Introduction to Distributed-Memory Parallel MM5

John Michalakes
michalak@ucar.edu
PSU/NCAR MM5 Tutorial

Outline

- Parallelism in MM5
- Performance
- Building and using the code
- "Same source" approach
- Linux cluster experiences
- Additional information

Parallelism in MM5

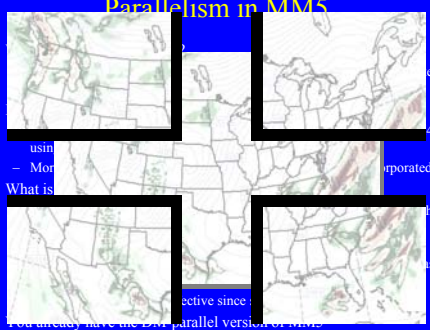
- What is meant by "parallel"?
  - Increase computational and memory resources available for larger, faster runs by having more than one computer work on the problem
- Isn't MM5 already parallel?
  - Yes, the model has been able to run shared-memory parallel since MM4 using Cray Microtasking directives
  - More recently, standardized OpenMP directives have been incorporated
- What is DM-parallelism? Why?
  - Processors store part of model domain in local memory, not shared with other processors, and work together on a problem by exchanging messages over a network
  - "Scalable" because it eliminates bottlenecks on shared resources such as bus or memory
  - Possibly also more cost effective since systems can be "commodity"
- You already have the DM-parallel version of MM5

Parallelism in MM5

- What is meant by "parallel"?
  - Incr ger, faster runs
- Isn't M
  - Yes ce MM4 usin
  - Mor rporated
- What is
  - Proc red with othe g mes
  - "Sca s such as bus
  - Possibly also more cost effective since systems can be "commodity"
- You already have the DM-parallel version of MM5

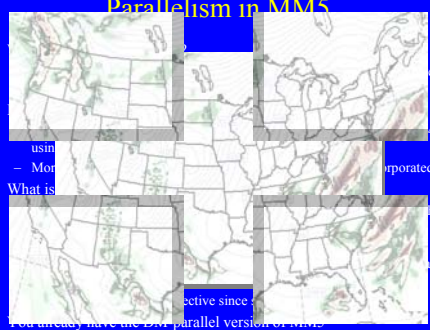
Parallelism in MM5

- What is
  - Mor rporated
- What is
  - ective since
- You already have the DM-parallel version of MM5



Parallelism in MM5

- What is
  - Mor rporated
- What is
  - ective since
- You already have the DM-parallel version of MM5



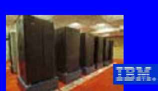
Parallelism in MM5

MM5 platforms

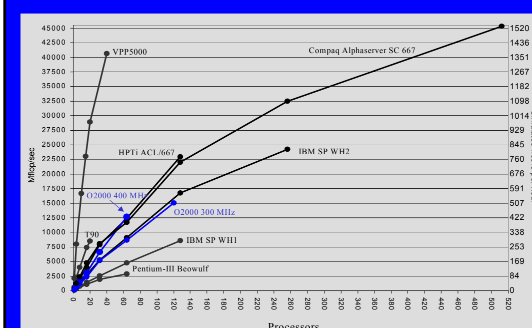
- Uniprocessor (non-parallel): workstations
- Vector shared memory: C90, T90, J90, SV1; NEC; ...
- Shared-memory multi-processors: Sun, Compaq, IBM, SGI, ….

More MM5 platforms

- Uniprocessor (non-parallel): workstations
- Vector shared memory: C90, T90, J90, SV1; NEC; ...
- Shared-memory multi-processors: Sun, Compaq, IBM, SGI, ….
- Pure Distributed Memory: IBM SP, Cray T3E, Fujitsu, Beowulf clusters
- Distributed Memory clusters of SMPs: IBM SP, Compaq, SGI, NEC, ...



Distributed Memory Parallel Performance



DM-parallel Features

- All MM5 options supported except:
  - Moving nested grids
  - Arakawa-Schubert Cumulus
  - Pleim-Xu PBL
- Bit-for-bit agreement with non-DM runs, with caveats:
  - Same hardware and libraries
  - No optimization
  - Fixed miter steps in certain boundary layer schemes
- I/O options and formats identical for model input and history; *restart* mechanism is different (see README.MPP)
- DM-parallel and non-DM parallel executables can be built in the same directory

Building the DM-parallel MM5

- Download:
  ```
  ftp://ftp.ucar.edu/mesouser/MM5V3/MM5.TAR.gz
  ftp://ftp.ucar.edu/mesouser/MM5V3/MPP.TAR.gz
  ```
- Unzip and untar:
  ```
  gzip -d -c MM5.TAR.gz | tar xf -
  cd MM5
  gzip -d -c ../MPP.TAR.gz | tar xf -
  ```
- Edit configure.user file for computer and configuration

Editing configure.user

- Find the MPP subsection in Section 7 of configure.user pertaining to your computer and uncomment those rules
- Adjust PROCMIN_NS and PROCMIN_EW settings at top of Section 7 for memory scaling
- Please see http://www.mmm.ucar.edu/mm5/mpp/cowbench/details.html

Memory scaling example

Processor Memory
50 MB

MM5
MIX=56
MJX= 68
46 MB

Memory scaling example

Processor Memory
50 MB

MM5
MIX=112
MJX= 136
184 MB

Memory scaling example

MM5
MIX=112
MJX= 136
184 MB

Memory scaling example

MM5
MIX_GLOBAL=112
MJX_GLOBAL= 136
MIX = MIX_GLOBAL / PROCMIN_NS = 56
MJX = MJX_GLOBAL / PROCMIN_EW = 68
Only 46 MB needed on each processor

