Atmospheric Pollution Research

www.atmospolres.com

Development and application of a parallelized version of the advanced modeling system for transport, emissions, reactions and deposition of atmospheric matter (AMSTERDAM): 1. Model performance evaluation and impacts of plume–in–grid treatment

Prakash Karamchandani [1,2], Krish Vijayaraghavan [1,2], Shu–Yun Chen [1,2], Rochelle Balmori–Bronson [1], Eladio M . Knipping [3]

[1] *AER, Inc., 388 Market Street, Suite 750, San Francisco, CA 94111, USA*
[2] *Currently with ENVIRON International, 773 San Marin Drive, Novato, CA 94945, USA*
[3] *EPRI, 2000 L Street NW Suite 805, Washington, DC 20036, USA*

ABSTRACT

The Community Multiscale Air Quality Model (CMAQ) is a comprehensive three–dimensional "one–atmosphere" air quality model that is now routinely used to address urban, regional–scale and continental–scale multi–pollutant issues such as ozone, particulate matter, and air toxics. Several updates have been made to CMAQ by the scientific community to enhance its capabilities and to provide alternative science treatments of some of the relevant governing processes. The Advanced Modeling System for Transport, Emissions, Reactions and Deposition of Atmospheric Matter (AMSTERDAM) is one such adaptation of CMAQ that adds an Advanced Plume–in–grid Treatment (APT) for resolving sub–grid scale processes associated with emissions from elevated point sources. It also incorporates a state–of–the–science alternative treatment for aerosol processes based on the Model of Aerosol Dynamics, Reaction, Ionization and Dissolution (MADRID). AMSTERDAM is configured to provide flexibility to the model user in selecting options for the new science modules. This paper describes the parallelization of AMSTERDAM to make it a practical tool for plume–in–grid (PinG) treatment of a large number of point sources, and presents results from its application to the central and eastern United States for summer and winter periods in 2002. Over 150 coal–fired power plants in the domain with high emissions of sulfur dioxide (SO_2) and nitrogen oxides (NO_x) were selected for PinG treatment in the CMAQ–MADRID–APT configuration of AMSTERDAM used for this application. Although both model configurations (grid–only and PinG) give similar model performance results (an aggregate measure of model skill), the results show significant differences between the two versions in the specific nature of the predicted spatial distribution of ozone and $PM_{2.5}$ concentrations. These differences can be important in determining source contributions to ambient concentrations. A companion paper examines the differences in the predicted contributions of hypothetical source regions from the two configurations of the model.

Keywords:
Plume–in–grid
CMAQ
MADRID
APT
Subgrid–scale
Plume chemistry

Article History:
Received: 15 April 2010
Revised: 31 August 2010
Accepted: 02 September 2010

Corresponding Author:
Prakash Karamchandani
Tel: +1-415-899-0754
Fax: +1-415-899-0707
E–mail: prakash@environcorp.com

doi: 10.5094/APR.2010.033

1. Introduction

The U.S. Environmental Protection Agency (EPA) Community Multiscale Air Quality (CMAQ) model (Byun and Schere, 2006), is a one–atmosphere three–dimensional grid model that is being used to predict the impacts of emission controls on the atmospheric concentrations and depositions of multiple pollutants such as ozone (O_3), fine particulate matter ($PM_{2.5}$) and air toxics. Because it is a community model, several enhancements to the model have been made by the air quality modeling community to provide alternative science treatments of some of the governing processes or to include treatments that are not supported in the base CMAQ.

The Advanced Modeling System for Transport, Emissions, Reactions and Deposition of Atmospheric Matter (AMSTERDAM) is a version of CMAQ that incorporates an alternative treatment of aerosol processes and also adds a plume–in–grid treatment to simulate the subgrid–scale features associated with pollutant emissions from point sources. Grid models, such as CMAQ, necessarily average emissions within the volume of the grid cell where they are released. This averaging process may be appro–priate for sources that are more or less uniformly distributed at the spatial resolution of the grid system. However, it may lead to significant errors for sources that have a spatial dimension much smaller than that of the grid system. For example, stack emissions lead to plumes that initially have a dimension of tens of meters, whereas the horizontal resolution in grid–based air quality models is typically several kilometers in urban applications and up to approximately 40 km in continental applications. This artificial dilution of stack emissions leads to (1) lower concentrations of plume material, (2) unrealistic concentrations upwind of the stack, (3) incorrect chemical reaction rates due to the misrepresentation of the plume chemical concentrations and turbulent diffusion, and (4) incorrect representation of the transport of the emitted chemicals.

Plume–in–Grid (PinG) modeling has been demonstrated to be an effective approach to resolve sub–grid scale effects associated with discrete sources (e.g., Seigneur et al., 1983; Sillman et al., 1990; Kumar and Russell, 1996; Gillani and Godowitch, 1999; Karamchandani et al., 2002; Godowitch, 2004; Karamchandani et al., 2006a; Vijayaraghavan et al., 2008; Karamchandani et al.,

2009). In this approach, the errors associated with the grid–averaging of stack emissions are addressed by using a subgrid–scale representation of stack plumes that is imbedded in the 3D grid system of the air quality model.

While PinG modeling provides a more accurate and detailed representation of point source emissions than a traditional grid model, it increases the computational time required for model simulations, particularly when a large number of point sources are treated explicitly with the embedded plume model. This additional computational overhead can make it impractical to use PinG modeling for large modeling domains and long simulation periods. In this paper, we describe the development of a parallelized version of AMSTERDAM to overcome these limitations, and present model performance results with and without PinG treatment. We also compare the spatial patterns of predicted ozone and $PM_{2.5}$ concentrations from the two model configura–tions to illustrate the differences between the two approaches. In a companion paper (Karamchandani et al., 2010), we present results from hypothetical emission control scenarios to illustrate the effect of PinG modeling on predicted impacts of emissions reductions on ozone and $PM_{2.5}$ concentrations and sulfur and nitrogen deposi–tion.

