NASA-CR-205004

INTERIM

IN-45-CR

044695

NASA PROGRESS REPORT

Title of Grant: Advanced Modeling Techniques to Study Anthropogenic Influences on Atmospheric Chemical Budgets

Grant Number: NAGW-4701

Type of Report: Year 2 Progress/Status Report

Principal Investigator: Rohit Mathur

Reporting Period: August 1, 1996 - June 1, 1997

Institution:
MCNC
3021 Cornwallis Road
Research Triangle Park, NC 27709

1. Introduction

This research work is a collaborative effort between research groups at MCNC and the University of North Carolina at Chapel Hill. The overall objective of this research is to improve the level of understanding of the processes that determine the budgets of chemically and radiatively active compounds in the atmosphere through development and application of advanced methods for calculating the chemical change in atmospheric models. The research performed during the second year of this project focused on four major aspects: (1) The continued development and refinement of multiscale modeling techniques to address the issue of the disparate scales of the physico-chemical processes that govern the fate of atmospheric pollutants; (2) Development and application of analysis methods utilizing process and mass balance techniques to increase the interpretive powers of atmospheric models and to aid in complementary analysis of model predictions and observations; (3) Development of meteorological and emission inputs for initial application of the chemistry/transport model over the north Atlantic region; and, (4) The continued development and implementation of a totally new adaptive chemistry representation that changes the details of what is represented as the underlying conditions change.

2. Work Progress and Status

2.1 Development of Multiscale Modeling Methodologies to Address Scale Interactions of Atmospheric Systems

The focus of this activity is to study grid resolution issues governing the transport, transformation, and subsequent fate of pollutants emitted from concentrated source areas as they are transported to remote environments, using grid nesting, grid refinement, or a combination of the two techniques. During the second year of the project we have continued with the testing of both one-way and two-way nesting approaches. We have implemented both one-way and two-way grid nesting techniques in the Multiscale Air Quality Simulation Platform (MAQSIP), an eulerian chemistry-transport modeling system developed at MCNC. We have evaluated a variety of spatial interpolation techniques including geometric interpolation techniques and advection equivalent interpolation methods (e.g. Smolarkiewicz and Grell, 1992). The geometric interpolation techniques evaluated include a zeroth-order and a quadratic scheme (Kurihara et al., 1979; Clark and Farley, 1984) while the two advection interpolation schemes were based on the upwind and Bott's (Bott, 1989) advection schemes. The performance of each scheme was evaluated by examining how well it maintains the amplitude, phase, and mass of an initially cone-shaped distribution that is advected from a coarse to a nested fine grid. Our analyses of such test problems suggest the viability of advection equivalent interpolation schemes for specification of boundary conditions in nested grid models. In particular, our results indicate that the formal equivalence of finite-difference operators for interpolation and advection can be exploited to yield more robust and accurate interpolation schemes that retain the in-built positive-definiteness and shape preserving characteristics of advanced advection schemes and consequently maintain better compatibility between solutions from meshes of different resolution as compared to traditional geometric interpolation schemes. During this phase of the project, we also prepared and submitted a journal article describing this work; the paper has been accepted for publication in the *Monthly Weather Review*.

We also continued investigation on development of a fully-interactive two-way nested grid approach. In the initial version of this approach we are investigating the feasibility of feedback of the fine-grid predictions to the coarse-grid model through updating each CGM cell after the FGM sub-steps by the average of concentrations of all FGM cell overlapping each CGM cell. An important issue being investigated relates to the impact of emission averaging of primary species over grids of different spatial resolution. When subject to nonlinear chemistry, these could result in different distributions of derived chemical species, even though the total mass is conserved in the two grids. Approaches to reconcile the chemical distribution in the two grids are being investigated.

In a parallel effort, we have also continued with the extension, evaluation and testing of a variable grid model using grid refinement techniques suggested by Mathur et al. (1992). In the first year of the project we had incorporated the more detailed tropospheric gas-phase chemical mechanism of Stockwell et al. (1990). This finite-element based model uses structured grids to discretize the horizontal domain based on a-priori emission source distribution using a mesh generation technique wherein local refinement is achieved by redistribution of grid points. A test case problem has been set-up to study the ability of the methodology in simulating the downwind O_3 formation in plumes from major source clusters and to investigate the impacts of artificial dilution associated with emission averaging over grid-cell dimensions on chemistry/transport calculations. We have conducted a number of simulations for these test conditions using four different grid structures: a uniform 20 km resolution grid; a variable grid using the same number of computational nodes as the 20 km grid; a uniform 10 km resolution grid ; and a uniform 5 km resolution grid (representing the finest resolution of the variable grid). Model predictions for both primary and secondary species computed on the different grids were compared and analyzed to study the impact of grid size on resolution of emission and subsequent chemical transformations. Comparison of model predictions for different grids were also performed to demonstrate the viability of the use of grid refinement techniques and their advantages over traditional uniform grid models. As an example, Figure A1 presents distribution of average O_3 predicted by the different grids. Comparison of O_3 predictions for the variable and 20 km uniform grid, both utilizing the same number of computational nodes, clearly demonstrates the ability of the variable grid model to predict a wider range of concentrations with distinctly higher peak values as the plumes travels downwind of the major source regions. A 5 km resolution uniform grid solution is also shown as a reference since it represents the smallest resolution (at locations of source clusters) in the variable grid. The relative over predictions at low concentrations can be attributed to effects of numerical diffusion. It can also be observed that the variable grid predicts an overall better solution than the uniform 10 km resolution grid which uses 4 times more the number of computational nodes.

One of the shortcomings of the current version of the variable grid model based on the grid refinement approach, relates to computational considerations. The current version employs an implicit two-dimensional transport algorithm in representing horizontal transport. This increase in dimensionality and associated band width of global matrices results in increase in computational demand compared to traditional locally one-dimensional time-splitting approaches. Since meteorological parameters were constant for the test case considered, this has not posed a serious problem; the global matrix for this case does not change with time and thus can be decomposed only once during the solution procedure. We have investigated the use of alternate advection schemes to overcome this potential shortcoming. In particular, we have devoted some effort to replace the current advection scheme with the piecewise parabolic method (Collela and Woodward, 1984). Testing and evaluation of this new scheme is currently underway.

2.2 Model and Data Analysis Methodologies

We completed adapting the process and mass balance analysis techniques developed by Jeffries and Tonnesen (1994) and Jang et al. (1995) in MAQSIP. This provides an additional tool for analysis of model predictions by providing more detailed process contributions which can then be used for budget analysis to provide insight to magnitudes of the various processes (e.g. from transport, chemical transformations, deposition, and emissions) that lead to model predictions. Once sufficient confidence in model predictions is gained, such detailed process analysis can further be used to aid interpretation of observational data. To help us understand the wealth of data generated by the MAQSIP process analysis routines, we explored the application of statistical exploratory data analysis techniques. The techniques we applied included scatterplot matrices, 3D rotating point clouds, conditioning (trellis) plots, and small multiples display using star plots. Preliminary application of these techniques have allowed us to discover patterns and relationships among multiple variables more easily than approaches commonly used in air quality modeling. For instance, the small multiples displays allows similarities and differences in the relative contributions of the model's processes (e.g., chemical production and horizontal advection) at many locations to be seen easily. With the assistance of our team members, a graduate student extended the application of small multiples display we developed and incorporated it into Vis5D, a freely available scientific visualization package.