Memory scaling example

Processor Memory
50 MB

MM5
MIX_GLOBAL=112
MJX_GLOBAL= 136
MIX = 56
MJX = 68
46 MB

Processor Memory
50 MB

MM5
MIX_GLOBAL=112
MJX_GLOBAL= 136
MIX = 56
MJX = 68
46 MB

Processor Memory
50 MB

MM5
MIX_GLOBAL=112
MJX_GLOBAL= 136
MIX = 56
MJX = 68
46 MB

Processor Memory
50 MB

MM5
MIX_GLOBAL=112
MJX_GLOBAL= 136
MIX = 56
MJX = 68
46 MB

PROCMIN variables

- Determine horizontal dimensions of MM5 arrays for each processor *at compile time*
- PROCMIN_NS divides MIX (north-south decomposition)
  PROCMIN_EW divides MJX (east-west decomposition)
- Can reduce per processor size of MM5 arrays to exploit the *aggregate* memory size of the parallel machine
- (PROCMIN_NS) x (PROCMIN_EW) specifies the *minimum* number of processors at compile time for which the MM5 executable is valid

PROCMIN variables (cont.)

- An executable compiled with PROCMIN_NS=1 and PROCMIN_EW=1 uses maximum per processor memory but is valid for any number of processors.
- Warning! An executable compiled with PROCMIN_NS=2 and PROCMIN_EW=2 can be run on no fewer than 4 processors, but for example it can NOT be run on 5 processors (MIX/2 dimension is too small for 1x5 decomposition)
- Violation will cause runtime abort with message in rsl.error.0000 file: 'MPASPECT: UNABLE TO GENERATE PROCESSOR MESH. STOPPING.'

PROCMIN variables (cont.)

- For the most efficient use of memory and the best performance, set PROCMIN_NS and PROCMIN_EW so that the product equals the number of processors you will be using.
- Experiment with different decompositions: e.g., runtimes for 16 processor jobs compiled as 2x8, 4x4, and 8x2 might vary significantly.

Building the code (cont.)

- Build the model: `make mpp`
- Resulting executable: `Run/mm5.mpp`
- To remake the code in different configuration:
  `make mpclean`
- To reinstall the code in different location:
  `make uninstall`

Running the model

- Generate the mmlif (namelist) file
  - `make mm5.deck`
  - Edit `mm5.deck`
  - `./mm5.deck` (creates namelist file in `Run/mmlif` ; does not run code)
- Run the model
  - `cd Run`
  - `mpirun -np 4 ./mm5.mpp` (standard, MPICH)
  - `dmpirun` (DEC MPI)
  - `mprun` (Sun MPI)
  - `mpimon` (Linux/ScaMPI)
  - `poe` (IBM)

Running the model (cont.)

- Model generates normal MMOUT_DOMAIN output files and 3 text files per processor:
  - rsl.out.0000 (contains standard output)
  - rsl.error.0000 (contains standard error)
  - show_domain_0000 (shows the domain decomposition)

Test datasets

- Storm of the Century
  - ftp://ftp.ucar.edu/mesouser/MM5V3/TESTDATA/input2mm5.tar.gz
  - ftp://ftp.ucar.edu/mesouser/MM5V3/TESTDATA/soc_benchmark_config.tar.gz
  - Good small case for initial testing
  - Includes a nest
- Large domain (World Series Rain-out)
  - ftp://ftp.ucar.edu/mesouser/MM5V3/TESTDATA/largedomainrun.tar.gz
  - Representative problem sizes for distributed memory

"Same source" concept

- Ideal – Source code for the DM-parallel and non-DM parallel model are identical *at the science level*
- Hide parallel details "under the hood"- automate and encapsulate
- Parallel toolbox:
  - FLIC - automatic generation of I and J loop indexes
  - RSL – routines for domain decomposition and message passing

"Same source" (cont.)

sound.F:

```
#ifdef MPP1
#  include <mpp_sound_30.incl>
#endif
```

MPP/RSL/mpp_sound_30.incl:

```
      CALL RSL_EXCH_STENCIL(DOMAINS(INEST),STEN_SB(INEST))
```

MPP/RSL/parallel_src/define_comms.F:

```
      COMM_3PT_NE(u3d,3)
      COMM_3PT_NE(v3d,3)
      messages(1) =          RSL_INVALID
      messages(2) =                n1
      messages(3) =                      ne
      messages(4) =          RSL_INVALID
      messages(5) =                      el
      messages(6) =          RSL_INVALID
      messages(7) =                RSL_INVALID
      messages(8) =                      RSL_INVALID
      call rsl_create_stencil(sten_sb(inest))
      call rsl_describe_stencil(did,sten_sb(inest),RSL_8PT,messages)
```


DM-parallel MM5 and Linux clusters

- Cost effective
- Scale well with good interconnect
  - Dolphin/Scali
  - Myrinet
- Reliable, but in-house expertise needed
- Distributed memory version of MM5 necessary

More

- Reporting problems with DM-parallel version
  - First rebuild the code and reproduce problem
  - Test non-DM parallel version with same configuration
  - Check for consistent MM5.TAR and MPP.TAR versions
  - Provide: good description of aberrant behavior, the version of code, plus the configure.user, mmlif, rsl.out.0000, rsl.error.0000, and any other rsl.error.* or rsl.out.* files that contain tell-tale error messages.
- Advanced topics
  - Adding/modifying code for DM-parallelism
  - Porting to new platforms
- Additional information
  - README.MPP file
  - http://www.mmm.ucar.edu/mm5/mpp
    - Downloading, compiling, running
    - Helpdesk
    - MPP Design and Implementation Document
  - mesouser@ucar.edu

Ask Rotang