2. AMSTERDAM

AMSTERDAM is actually a suite of models, based on CMAQ, with user–selectable configurations for chemistry, aerosols and plume–in–grid (PinG) treatment. In addition to the standard CMAQ configurations, the new configurations offered by AMSTERDAM include CMAQ–AERO3–APT, CMAQ–MADRID, and CMAQ–MADRID–APT. MADRID, which refers to the Model of Aerosol Dynamics, Reaction, Ionization and Dissolution, is an advanced alternative aerosol treatment developed by Zhang et al. (2004). MADRID is available with both the Carbon Bond IV and SAPRC–99 gas–phase chemistry options. PinG treatment is provided with the Advanced Plume Treatment (APT) option (Karamchandani et al., 2002; Karamchandani et al., 2006a). This option is available with both the AERO3 aerosol module of CMAQ and the MADRID aerosol treatment. AMSTERDAM also includes options for the treatment of mercury (Hg) species based on Seigneur et al. (2004; 2006).

The embedded reactive plume model for the APT option is adapted from the Second–Order Closure Integrated puff model with Chemistry (SCICHEM) (Karamchandani et al., 2000). SCICHEM simulates plume transport and dispersion using a second–order closure approach to solve the turbulent diffusion equations. The plume is represented by a myriad of three–dimensional puffs that are advected and dispersed according to the local micrometeo–rological characteristics. Each puff has a Gaussian representation of the concentrations of emitted inert species. The overall plume, however, can have any spatial distribution of these concentrations, since it consists of a multitude of puffs that are independently affected by the transport and dispersion characteristics of the atmosphere. The model can simulate the effect of wind shear since individual puffs will evolve according to their respective locations in an inhomogeneous velocity field. As puffs grow larger, they may encompass a volume that cannot be considered homogenous in terms of the meteorological variables. A puff splitting algorithm accounts for such conditions by dividing puffs that have become too large into a number of smaller puffs. Conversely, puffs may overlap significantly, thereby leading to an excessive computa–tional burden. A puff–merging algorithm allows individual puffs that are affected by the same (or very similar) micro–scale meteorology to combine into a single puff. Also, the effects of buoyancy on plume rise and initial dispersion are simulated by solving the conservation equations for mass, heat, and momentum. The formulation of nonlinear chemical kinetics within the puff framework is described by Karamchandani et al. (2000). Chemical species concentrations in the puffs are treated as perturbations from the background concentrations. The chemical reactions within the puffs are simulated using a general framework that allows any chemical kinetic mechanism to be treated. The puff chemical mechanism is the same as the host grid model mechanism for consistency.

The APT option for PinG treatment was initially developed and applied for ozone (Karamchandani et al., 2002; Vijayaraghavan et al., 2006) and subsequently extended to particulate matter (Karamchandani et al., 2006a) and mercury (Karamchandani et al., 2006b; Karamchandani et al., 2006c, Vijayaraghavan et al., 2008).

Because of the computational overhead associated with the PinG treatment (about 20 to 30% for 50 sources), early model applications were limited to small domains and/or short–term simulations, with no more than 50 point sources treated explicitly with the embedded plume model. However, these constraints limited the utility of the model and it became apparent that it would be necessary to reduce the turn–around time for PinG applications. In the next section, we describe our approach to achieve this speed–up, based on parallelization of the PinG code. This approach relies on the widespread availability of multi–processor workstations and workstation clusters that are commonly used today for air quality model simulations. A parallel PinG code allows efficient utilization of the available compute cycles in these modern computer systems.

3. Parallelization of AMSTERDAM

The traditional approach to parallelizing a grid model such as CMAQ is to perform domain decomposition by subdividing the horizontal domain into a number of roughly equal subdomains, with each subdomain assigned to a separate processor. Each processor then performs the transport/chemistry/removal calcula–tions on the grid cells within the subdomain. However, inter–processor communication is required for I/O purposes and horizontal transport calculations. In CMAQ, this inter–processor communication is accomplished by using the parallel input output (PARIO) management library based on the Message Passing Interface (MPI) library (http://www.mcs.anl.gov/research/projects/mpi/), a standard for message passing in parallel comput–ing. CMAQ uses the Argonne National Laboratory open–source implementation of MPI, referred to as MPICH (http://www.mcs.anl.gov/research/projects/mpich2/), because of its widespread usage and availability.

While the domain decomposition paradigm is appropriate for the grid model, the plume component (SCICHEM) in the PinG model requires a different approach because the puffs are not distributed uniformly among the subdomains. For example, one could expect a higher density of puffs in subdomains with many point sources than in other subdomains. Thus, using a domain decomposition approach for SCICHEM would result in inefficient utilization of processors. Furthermore, there could be potential issues with puffs crossing subdomain boundaries during a simulation time step.

Hence, we selected “puff decomposition” as the approach for parallelizing the plume component of the model. The total puffs at any given time step are divided uniformly among the available processors. However, the strongly interactive nature of the puff calculation, including splitting, merging, and overlap calculations posed an additional challenge in the parallelization of the plume–in–grid code. Because these puff interactions could occur between puffs distributed among different processors, there would be a significant communication overhead associated with performing the interaction calculations on independent processors.

To overcome this issue, we focused our parallelization effort on the chemistry component of the plume model. This component that includes gas–phase chemistry, aerosol calculations, and aqueous–phase chemistry, requires more computing resources