Computational Aspects of the Air Quality Forecasting Version of CMAQ (CMAQ-F)

David C. Wong
Lockheed Martin Information Technology
Research Triangle Park, NC
wong.david-c@epa.gov

Jeffrey O. Young
NOAA (on assignment to the EPA)
Research Triangle Park, NC
young.jeff@epa.gov

1. INTRODUCTION

The air quality forecast version of the Community Modeling Air Quality (CMAQ) model (CMAQ-F) was developed from the public release version of CMAQ (available from http://www.cmascenter.org), and is running operationally at the National Weather Service's National Centers for Environmental Prediction (NCEP). Most of the modifications to CMAQ focus on tailoring the model's performance to the operational hardware and the parallel computing environment at NCEP and effectively managing the parallel input and output (I/O) processes to achieve scalability. This has been accomplished by asynchronously overlapping computation with I/O writing to disk (WTD) by a dedicated processor.

In this paper, we describe two different ways of placing the WTD processor. Benchmark results with regards to the WTD processor placement and an extension to more than one WTD processor are presented. In addition, a benchmark comparison between the I/O implementation in the CMAQ-F and the community version of CMAQ is provided.

2. I/O STRATEGY

In CMAQ, all processors perform computation. At the end of each time step, output data is sent to one of the computational processors as a WTD processor, which then gathers and assembles data and writes it. We will label this strategy, "stnd" (standard).

CMAQ-F uses a dedicated WTD processor to perform the output task, which gathers the data from the computational processors and writes the data to disk. We label this strategy, "dwtd" (dedicated write to disk).

In either approach, there are two distinct implementations: assign either the first or the last processor from the allocated processor group to be the WTD processor. The former choice is used currently in both CMAQ and CMAQ-F.

Figure 1 depicts a scenario of allocating 9 processors on an IBM eServer p655+ system, which has 8 CPU's on a node with two on a chip. There are 8 processors that perform computation and one, indicated in the darker color, is dedicated for WTD. The logical view of the 8 computational processors is shown in the middle of Figure 1. We considered two physical mappings for the placement of the WTD processor, as shown in Figure 1.

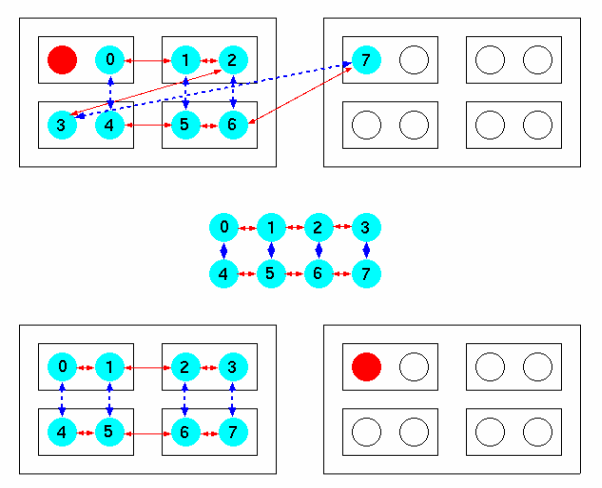

Figure 1: Processor allocations and interprocessor communication (top: WTD processor at beginning, bottom: WTD processor at end)

Figure 1 also shows interprocessor communication, where dotted arrows indicate communication in the *y* direction and red arrows in the *x* direction. It is clear that with these two WTD placement choices, there are different off-chip and off-node communication pathways in the *x* and *y* directions in CMAQ-F. For CMAQ, the first processor usually has a higher work load than the last processor because of the domain decomposition is uneven. Therefore it would seem to be advantageous to assign the last processor for WTD.

3. DESCRIPTION OF EXPERIMENTAL

RUNS

A set of experiments was conducted that consists of six groups with various numbers of processors and configurations. Each group consists of four cases: 1) the standard (stnd) strategy; 2) using the standard strategy but with the same number of computational processors as the dedicated WTD processor case (stndw); 3) a single dedicated WTD processor case (1-dwtd), and 4) with two dedicated processors (2-dwtd). Both WTD processor placement schemes were applied to the entire set of experiments. Table 1 summarizes the processor configuration for each case. All runs were conducted on EPA's IBM eServer during regular operating conditions. The domain that was tested covers the eastern half of the US. A 3-hour simulation without aerosols, and a longer, 12-hour simulation with aerosols were conducted. Results are presented as an average of three separate runs. (The 3-hour data for stndw are not presented.)