We have also acquired from NASA the TOMS version 7 gridded ozone data for 1978-1993 and selected SBUV field for 1978-1990. The use of this data for model evaluation and analysis purposes will be explored.

2.3 Meteorological and emission input development and preliminary model applications

In preparation for comprehensive modeling applications over interregional scales, we have devoted considerable effort in extending the spatial domain of applicability of MAQSIP for such applications and also towards development of appropriate meteorological and emission inputs for such applications. Figure A2 presents the extent of proposed modeling domain. We have investigated the use of two models as possible meteorological drivers for the CTM. The MM5 (Grell et al., 1994), which utilizes data assimilation, has been widely used to provide meteorological inputs for regional episodic and seasonal tropospheric chemical calculations. The RegCM2 (Giorgi et al., 1993) on the other hand has been used for long term regional climate simulations. We have applied both the MM5 and the RegCM2 models over domain shown in Figure A2. We have also initiated the adaptation of the current MM5/MAQSIP interface to provide a similar interface for the RegCM2 output. The ability to use meteorological data from either of these models would provide a useful platform to evaluate impacts of meteorological parameters on modeling of distribution of various tropospheric chemical species.

Accurate and detailed representation of both anthropogenic and biogenic emissions of sulfur and nitrogen containing compounds and hydrocarbon precursors are essential inputs for any model used for studying tropospheric oxidant and aerosol distribution and formation. For this project we are using a global emission inventory compiled by Dr. Rick Saylor at Georgia Institute of Technology. This emission data set contains SO_2 from biomass burning (Spiro et al., 1992), NO_x from biomass burning (Dignon, 1992), DMS (Spiro et al., 1992), anthropogenic hydrocarbons (Piccot et al., 1992), and CO_2 from biomass burning. In addition, from the Global Emission Inventory Activity (GEIA), we have also used emission estimates for anthropogenic NO_x and SO_2 and biogenic hydrocarbons. We have combined these various data sets to create a gridded and speciated emission data for use as input to the chemistry transport model.

We have also initiated preliminary chemistry/transport calculations over the domain shown in Figure A2. We have performed both tracer transport calculations for Radon to analyze transport characteristics over the domain as well as more detailed tropospheric chemistry calculations by including a detailed chemical mechanism (a modified version of the Gery et al. (1989) mechanism). These initial simulations have been made for a period of 10 days and use the meteorological and emission data described above. Figure A2 also presents a sample distribution of predicted O_3 from these simulations. Detailed analysis of these simulations is in progress.

2.4 Formulation and Testing of the Allomorphic Chemical Mechanism

The current methods for representing volatile organic compounds and NO_x atmospheric chemistry reacting systems is quite limited. To address the arising need to more fully represent the atmospheric chemical content, this research component conducted by Dr. Jeffries at University of North Carolina at Chapel Hill, has focused on creation of a new representation for organic species in photochemical reaction models. In this approach the reacting entities are divided into molecules, which have static properties for the whole simulation, and "morphecules", which are reacting entities like molecules but have dynamic properties that change throughout the simulation. For each morphecule there are two or more "allomorphs", which are variants on a molecule's shape (i.e., its properties). The minimum representation of a molecule is its name and its minimum state information is its concentration. The minimum representation of a morphecule is its name and the names of the associated allomorphs and its minimum state information are a total concentration (equal to sum of allomorph concentrations) and the mole fraction of its associated allomorphs. This new approach permits much more detail to be retained while still being computationally efficient.

During the second project year, we finalized and completed the mathematical formulation of this approach. This consist of three coupled problems: (1) the molecule/mophecule problem, in which concentrations of molecules and mophecules are solved for; (2) the allomorph problem, in which distribution of mole fractions of a morphecule's allomorphs are determined; and, (3) the reconciliation problem, in which the first two solutions are merged to produce an internally consistent set of molecule and morphecule concentrations, and a set of allomorph mole fractions that are different from those at the beginning of the time step. Further details on the mathematical formulation are presented in Appendix B.

We have also completed the development of a "mechanism compiler" to effectively treat the additional complexity introduced by this new chemical representation. A formal, context-free grammar for a new computer language (called MORPHO) has been formulated and implemented in C++. The use of MORPHO results in a very compact representation of atmospheric chemistry. All the tedious effort associated with hand coding of chemical reaction mechanisms into models is virtually eliminated by this software. The mechanism compiler can also produce C declarations for storage allocations and C-code output that implements the various functions and matrix updates of the mathematical problem. Further, by formulating the core problem in a sparse matrix form, clarity of operations is achieved. Sparse matrix storage and operation methods are key in the implementation of the allomorphic mechanism (refer to Appendix B for details). A variety of methods that would permit the construction and updating of block structured sparse matrices arising in the allomorphic formulation have also been investigated. While most of these are suited for iterative methods, only a few are optimized for direct operations and none are block oriented.

We have also devoted some effort towards understanding how the morphecule chemistry mechanism will be incorporated into MAQSIP. This involved understanding how the MORPHO compiler could be modified to generate stand-alone C subroutines that represent the mechanism presented to the MORPHO compiler. The C subroutines will be called from within MAQSIP each time chemical changes within a cell are computed.

3. Planned Activities for Year 3

Our research during the first two years of the project has focused primarily on the development and continuous refinement of a series of prototypes each tailored to allow exploration of specific issues with some initial implementation in a comprehensive tropospheric chemistry/transport model (CTM). During the third and final year of the project we plan to refine these prototypes and implement the successfully demonstrated concepts related to multiscale modeling and chemistry representation in the CTM. Some of the refinements such as development of methodologies to reconcile mass fractions of various species will be applicable to both the implementation of two-way nesting as well as interfacing the new chemical representation with transport processes. Additional refinements to the MORPHO compiler are also required. These include development of a compiler backend that outputs C/C++ storage declarations and functions to evaluate the various mathematical operations (refer to Appendix B for details). Subsequent to this we will begin systematic testing of the mechanism evaluator code with the mechanism solver code in MAQSIP. We will begin by replicating an existing molecule only reaction mechanism such as CBM-IV to assure that we can obtain same results as the currently hard-wired version and then proceed to the more detailed molecule/morphecule mechanisms with special focus on adaptations for both single grid models as well a nested grid models, as described in Appendix B. This integration task will involve the largest collaborative effort of the group, insuring consistent operation of the chemistry solver with rest of the model.

In parallel to these activities, we will perform a detailed analysis of the initial interregional scale simulations completed in year 2. Based on conclusions of these model analyses and evaluation activities, aditional refinements the system may be required. For example, as newer emission estimates become available from activities such as GEIA, it would be desirable to include them in the system. In addition, we also plan to perform a series of episodic model application to continuously test and evaluate new implementations in the model. We will also include treatment of aerosol processes in the model; this would be based on the modal aerosol model of Binkowski and Shankar (1995). Following the implementation of the alomorphic mechanism in the CTM, we will identify and simulate a test case problem with the new model. Model predictions will be analyzed in conjunction with available observations to demonstrate the use of the new modeling techniques and to evaluate their applicability for future studies.

4. Publications and Conference Presentations Resulting from Project Activities

Alapaty, K., R. Mathur, and T. Odman, Intercomparison of spatial interpolation schemes for use in nested grid models, *Monthly Weather Review*, in press, 1997.

