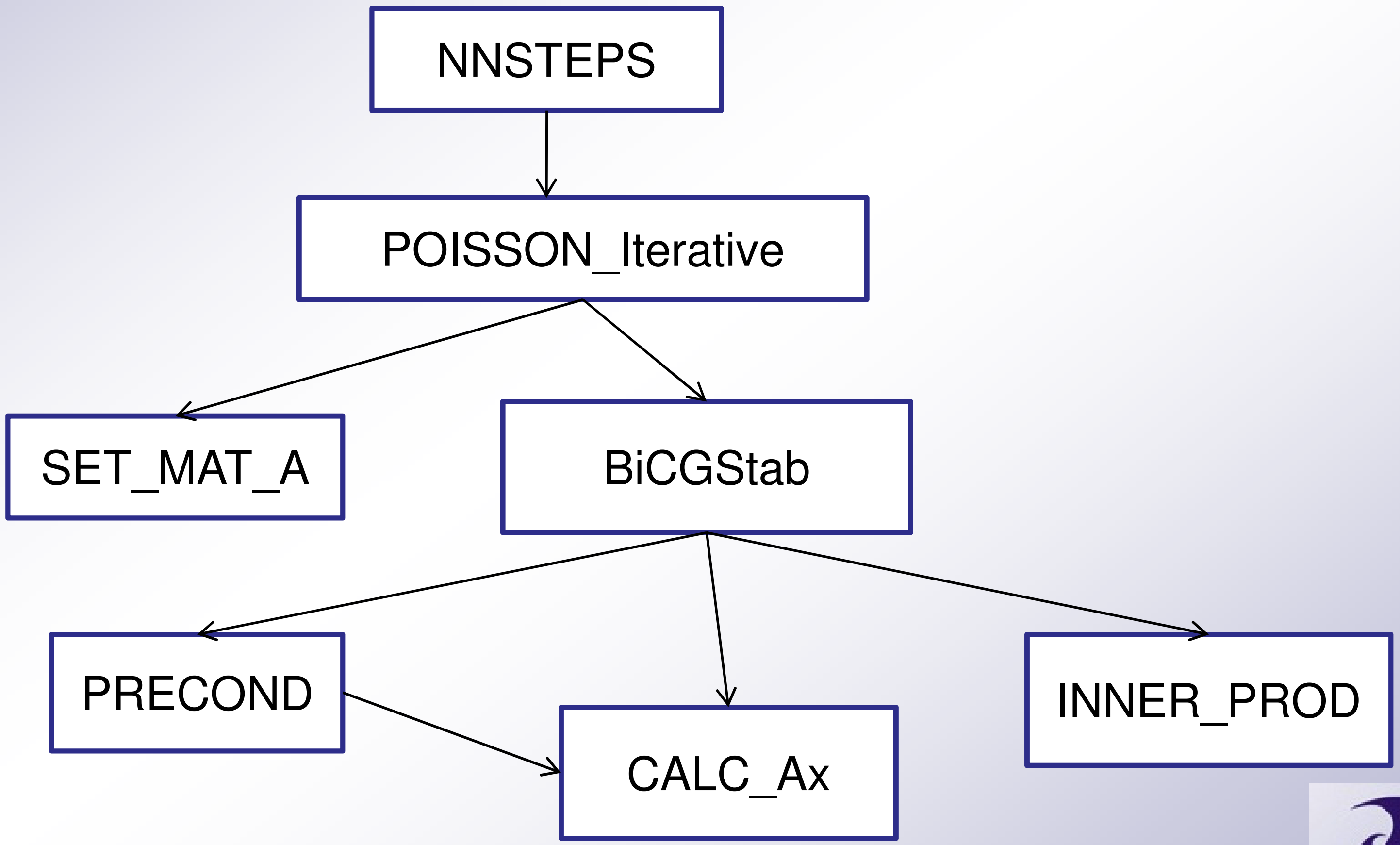
Parallel BiCGStab

- Assume the same decomposition along the x-direction.
- There are a number of dot products whose results are required by all processes.
- Calculating expressions of the form $\mathbf{Ax}$ requires halo-exchange between neighbouring processes.
  - Periodic boundary conditions in x and y assumed at the moment.

HECToR

RESEARCH COUNCILS UK

Call Graph for Iterative Solver



Testing the Implementation

- Generate a set of known pressure values, ϕ
- Calculate the input source terms using

  $$\mathbf{f} = \nabla^2 \phi$$
- Use the iterative solver with $\mathbf{f}$ as the input.
- Does it give us back our original set of pressure values, ϕ ?