Table 1: Processor configuration of each run

Group	*stnd*	*stndw*	*1-dwtd*	*2-dwtd*
1	3x3	4x2	4x2+1	4x2+2
2	4x4	5x3	5x3+1	5x3+2
3	5x5	8x3	8x3+1	8x3+2
4	11x3	8x4	8x4+1	8x4+2
5	7x7	8x6	8x6+1	8x6+2
6	13x5	8x8	8x8+1	8x8+2

Performance is affected by domain decomposition as can be seen in Table 2. However, in this study, we did not attempt to account for performance differences due to processor configuration.

Table 2: Performance of a 3-hr simulation with various configuration of 16 and 64 processors

PE conf.	*1x16*	*2x8*	*4x4*	*8x2*	*16x1*
Time (sec)	409.7	372.0	378.0	391.7	372.3

PE conf.	*1x64*	*2x32*	*4x16*	*8x8*	*16x4*	*32x2*	*64x1*
Time (sec)	333.3	236.0	241.3	241.0	227.3	217.0	224.3

4. EXPERIMENTAL RESULTS

Figure 2 shows the results of the 3-hour simulations (without aerosols), with the WTD processor placed at the beginning and at the end of the group of allocated processors.

Figure 2 shows the standard method performs better for a small number of processors. With more processors, there is not much difference between the standard and the dedicated WTD methods. In addition, using more than one dedicated WTD processor does not gain any additional performance. Also placing the WTD processor at the end results in slightly better performance for the standard method but not for the dedicated WTD processor cases.

Figure 3 shows the results of the 12-hour simulation with aerosols. Comparing "stndw" to the "1-dwtd" since both have the same computational processor configuration, we see the "1-dwtd" case performs better than the "stndw" method as anticipated. However, the "stnd" method is better than the "1-dwtd" as we saw in the 3-hour run cases. Also, as in the 3-hour cases, it can be seen that using more than one dedicated WTD processor does not render any additional performance gain and that placing the WTD processor at the end results in slightly better performance for the standard method but not for the dedicated WTD processor cases.

5. CONCLUSIONS AND FUTURE RESEARCH

The dedicated WTD scheme with an m X n computational processor configuration performs slightly better than the standard method with an m X n configuration. However, if that dedicated WTD processor is part of the computational processors (no longer dedicated) in the standard method with an m' X n' processor configuration, where $m' * n' = m * n + 1$, then a slightly better performance will result. If the computational portion of CMAQ becomes more intensive due to an increased problem size, e.g. larger domains, higher resolution, we would expect even better performance.

We observed a significant load balance issue across the processors in these runs. Figures 4 and 5 provide snapshots of minimum and maximum processor execution times of the science processes from the 12-hour runs. As expected, the absolute difference between the min and max gets smaller as the number of allocated processors increases. However, the load imbalance remains, and we believe that it will get worse with larger problem sizes. We will begin to focus more on this issue for future work.

The IBM eServer p655+ on which the experiment was run has 8 CPU's per node (8-way).

A regular Linux cluster is usually 2-way or 4-way. For such a platform, the latency of off module communication will be magnified, and the network bandwidth will be different than on the IBM. Since there is considerable interest in running CMAQ on Linux clusters, we will continue this study on such platforms.