Mathur, R. , J. Young, K. Schere, and G. Gipson, A Comparison of numerical techniques for solution of atmospheric kinetic equations, *Atmospheric Environment,* submitted, 1997.

Mathur, R. and L.K. Peters, On the Use of Irregular Grids in Multiscale Atmospheric Transport/Chemistry Modeling, *The 5th International Atmospheric Sciences and Application to Air Quality Conference*, June 18-20, Seattle, Washington, 1996.

Fine, S. S., R. Mathur, 1996: Applying multivariate analysis techniques to interpreting results produced by chemistry-transport models. *Proc. Computing in Environmental Resource Management,* 2-4 December, Research Triangle Park, NC, Air & Waste Management Association, 250-260.

J. E. Rickman, V. P. Aneja, S. Fine, C. Jang, 1997: Three-dimensional visualization of ozone process data. To appear in *Proc., Measurement of Toxic and Related Air Pollutants*, April 29 - May 1, Research Triangle Park, NC, Air & Waste Management Association.

5. References

Binkowski, F.S. and U. Shankar, The Regional Particulate Matter Model 1. Model Description and Preliminary Results, *J. Geophys. Res.*, 100, 26191-26209, 1995.

Bott, A., A Positive Definite Advection Scheme Obtained by Nonlinear Renormalization of the Advective Fluxes. *Mon. Wea. Rev.* **117**, 1006-1015, 1989.

Clark, T.L., and R.D. Farley, Severe Downslope Windstorm Calculations in Two and Three Dimensions Using Anelastic Interface Grid Nesting: A Possible Mechanism for Gustiness. *J. Atmos. Sci.*, **41**, 329-350, 1984.

Collela, P. and P.L. Woodward, The Piecewise Parabolic Method (PPM) for Gas-Dynamical Simulations, *J. Comput. Phys.*, 54, 174-201, 1984.

Dignon, J., Trace Gas Emission Data Bases for Atmospheric Chemistry Studies, UCRL-ID-108511, Lawrence Livermore National Laboratory, 1991.

Gery, M. W., Whitten, G. Z. , Killus, J. P. , and Dodge, M. C., A Photochemical Kinetics Mechanism for Urban and Regional Scale Computer Modeling , J. Geophys. Res., 94(D10):12,925–12,956, 1989.

Giorgi, F., M. R. Marinucci, and G. T. Bates, Development of a Second Generation Regional Climate Model (RegCM2). Part I: Boundary-layer and radiative transfer processes. Mon. Wea. Rev., 121, 2794-2813, 1993.

Grell, G.A., J. Dudhia, D.R. Stauffer, A Description of the Fifth-Generation Penn State/NCAR Mesoscale Model (MM5), NCAR Technical Note, *NCAR/TN-398+STR*, Boulder, Colorado, 1994.

Jang, J.C., H.E. Jeffries, and S.Tonnesen, Sensitivity of Ozone to Model Grid Resolution: Part II. Detailed Process Analysis for Ozone Chemistry, *Atmos. Environ.*, 29, 3100-3114, 1995.

Jeffries, H.E. and S. Tonnesen, A Comparison of Two Photochemical Reaction Mechanisms Using a Mass Balance Process, *Atmos. Environ.* , 28, 2991-3003.

Kurihara Y., G.J. Tripoli, and M.A. Bender, Design of a Movable Nested-Mesh Primitive Equation Model. *Mon. Wea. Rev.*, **107**, 239-249, 1979.

Mathur, R., L.K. Peters, and R.D. Saylor, Subgrid Representation of Emission Source Clusters in Regional Air Quality Models, *Atmos. Environ.*, 26A, 3219-3238, 1992.

Piccot, S.D., J.J. Watson, and J.W. Jones, A global Inventory of Volatile Organic Compound Emissions from Anthropogenic Sources, *J. Geophys. Res.*, 97, 9897-9912, 1992.

Smolarkiewicz P.K. and G.A. Grell, A Class of Monotone Interpolation Schemes. *J. Comput. Sci.*, **101**, 431-440, 1992.

Spiro, P.A., D.J. Jacob, and J.A. Logan, Global Inventory of Sulfur Emissions with $1^0 x 1^0$ Resolution, *J. Geophys. Res.*, 97, 6023-6036, 1992.

Stockwell, W.R., Middleton, P., Chang, J.S., The Second Generation Regional Acid Deposition Model Chemical Mechanism for Regional Air Quality Modeling, *J. Geophys. Res.*, 95(D10):16,343–16,367, 1990.

Funding Requested

NASA	Year 3
Direct Labor (ITD)	22,668
Benefits	9,430
Total Direct Labor	32,098
Direct Labor Overhead (ITD)	17,654
Expenses:	
Travel	1,500
Materials and Supplies	37
Publication Costs	
Equipment	
Subtotal Expenses	1,537
Total MCNC Incurred	51,289
Subcontract Expense:	
UNC	50,000
Total Subcontracts	50,000
Subtotal Contract Costs	101,289
General and Administration	8,711
TOTAL FUNDING REQUEST	110,000
Provisional Rates	
Benefits	41.60%
Tech Ovhd	55.00%
G & A	8.60%

7/1/97

Appendix A
Figures/Preliminary Results

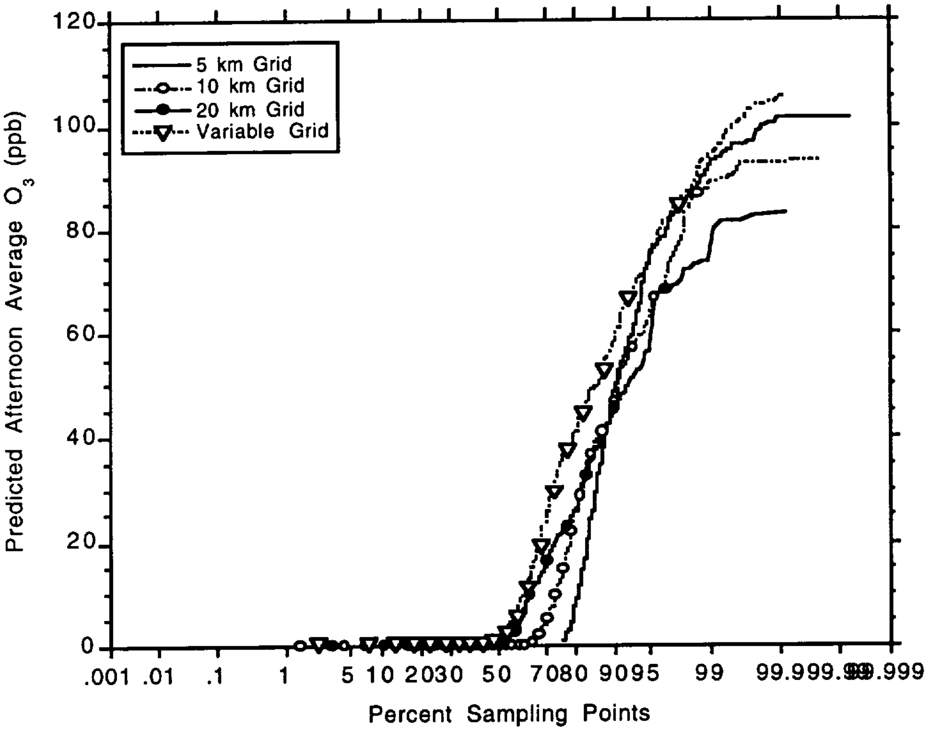

Figure A1. Predicted distribution of average O3 (ppb) for the different grid systems.