HECToR

RESEARCH COUNCILS UK

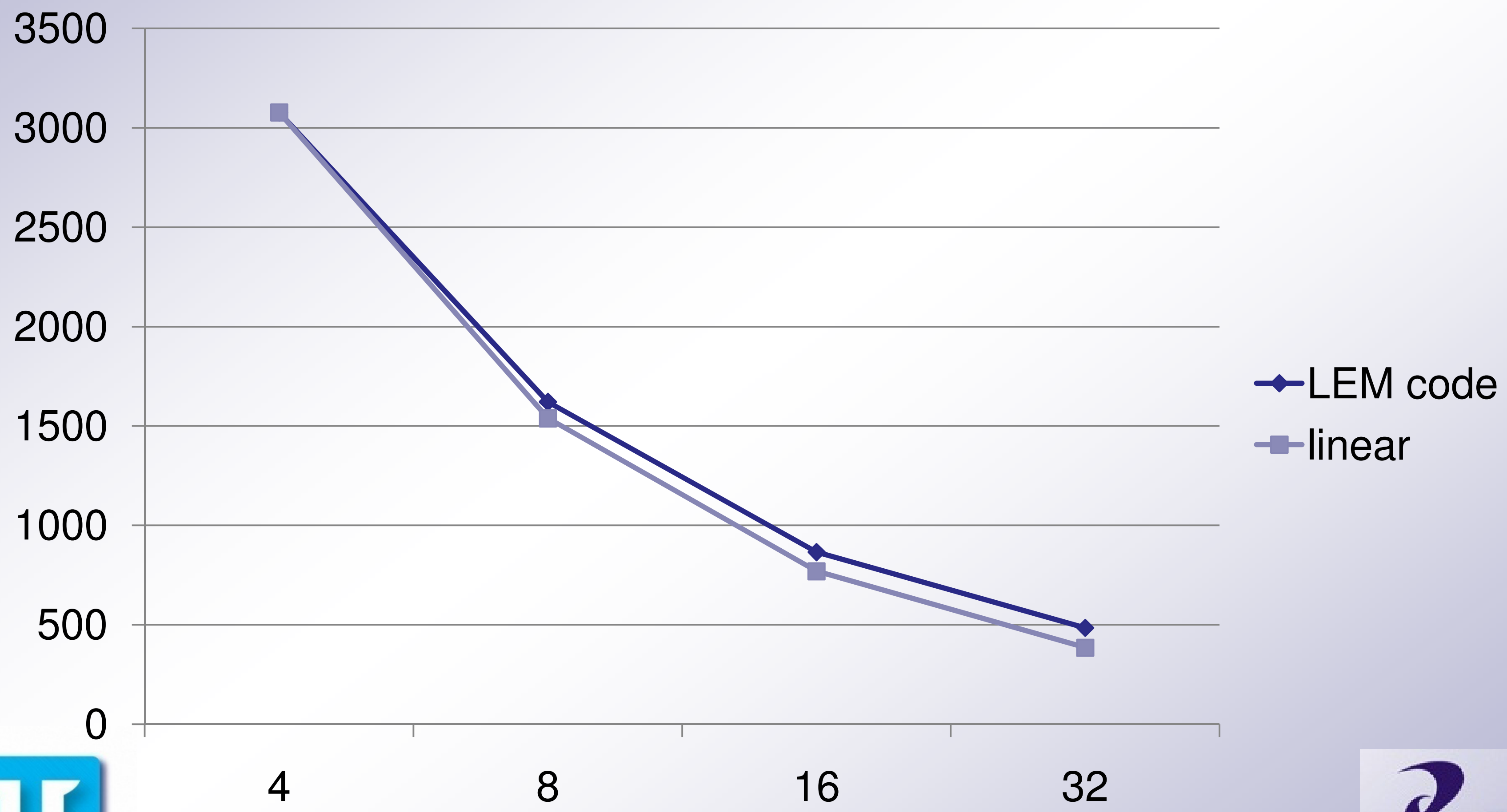
Optimising the Implementation

- Used CrayPat and output from -Minfo/-Mneginfo PGI compiler flags
  - Modified code to improve cache re-use
  - Sometimes compiler did the wrong thing
  - Improved the communication...

GCOM, MPL or MPI?

- Existing communication in the code is very inefficient.
- I used GCOM initially...
- ...but to give better communication performance, now call the intermediate MPL library layer.

Wot, no scaling graph?



Potential Future Work

- Move to a 2D domain decomposition
- Improve the communication in the rest of the code
- Modify the memory use

Questions?