Profiling of the operational and experimental codes of the National Air Quality Forecast Capability running in NCEP power 6 IBM super computer

Pius Lee[1*], Daewon Byun[1], Hsin-Mu Lin[2], Daniel Tong[2], Tianfeng Chai[2], David Wong[3], Youhua Tang[4], Jeff McQueen[5], Marina Tsudlko[4], Jeff Young[3], Ho-Chun Huang[4], Sarah Lu[4], Ivanka Stajner[6], and Paula Davidson[7]

1. INTRODUCTION

Air Quality forecast models are employed to provide numerical guidance for forecasters to issue timely ozone and particulate matter concentration forecast pertinent to human health exposures to the communities they serve. NOAA's Air Resource Laboratory (ARL) is leading the research effort to facilitate such Air Quality Forecast (AQF). The National Centers for Environmental Prediction (NCEP), National Weather Service Weather Research and Forecasting Non-hydrostatic Mesoscale Model (WRF/NMM) has been coupled with the EPA Community Multi-scale Air Quality (CMAQ) model to form the National Air Quality Forecast Capability (NAQFC). It provides such guidance for the Eastern half of the U.S. since 2005 and over the Continental U.S. (CONUS) with 12 km horizontal grid spacing since 2007. Geographic expansion and the anticipated refinement in vertical and horizontal grid spacing make aggressive optimization and speeding up of the code more imperative so as to meet the lead time requirement of AQF. There is evidence that both the operational and experimental codes of NAQFC show signs of rather rapidly diminishing speedup with respect to processor allocation due to communication bound issues. This study provides a fine granularity process and communication/computation ratio analyses of both the NAQFC 2009 codes running on NCEP's power 6 chip IBM supercomputer. Individual processes, such as: advection; diffusion; convective mixing; cloud-processes; gaseous, aqueous and aerosol chemistries; and removal processes such dry deposition and scavenging processes will be examined. Therefore a thorough profiling of the code and a process-specific break down of the cpu time consumption given in this study can guide effort to optimize these codes with sight on bettering load balancing and higher parallelization efficiency.

2. MODEL AND HARDWARE CONFIGURATIONS

CMAQ of the current NAQFC over CONUS with 12 km horizontal spacing with 265 grid points in the latitudinal and 442 grid points in the longitudinal directions, respectively; has been running in NCEP's new power 6 super-computer since March 2009. CMAQ version 4.5 with aero3 turned off was used. The power 6 machine has been used to provide a 48 hours free-forecast twice daily initialized at 06 and 12 UTC, respectively. Two nodes have been allocated with the following Processing Element (PE) arrangements: ncol=9, and nrow=7. Each of these PE's – worker PE, is assigned to one of the 63 sub-domains. One additional I/O dedicated PE is allocated, totaling 64 PE's. The general hardware specifications are as follows:

[*1] Corresponding Author Address: Pius Lee,
[1]NOAA/OAR/ARL, 1315 East West Hwy, Room 3461, Silver Spring, MD 20910.
[2] Science and Technology Corporation, Hampton, VA.
[3] EPA, Research Triangle Park, NC.
[4] Scientific Applications International Corporation, Camp Springs, MD.
[5] NOAA/NWS/National Centers for Environmental Prediction, Camp Springs, MD.
[6] Noblis, Inc., Falls Church, VA.
[7] Office of Science and Technology, National Weather Service, Silver Spring, MD.

NCEP power6 IBM supercomputer	
Total nodes	144 Compute Nodes
Processors/node	32 physical PE
Total processors	4,544
speed	85.4 Tflops
RAM/node	128 GB Memory

There is no significant wallclock time difference between the 06 and 12 UTC runs. They averaged 80 minutes. The variation of clock-time lied within a few percent attributable to difference in meteorological and chemical conditions.

3. CPU BREAKDOWN OF CMAQ

There are about seven major components in the cpu breakdown. Each of them is associated with a science process in the model. In terms of their time split sequence, they are: (1) met and emission data ingest and vertical diffusion calculations -- vdiff, (2) advection in x-direction -- xadv, (3) advection in y-direction -- yadv, (4) horizontal diffusion -- hdiff, (5) cloud process -- cldproc, (6) gas-phase chemistry -- chem, and (7) aerosol processes -- aero. The hardware performance monitor (hpm) tool is used to register the time consumed in each segment of the code pertinent to these processes. Figure 1 shows the percent time spent in these processes. It is noted that horizontal advection calculation consumed the bulk of the cpu amounting to roughly 40% of the total 70 minutes per 2 days forecast. This proportion is expected to increase when finer resolution of NAQFC is pursued, since the Computational Fluid Dynamics (CFD) criterion will dictate smaller time steps for this advection process and consequently boost up the cpu consumption of this module. Another observation is that, vdiff, the first process includes rather large amount of I/O and temporal interpolation of meteorological and emission fields. Furthermore the separate handling of communication time and computational time was not done in this coarse granularity approach. To obtain a ratio of communicational to computational burden of CMAQ, one could have used the prof tool. Hereafter results of both using prof and hpm with refined granularity were prepared and analyzed.