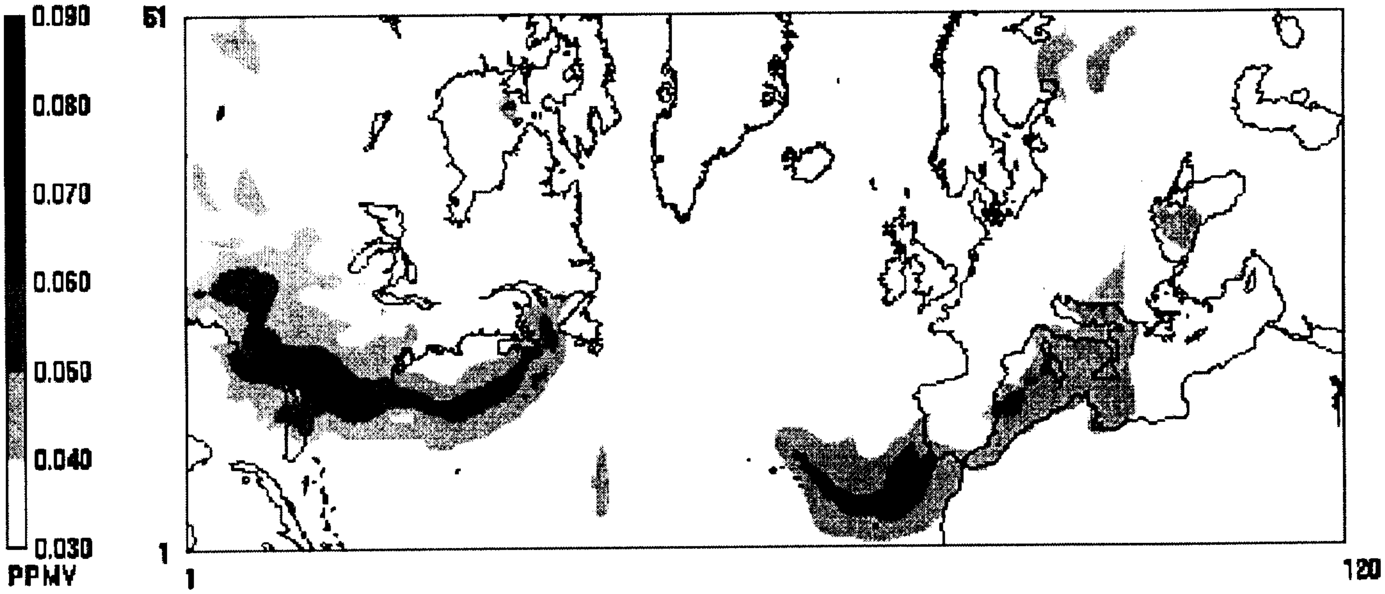

Figure A2 The north atlantic modeling domain. Also shown are surface level O3 predictions at 2300 GMT on April 18, 1994 (from preliminary model simulations).

Appendix B
Mathematics of the Morphecule Concept

Mathematics of the Morphecule Concept

Harvey Jeffries, Marc Kessler, and Michael Gery

June 30, 1997

1 New Representations Are Needed

The current methods for representing volatile organic compounds (VOC)s and NO_x atmospheric chemistry reaction systems in models is quite limited. For example, the popular Carbon Bond Four mechanism, presently used in many air quality models, has only 12 organic species that are used to represent the entire distribution of more than 300 VOCs in the urban atmosphere. To address arising needs to more fully represent the urban atmospheric content, we have created a new representation for the organic species in photochemical reaction models that permits much more detail to be retained while still being computationally efficient. The new method does require that additions be made to the model's solver system, but these are done without changing the ordinary differential equation (ODEs) system solver used by the models and instead auxiliary routines are added. We have also created new software that implements the representation and that produces the auxiliary routines needed.

1.1 Current Mechanism Representation Schemes

Previous methods applied one of three basic methods to represent VOCs:

- The "surrogate approach," in which the explicit chemistry of a particular member of a class of compounds was used to represent all members of the class (e.g., n-butane to represent all alkanes)
- The "lumped approach" in which a fictitious model species with properties determined by some type of kinetic or mole fraction weighting of the properties of all class members replace all members of a class (e.g., ALK7 in Carter, Atkinson, Lurmann mechanism represented all the alkanes from C_6 to C_9 by using a mole fraction weighted reaction rate constant and by having constant product stoichiometry that was calculated by a mole fraction weighting of values derived from the molecular stoichiometries of each of the individual alkanes)

- The "carbon bonding approach" in which the functional groups of each of the organic species are lumped together, e.g., all alkanes are represented by a single fictitious one-carbon model species named PAR that has properties averaged over the whole alkane distribution, or all terminal alkenes are represented by a single two-carbon model species named OLE plus one additional PAR species for each saturated carbon in the ligands attached to the double bond.

The only state memory in these representations is species concentration, and thus adding more detail to these representations has always involved increasing the number of model species. Adding new species, however, increases the number of stiff ODEs that must be solved to make predictions with the reaction mechanism. The difficulty of solution of the coupled ODE system increases at approximately the cube of the number of species, so increasing the number of species to obtain more representational detail is computationally expensive.

Furthermore, in these representations, the properties of the lumped or surrogate model species are held constant for the entire simulation. Even if the initial properties of the principle reacting organic species were computed dynamically from time step to time step, the existing modeling methods lose all such detail after the first reaction of the emitted VOC. This is because the reaction products—being static entities with constant properties—do *not* capture any of initial variation in properties. If one wants to remember more of the initial detail, then more intermediate and more final product species must be added to these mechanisms.

There is an increasing demand that these models perform better in a "one atmosphere" sense, that is, researchers are increasingly expecting these models to provide a virtual laboratory for investigating the interactive behavior of atmospheric systems involving natural and man-made emissions. This requires that the models must predict reliability a much wider variety of species than just ozone or sulfate deposition. For example, the 1990 Clear Air Act Amendments require assessments of: consumer product VOC reactivities; the effects of switching to reformulated fuels; the loss and secondary formation of a variety of organic air toxics; and, most recently, the formation of secondary particulate matter from gas-to-particle conversion due to chemical transformation of certain types of VOCs. All of these problems require a greatly expanded representation of VOC's detail chemical transformations and product production.

Further, as large regional and global models become increasingly used, the need to have efficient mechanisms that have the proper level of detail for the different regions of the model will increase. We t herefore seek a representation that will permit us to have a set of interacting mechanisms, each optimized for its scale. We have called these "adaptive chemistry" mechanisms.

1.2 Conceptual Basis of a New Representation

Our new representation is based on three principle concepts:

- It is conceptually easy to introduce more state memory into a chemical reaction representation in the form of auxiliary storage in variables, vectors, and arrays. Obstacles to implementing these operations have been the limited computer tools in use by the mechanisms designers. Assuming that we can overcome these obstacles, it should be possible to provide better representation without increasing the number of species and therefore the number of ODE's. A programmer who is hand-coding a reaction mechanism into an air quality model can easily introduce such additional storage and can easily write code that will perform calculations using this storage. An example of additional state memory might be the value of the average carbon number of a species that is both a reactant and a product and undergoes a reaction that leads to a change in the carbon number of the lumped representation. This is a property that influences the rate and even the stoichiometry of reactions, but does not change the number of ODE's. Non-ODE calculations can track this change, we just need a way for the mechanism designer to incorporate such calculations into the design.