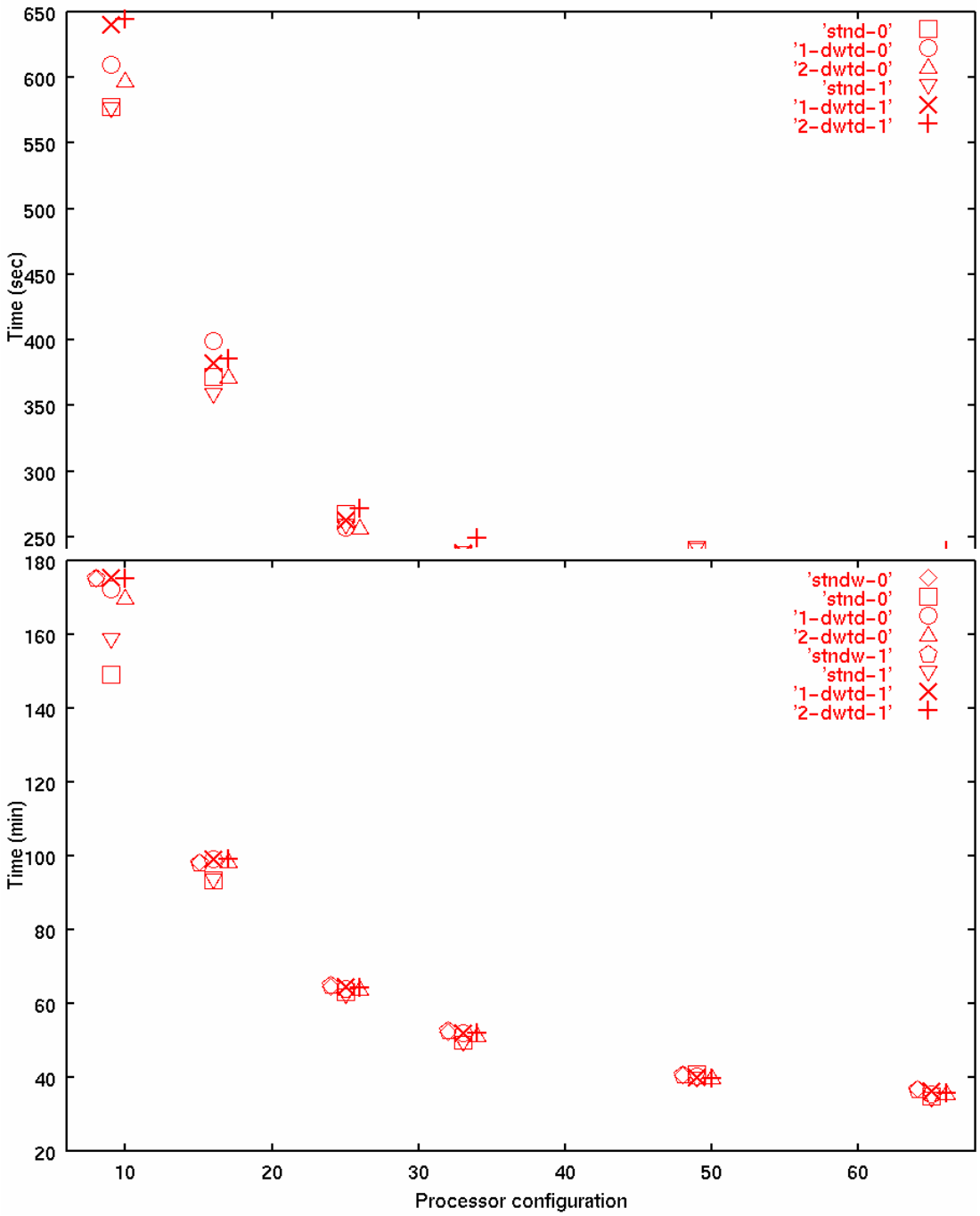

Figure 3: 12-hour simulation with aerosols with WTD processor(s) at the beginning (-0) and end (-1) of the processor group

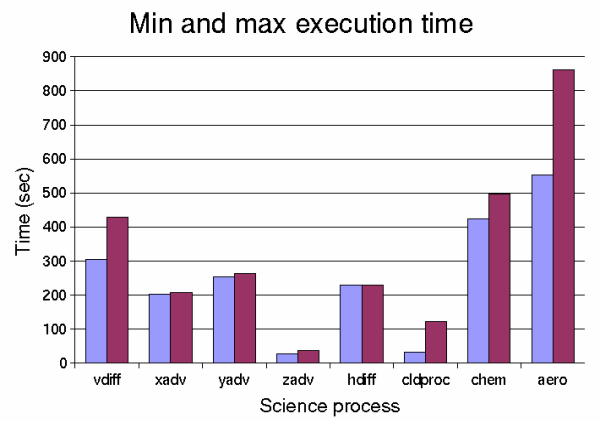

Figure 4:Min and max execution time with 11x3 processor configuration

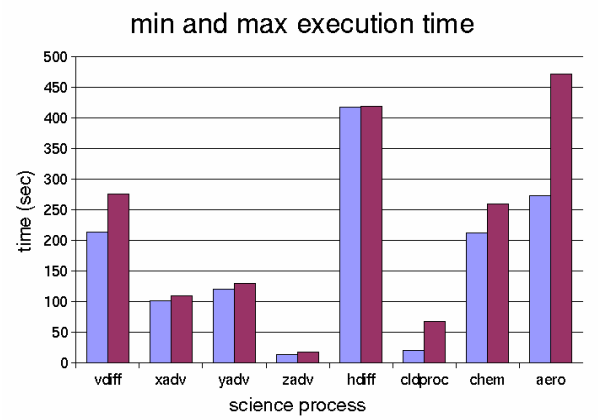

Figure 5: Min and max execution time with 13x5 processor configuration

DISCLAIMER

The research presented here was performed under the Memorandum of Understanding between the U.S. Environmental Protection Agency (EPA) and the U.S. Department of Commerce's National Oceanic and Atmospheric Administration (NOAA) and under agreement number DW13921548. Although it has been reviewed by EPA and NOAA and approved for publication, it does not necessarily reflect their policies or views.