It is noted that the I/O dedicated PE is almost 100% utilized by communication requirement. However, the I/O dedicated PE in reality is doing the outputting "quilting", where it is seen not to be the speed limiting step, as it spent a lot of cpu in "probing" the readiness of the "submitted" output from the computational PE's. Therefore CMAQ is not bounded by the output quilting step as expressed by the ratio of time used by Iprobe/mpi-receive and actual writing out to an external file.

The utilization ratio output by the hpm tool provides a measure of parallelization efficiency for that segment of the code, or of a particular module in the science-module context. for sto c To is significant as With this coarse granularity of analysis in terms of these science process modules, one may not able to delineate times consumed in

4. OPTIMAL ALLOCATION OF PE's

Optimal load balancing in CMAQ is a challenging task. As alluded in the previous section on CFD conditions that swifter wind in one sub-domain may require finer advection time step and lengthen the cpu requirement. This and many a meteorology and chemistry regime dependent parameter govern the converge-time of the processes and thus dictate the total science processes computational time of each of the sub-domains. Therefore to ensure a similar number of total grid-cells in each of the sub-domains may guarantee temporally averaged load-balancing but not actual per synchronization time-step load balancing. This is an inherent difficulty unless dynamic load balancing is considered to resolve this challenge.

Otherwise even distribution of the grid-cells to the sub-domains each assigned to a worker PE is an optimal load balancing approach adopted in CMAQ.

On the other hand an optimal minimization of communication overhead is by assuring an aspect ratio of a sub-domain as close to unity as possible (e.g. Clement et al. 2007, Truong and Fahringer). The length of the perimeter of a square is minimal among all rectangular-shaped sub-domains. The length of the perimeter is a measure of how many grid-cells on the boundary between neighboring sub-doamins that message passing is required. In CMAQ the intra processor message passing is done in each synchronization time steps.

Several levels of resource allocation have been tested. For instance 145 PE's has been attempted with disappointingly almost no reduction to the 80 minutes wall-clock time when 64 PE's were used. Figure 2 showed clock time versus allocation levels. The diminishing return curve in the figure together with operational considerations, a resource level of 3 nodes and around 90 PE's were chosen. Two levels of PE resource have been tested: around (1) 96 PE's and (2) 50 PE's. In addition, different aspect ratios of the sub-domains for these levels of resource allocation have been attempted. For the 96± PE's, the attempted domain decompositions are (1a) ncol=10, nrow=10; (1b) ncol=12, nrow=8; (1c) ncol=14, nrow=7, (1d) ncol=16, nrow=5. Similarly for 64 ± PE's, the attempted decompositions are: (2a) ncol=8, nrow=8, (2b) ncol=9, nrow=7, (2c) ncol=11, nrow=6, and (2d) ncol=13, and nrow=5. Table 2 shows the total grid cells and aspect ratios of these configurations. Figure 2a and b show the cpu time with respective to the aforementioned science processes for these two levels of resource allocations.

It is shown that at current operational set up of (2b) at NCEP is not the optimal PE configuration. As the argument of this section's second paragraph has postulated the configuration with the minimum width to length aspect ratio of the sub-domains should result in best parallelization efficiency as message passing overhead is minimized.

5. REFERENCES

Cermele, M. Colajanni, G. Necci (1997): Dynamic load balancing of distributed SPMD computations with explicit message-passing. *Preprint*: *6th Heterogeneous Computing Workshop*, Geneva Switzerland, April 1st 1997. pp 1-2.

Clement, B. J., E. H. Durfee, and A. C. Barrett (2007): Abstract Reasoning for Planning and Coordination, Journal of artificial intelligence research, **28**, pp 453-515.

Cortes, A., A. Ripoll, M. A. Senar, and E. Luque (2004): Varying the domain size of thedynamic load-balancing algo algorithm DASUD for SPMD and MPMD programming scenarios, *International Journal of High Performance Computing and Networking*, ***Vol 1 No*** *, pp180-192 DOI:* 10.1504/IJHPCN.2004.008347.

Otte, T. L., G. E. Pouliot, J. E. Pleim, J. O. Young, K. L. Schere, D. C. Wong, P. C. S. Lee, M. Tsidulko, J. T. McQueen, P. Davidson, R. Mathur, R. H. Chuang, G. DiMego, and N. L. Seaman (2005): Linking the Eta Model with the Community Multiscale Air Quality (CMAQ) Modeling System to Build a National Air Quality Forecasting System, *Wea. Forecasting*: **20**, 367–384.

Truong, H-L and Fahringer, T. (2002): Performance Analysis for MPI Applications with SCALEA. *Proceeding. of the 9th European PVM/MPI Conf*., Linz, Austria (September 2002).

Wilks, D. S. (1995): *Statistical Methods in the Atmosphic Sciences: An Introduction.* Academic Press, 238 – 241.