- It is common to solve for concentrations in large air quality models by a technique called "operator splitting," in which various processes that actually occur simultaneously are solved for individually and sequentially while keeping the other process changes constant. Thus, when solving for advection, chemistry is not operating; after the advection change is over, the chemistry is solved assuming no transport. By applying a similar concept inbetween the calls to the normal ODE solver for the chemistry, or by even dividing the chemistry problem into two or more successive but interleaved problems, each operating with constant assumptions about the rate of other processes, we should be able to achieve a significant increase in representational power without a similar increase in computational burden.

- Software engineering and compiler construction techniques have advanced sufficiently that a "mechanism compiler" can be used to effectively treat the additional complexity introduced by applying the first two concepts to the chemical representational problem. In this work, a formal, context-free grammar for a new computer language (called *Morpho*) has been formulated and implemented in C++ to permit a higher-level abstract view of the problem to be created. Use of *Morpho* results in a very compact representation of the chemistry. All the tedious effort so often associated with hand coding of chemical reaction mechanisms into models is virtually eliminated by this software tool, yet run-time efficiency is not sacrificed as is the case when run-time interpretaton of a simple data structure is used to evaluate the right-hand-side of ODEs. Further, by formulating the core problem in a direct sparse matrix form, clarity of operations is achieved.

In the sections below, we will start with the more abstract view and move to the implementation and details in later sections.

1.3 Molecules, Morphecules, and Allomorphs

We divide our reacting entities into molecules, which have static properties for the whole simulation, and "morphecules," which are reacting entities like molecules but that have dynamic properties that change throughout the simulation.[1]

For each morphecule there are two or more "allomorphs," which are variants on a morphecule's shape (i.e., its properties). These allomorphs may themselves be either an explicit molecule with fixed properties or they may be a lumped-type species with time varying character (e.g., have a time-varying carbon number). For example, a morphecule called ALK-CO-H (alkyl aldehydes) could have allomorphs that would the explicit species Me, Et, Pro, ... C8 and the properties of the morphecule would be computed by the sums of allomorphic-concentration-weighting of properties of these allomorphs. Changes in the allomorphs are not represented by *coupled, non-linear* ODE equations, only the single ALK-CO-H species appears in reactions with the other molecules and morphecules, but the ALK-CO-H morphecule's reactive properties and production stoichiometry—while constant for one solver step—do change from solver step to solver step based on how its and other morphecule's allomorphs change. The allomorphic transformations (productions and loss) are separately represented and solved for in a decoupled and *linear,* or first order, sub-set of ODE's that are easily solved by simple explicit solvers. The allomorphic distributions are then used to update the properties of the parent morphecule before the next non-linear stiff ODE solver step occurs. Thus, not only will the concentration of the whole morphecule change over the simulation period, its allomorphic distribution, which determines the properties of the morphecule, will also change as well.

The minimum representation of a molecule is its name and its minimum state information is its concentration. The minimum representation of a morphecule is its name and the names of the associated allomorphs and its minimum state information are a total concentration (at all times assumed to be equal to the sum of the allomorph concentrations) and the mole fraction of each of its associated allomorphs. The use of morphecules as reactants in a given reaction requires that a vector of rate constants be given with one element for each reacting allomorph. The use of morphecules as products in a reaction requires that the product stoichiometry be given as a vector (if the reactants were all molecules) or a matrix (if a reactant was a morphecule).

[1] We also use auxiliary "ligand" morphecules that are associated with reactive-center morphecules, but that do not appear in the reactions themselves. These ligand morphecules hold information about the properties of the side-chains attached to the reacting portion of the morphecule. These can be used to determine the carbon mass attached to the radical centers and the stoichiometry of morphecule to morphecule transformation. In some ways our representation is a blend of the carbon bond approach and the lumped approach. When reacting center morphecules change, simple post-solver-step calculations are used to change the associated ligands. There can be a one-to-one or a one-to-many relationship among the reactive centers and the ligands permitting a wide variation in the level of detail, but effecting only the amount of computer storage while not increasing the ODE solver's computational demands.

2 The Formulation of the Problem

Figure 1 is a schematic that combines pictorial representations of the matrices and vectors with the formulas that solve the combined molecule/morphecule and allomorph prediction problems. As shown, there are three coupled problems

- the molecule/morphecule problem, in which we solve for the concentrations of molecules and morphecules using fixed properties of the morphecules computed from the mole fraction distribution of each morphecule's allomorphs at the beginning of the time step
- the allomorph problem, in which we solve for the distribution of mole fractions of a morphecule's allomorphs, given the concentration of the morphecules and molecules from the first problem
- the reconsiliation problem, in which we merge the two solutions above to produce an internally consistent set of molecule and morphecule concentrations, and a set of allomorph mole fractions that are different from those at the beginning of the time step.

2.1 Initialization

At $t = t_0$, the initial starting time,

$$\begin{aligned} \eta_t &= \eta_0 \\ \rho_t &= \rho_0 \\ \psi &= \psi_0 \end{aligned}$$

where

η is a vector of molecule and morphecule concentrations in which m is the number of molecules (all present in the first part of the vector), and M is the number of morphecules (all present in the last part of the vector)

ρ is a vector of allomorph mole fractions; i.e., ρ_j is the mole fraction of the v^{th} allomorph in the i^{th} morphecule and $j = \chi_i + v$ where χ_i is the sum of the number of allomorphs in all morphecules prior to the i^{th} morphecule

ψ is a vector of zeroth-order molecule and allomorph constant production rates, e.g., from emissions.

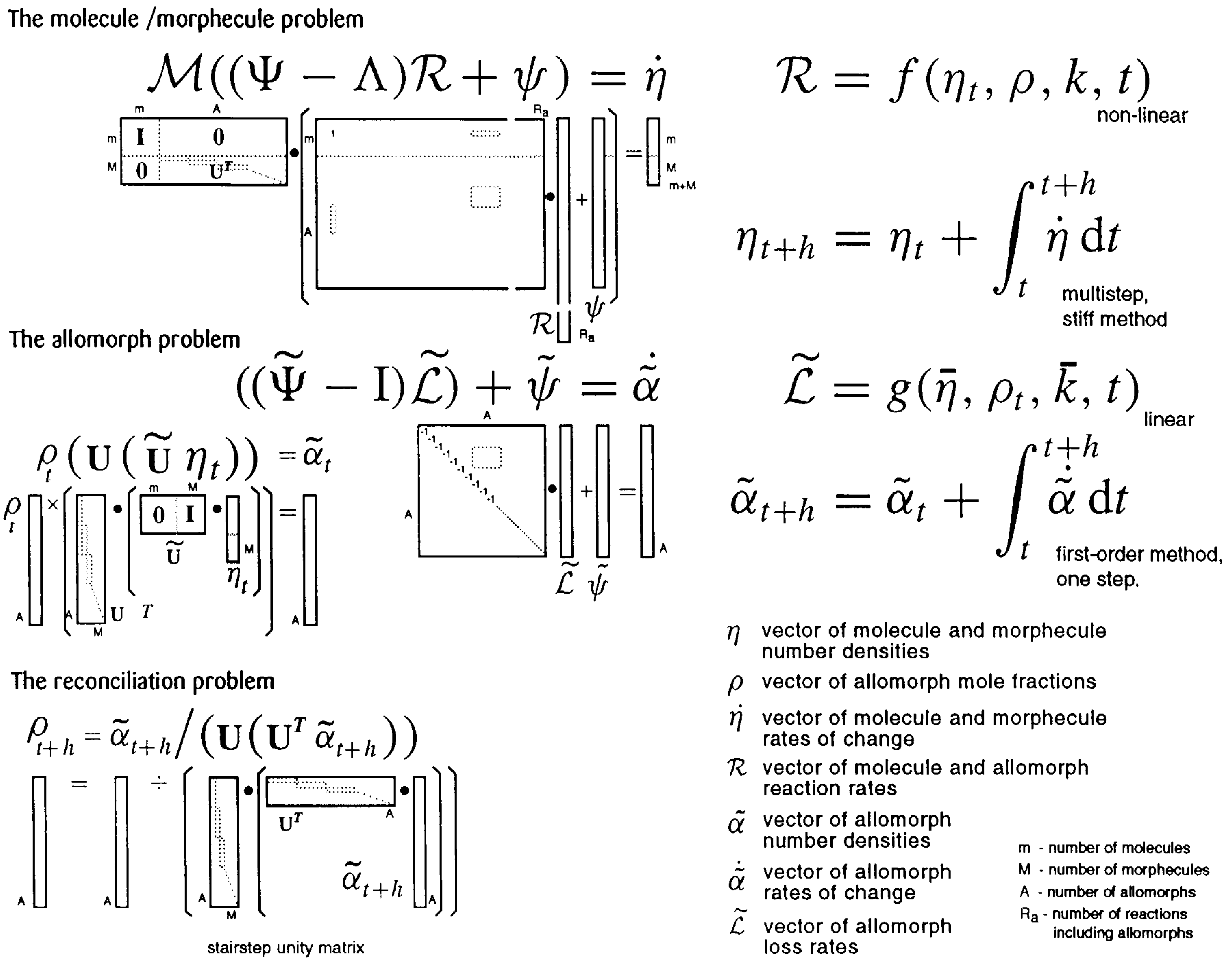
The molecule /morphecule problem
$\mathcal{M}((\Psi - \Lambda)\mathcal{R} + \psi) = \dot{\eta}$
m
A
m
M
I
0
0
U^T
m
A
1
R_a
$\mathcal{R}$
ψ
R_a
m
M
m+M
$\mathcal{R} = f(\eta_t, \rho, k, t)$
non-linear
$\eta_{t+h} = \eta_t + \int_t^{t+h} \dot{\eta}\,\mathrm{d}t$
multistep,
stiff method
The allomorph problem
$((\tilde{\Psi} - \mathrm{I})\tilde{\mathcal{L}}) + \tilde{\psi} = \dot{\tilde{\alpha}}$
$\rho_t(\mathbf{U}(\tilde{\mathbf{U}}\,\eta_t)) = \tilde{\alpha}_t$
ρ_t
×
m
M
0
I
$\tilde{\mathbf{U}}$
M
η_t
A
A
U
T
M
A
A
-1
A
A
$\tilde{\mathcal{L}}$
$\tilde{\psi}$
$\tilde{\mathcal{L}} = g(\bar{\eta}, \rho_t, \bar{k}, t)$
linear
$\tilde{\alpha}_{t+h} = \tilde{\alpha}_t + \int_t^{t+h} \dot{\tilde{\alpha}}\,\mathrm{d}t$
first-order method,
one step.
The reconciliation problem
$\rho_{t+h} = \tilde{\alpha}_{t+h} / (\mathbf{U}(\mathbf{U}^T\tilde{\alpha}_{t+h}))$
A
A
A
M
$\mathbf{U}^T$
A
$\tilde{\alpha}_{t+h}$
A
stairstep unity matrix
η vector of molecule and morphecule number densities
ρ vector of allomorph mole fractions
$\dot{\eta}$ vector of molecule and morphecule rates of change
$\mathcal{R}$ vector of molecule and allomorph reaction rates
$\tilde{\alpha}$ vector of allomorph number densities
$\dot{\tilde{\alpha}}$ vector of allomorph rates of change
$\tilde{\mathcal{L}}$ vector of allomorph loss rates
m - number of molecules
M - number of morphecules
A - number of allomorphs
R_a - number of reactions including allomorphs

Figure 1: A matrix-vector formulation of the molecule/morphecule/allomorph mechanism solution method.

2.2 The Molecule/Morphecule Problem

The first problem is the prediction of the molecule and morphecule concentration changes over one time step of size h. During this part, the mole fractions, ρ, of the allomorphs that make up each morphecule are held constant. The mole fractions enter the problem as arguments to the function f

$$\mathcal{R} = f(\eta_t, \rho, k_t, t)$$

where they are used with the morphecule concentrations in the vector η to compute the vector of reaction rates, $\mathcal{R}$, which has one element for each reaction among molecules and v elements for each reaction that has a morphecule as a reactant where v is the number of allomorphs in the reacting morphecule. For reactions among molecular species only,

$$\mathcal{R}_{\chi_r+1} = \begin{cases} {}^j k \eta_a & \text{for first order reactions} \\ {}^j k \eta_a \eta_b & \text{for second order reactions} \\ {}^j k \eta_a \eta_b \eta_c & \text{for third order reactions} \end{cases}$$

where j is the index of the reaction and ${}^j k$ is a scalar reaction rate constant and χ_r is defined below. Zeroth-order reactions, such as emissions, are specified in the ψ vector.

For each reaction that includes a morphecule as a reactant, a reaction rate is computed for each allomorph of the reacting molecule and the reaction rate constant must be given as a vector with one rate constant for each reacting allomorph,

$$\mathcal{R}_{\chi_r+v} = \begin{cases} {}^j \vec{k}_v \eta_a \rho_{\chi_a+v} & \text{for first order reactions} \\ {}^j \vec{k}_v \eta_a \rho_{\chi_a+v} \eta_b & \text{for second order reactions} \\ {}^j \vec{k}_v \eta_a \rho_{\chi_a+v} \eta_b \eta_c & \text{for third order reactions} \end{cases}$$

where χ_r is the number of reaction rates before the j_{th} reaction, η_a is a morphecule that is a reactant in the j_{th} reaction, η_b and η_c are molecules that are reactants, v is the index of the allomorphs that compose the morphecule, χ_a is the number of allomorphs in all morphecles prior to morphecule η_a, and ${}^j \vec{k}$ is a vector of v reaction rate constants. (Note that reactants can only contain one morphecule, but that the rate constant vector can be pseudo-first order in another morphecule).

Thus, the $\mathcal{R}$ vector has one entry for each molecule-only reaction and has v_ω entries for each reaction that has a morphecule reactant, where v_ω is the number of allomorphs in the reacting morphecule ω.

The Ψ matrix is a product by reaction stoichiometry matrix, where "product" includes each individual allomorph of the morphecules. All molecule-reactant to molecule-product reactions have only scalar entries in the upper portion of the matrix, i.e., the stoichiometry is a single scalar. Molecule-only reactions that produce a morphecule have vertical vector entries in the lower part of Ψ, and morphecule reactions that produce molecules have row vector entries in the upper part of Ψ. Morphecule reactants that produce morphecule products have sub-matrix entries in Ψ. The Λ matrix is the reactant by reaction loss stoichiometry matrix (this is mostly filled with 1's). Even after these two matrices are combined by subtraction, the resulting matrix is very sparse. In fact, individual sub-matrices for morphecule to morphecule reactions are usually very sparse, e.g., the stoichiometry for a aldehyde morphecule reacting to a peroxy radical morphecule has a allomorph to allomorph stoichiometry similar to

$$\begin{bmatrix} 0.0 & 1.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.03 & 0.70 & 0.2 & 0.02 & 0.05 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 1.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 1.0 \end{bmatrix}$$

The numbers in row two are updated before each solver time step to reflect the change in reactant allomorph distribution that occurred during the last step. Both the Ψ and the Λ matrices can be build by the mechanism compiler which also creates a subroutine to update the individual elements in the Ψ matrix as the allomorph composition changes based on assignment statements supplied by the mechanism's author.

The mechanism compiler can also create a constant **U** matrix that is a stairstep unity matrix which reflects the member relationship between morphecules and allomorphs. The mechanism compiler can also build a constant $\mathcal{M}$ matrix from an identity matrix and the $\mathbf{U}^T$ matrix that is used to combine the production and loss of molecules and allomorphs into the production and loss of molecules and morphecules.

The dot product of the $\mathcal{M}$ matrix with the dot product of the $(\Psi - \Lambda)$ matrix and the $\mathcal{R}$ vector augmented by summing in the zeroth-order rates in the ψ matrix performs all the "lumping calculations" of allomorphs into morphecules, and computes the rate of change of all the molecules and morphecules,

$$\dot{\eta} = \mathcal{M} \cdot ((\Psi - \Lambda) \cdot \mathcal{R}) + \psi)$$

(see Figure 1 for more details on these operations).

The concentrations of the molecules and morphecules at the start of the time step and the $\dot{\eta}$ are used to compute the concentrations at the end of the time step by using a multistep, stiff integration method such as Gear's method, or some other type of coupled, stiff ODE integrators in common use,

$$\eta_{t+h} = \eta_t + \int_t^{t+h} \dot{\eta}\, \mathrm{d}t$$

and this solves the first problem.

2.3 The Allomorphic Distribution Problem

During the solution of the first problem, the mole fractions ρ of morphecules' allomorphs were maintained constant over the interval h. If we use these mole fractions to partition the morphecule masses at $t + h$ back to their allomorphs, we would be assuming that each allomorph contributes only proportionally to its mass and we would ignore the differences in k values of each allomorph. Thus, for example in situations where the mole fractions are similar, in the solution to the first problem, we have reacted the slower allomorphs too fast and the faster allomorphs too slow.

Suppose we have 1.000 concentration units of a morphecule (ω) that has two allomorphs (α_s, α_f), which initially have equal mole fractions, $\rho = (0.5000, 0.5000)$, but with rate constants that differ by a factor of 5, e.g., $\vec{k} = (10.0, 2.0)$. Typically, time steps are limited by most solver systems so that significant reactants change by less than 0.02 of their initial concentration over the time step. Let us assume that the solver system has chosen a value of h such that in combination with the concentration of the molecular species that are reacting with ω, we just achieve the 0.02 change. Therefore, if we use a mole-fraction-weighted rate constant of 6.0 for our ω morphecule, would achieve a morphecule concentration of 0.9800 at $t + h$. The mole fraction of the two allomorphs would still be (0.5000, 0.5000), giving concentrations of the allomorphs of (0.4900, 0.4900). Now, instead of using the mole-fraction-weighted rate constant, let us solve for each of the allomorphs with their own rate constants (but assuming that the rest of the conditions remain the same, i.e., the concentration of any reaction molecules). The sum of concentrations for both allomorphs using this method is 0.9801, or 0.01% higher than when we used the average k. More importantly, the mole fractions under this scenario would be (0.5067, 0.4933). That is, in calculating the mass that reacted in a morphecule reaction using a mole-fraction-weighted rate constant, we have over-consumed the slower reacting allomorph and under-consumed the faster reacting allomorph. This also implies that we have over-produced the products of the slower allomorph and under-produced the products of the faster allomorph, while at the same time producing a total sum of products that is consistent with the total amount of reacted morphecule. This example is oversimplified in that we did not consider the simultaneous production of the ω morphecule from reactions of other morphecules along with the loss of the initial amount of ω present, but it nevertheless illustrates the problems adequately.

There are potentially a large number of ways in which we could adjust the allomorph mole fractions to approximate better the true behavior of a fully coupled system of molecules and allomorph ODEs, however, there is no way to satisfy the conditions in both problems presented here at the same time. The approach we adopt is to solve a decoupled allomorph problem over the same time step h using average or linear conditions for the concentrations of the molecules as determined in the solution to the first problem. This linear problem can be solved using a much simpler first-order ODE solver to find the allomorph concentrations at $t + h$. Then we reconcile the second solution with the first solution. In computing just the rates of change of the allomorph concentrations, we approximate the molecular species concentrations using the results of the first problem (e.g., molecular concentrations may be average over h, or they may vary linearly from their starting to ending concentrations over the interval). This results in a set of linear, non-stiff ODE's that can be solved using simple explicit integration techniques.

We compute the starting allomorph concentrations for the time step from the initial mole fractions and the initial morphecule concentrations, (the matrix $\widetilde{\mathbf{U}}$, which is a concatenated null and identity matrix, selects just the morphecule sub-vector from the η vector),

$$\tilde{\alpha}_t = \rho_t \times (\mathbf{U} \cdot (\widetilde{\mathbf{U}} \cdot \eta_t)).$$

To compute the rates of change of the allomorph species we use,

$$\dot{\tilde{\alpha}} = ((\widetilde{\Psi} - \mathbf{I}) \cdot \widetilde{\mathcal{L}}) + \tilde{\psi}.$$

The first-order reaction losses of the allomorphs, $\widetilde{\mathcal{L}}$, are computed by a function,

$$\widetilde{\mathcal{L}} = g(\bar{\eta}, \rho_t, \bar{k}, t).$$

constructed by the mechanism compiler similar to the way $\mathcal{R}$ was constructed. Each element of $\widetilde{\mathcal{L}}$ consists of the product of a element of $\tilde{\alpha}$ times a sum of k, $k\bar{\eta}_b$, or $k\bar{\eta}_b\bar{\eta}_c$ terms, all of which are constant because we assume that molecules are constant for this step.

The $\tilde{\psi}$ vector contains constant production rates of allomorphs arising from reactions among just molecules, plus any other sources of allomorphs, such as emissions.

The $\widetilde{\Psi}$ square matrix contains the morphecule to morphecule production stoichiometry; these are the same as those that were used to build the Ψ matrix. The allomorph loss stoichiometry is just an identity matrix **I**.

Given the initial allomorph concentrations $\tilde{\alpha}_t$ and the rates of change $\dot{\tilde{\alpha}}$, we can perform an integration using a simple explicit (i.e., fast) method,

$$\tilde{\alpha}_{t+h} = \tilde{\alpha}_t + \int_t^{t+h} \dot{\tilde{\alpha}}\, \mathrm{d}t$$

and this solves the second problem.

2.4 The Reconciliation Problem

During the solution of the first problem, the mole fractions of the morphecules' allomorphs, ρ, were maintained constant over the interval h. During the solution of the second problem, the molecular concentrations, η, were held constant over the interval while the allomorphic concentrations at the end of the time step, $\tilde{\alpha}_{t+h}$, were computed. From the latter we can compute a new set of mole fractions for the morphecules. Note, however, that when we convert these α-concentrations into morphecule concentrations to compute the mole fractions, we will most likely obtain morphecule concentrations that will differ slightly from those found in the first solution. Because the first solver is more accurate than the second solver, we will accept the first solver's morphecule concentrations, but we will assume the second solver's mole fractions are an acceptablly accurate approximation of the distribution of allomorphs within each morphecule at the end of the time step. That is, we depend on the high accuracy, stiff equation solver that is integrating both the molecules and the morphecule concentrations to determine the molecule and morphecule concentrations—and thus the sum of allomorph concentrations for each morphecule—but we use the second solver to approximate the allomorph distribution within this sum. Thus, the two results are reconciled in a way that preserves the accuracy and stability of the more powerful first solver or integrator.

The first operation in the reconciliation problem is to compute a vector that has in each element the sum of the allomorph concentrations that belong to each morphecule. So if the first morphecule has five allomorphs, then the first five positions of this vector will *all* have the sum of the first five elements of $\tilde{\alpha}_{t+h}$. This is accomplished with the aid of the matrix, U, which is the same morphecule by allomorph stairstep matrix produced by the mechanism compiler that was already described above. This sum matrix is used to do an element-by-element division of the vector $\tilde{\alpha}_{t+h}$ to create a new ρ vector. The required operations can be written as,

$$\rho_{t+h} = \tilde{\alpha}_{t+h} / \mathbf{U} \cdot (\mathbf{U}^T \cdot \tilde{\alpha}_{t+h})$$

Finally, $\hat{\alpha}_{t+h}$, the allomorph concentrations that are consistent with both the morphecule concentrations from the first problem and the mole fractions from the second problem are computed by,

$$\hat{\alpha}_{t+h} = \rho_{t+h} \times (\mathbf{U} \cdot (\widetilde{\mathbf{U}} \cdot \eta_{t+h})).$$

and this solves the reconsiliation problem.

2.5 Error Analysis

We can compute a measure of how closely the two different problems agree by computing an error term, ϵ. First, we compute the individual allomorph errors in the vector β,

$$\beta = \zeta \times \left(\frac{\tilde{\alpha}_{t+h} - \hat{\alpha}_{t+h}}{\hat{\alpha}_{t+h}} \right)$$

where ζ is a vector that permits us to give some allomorphs more or less weight than others. A single measure of disagreement can be obtained by

$$\epsilon = \sqrt{\beta^T \cdot \beta}$$

which computes the root mean square difference of the two methods. This value can be monitored to determine when the step size, h, was too big for the approximations in the two methods to meet a desired error tolerance.

3 Implementation

3.1 Storage

Assume that our working reaction mechanism will have 15 molecules (mostly inorganic) and 35 morphecules (all organic) for a total of 50 species to be solved by the stiff solver. Assume that each morphecule has on the average 8 allomorphs, then we will have 295 total molecules and allomorphs and the matrix $\widetilde{\Psi}$ will be a 295 by 295 or 87,025 elements. The total number of reactions in a molecule/morphecule mechanism with a high resolution organic design will be about 150 reactions and the total size of the $\mathcal{R}$ vector will be about 400. So the size of the Ψ matix will be 295 by 400 or 118,000 elements. Given that the majority of the morphecules do not interact with each other, that typically one morpecule and one molecule react to produce about two morphecules, the the typical number of sub-matrices under a set of related $\mathcal{R}$'s is three with each morphecule having about 8 allomorphs. Also given the typical morphecule to morphecule stoichiometry in current morphecule reaction mechanisms (as shown in the example above), each product sub-matrix will have less than one-third fill in. Thus if there are about 100 morphecule reactions with about 64 elements per morphecule reaction, the Ψ matix will have about 6400 non-zero elements. Even if we double this number, both $\widetilde{\Psi}$ and Ψ matices will have only about 10% of their total number of elements filled in. Sparse matrix storage and operations methods will be central to the sucessful implementation of an allomorphic mechanism. Further, sparse methods that permit the construction and updating of a block structured sparse matrix would be best suited for this problem.

3.2 Code Production

We have searched the internet and the library for sparse matrix methods and we find many methods suited for iterative methods, but few are optimized for direct operations and none are block oriented.

Kessler has designed, implemented in C++, and tested a complete sparse matrix library for composable block matrices. This was done when it appeared that we would need to invert the $\widetilde{\Psi}$ matrix. This sparse matrix package makes extensive use of inherentance and C++ template functions. It is impossible to duplicate its functionality in C or FORTRAN.

After considering the difficulties of moving this full code to the CRAY and the complexities of instructions to aggregate the Ψ matix, we hae decided to forego for now the solution of these problems by matrix algebra as shown in Figure 1. Instead, because the matrices are so sparse, we will use the mechanism compiler to just write out subroutines that will perform the equilivant element by element operations that would have occurred with the matrix algebra operations.

The multistep, stiff method for integrating $\acute{\eta}$ already exists in the model. A simple Euler forward or Runga-Kutta explicit integrator will have to be added to the model.

4 Future Work

We have a *Morpho* compiler that runs on Windows-NT, Windows-95, and Macintoshs that currently has a "back-end" that writes out storage instructions and calculator program instructions for the virtual calculator in the single cell solver program which is used to develop and test the formulation of molecule and morphecule mechanisms. This compiler can also be produced on any platform that has a C++ compiler that supports the current draft C++ standard library, supports templates with default arguments, and supports exceptions. Public domain C++ compilers like gnu's are not able to do this yet, but many workstation vendor's C++ compilers are.

Our next task for this project is to derive a new *Morpho* compiler backend that outputs C/C++ storage declarations and C/C++ functions that will evaluate the various operations shown in Figure 1. Thus, we need the following "calculator" functions:

k – calculator a function that evaluates, for each molecule-only reaction, a scalar of the current value of the rate constant expressions as a function of temperature and pressure obtained from the environment vector.

Aks – calculator a function that evaluates, for each morphecule reaction, a vector of current values of rate constant expressions, where the vector has one element for for each allomorph that is reacting.

Arrs – calculator a function that evaluates, for each morphecule reaction, a vector of reaction rates, where the vector has one element for for each allomorph that is reacting.

rr – calculatator a function that evaluates, for each molecule-only reaction, a scalar of the reaction rate.

Lam – calculator a function that evaluates, for each molecule and morphecule, a scalar of its loss rate.

Phi – calculator a function that evaluates, for each molecule and morphecule, a scalar of its production rate.

Alpha – calculator a function that evaluates, for each allomorph, a scalar of its concentration using the morphecule concentration and the current mole fractions.

Rho – calculator a function that evaluates, for each morphecule, for each of its allomorphs, a scalar of its mole fraction using the allomorph concentrations.

M – calculator a function that computes, for each morphecule, the morphecule concentration from its allomorphs.

ALs – calculator a function that evaluates, for each morphecule, a vector of allomorph losses, where the vector has one element for for each allomorph that is reacting.

APs – calculator a function that evaluates, for each morphecule, a vector of allomorph productions, where the vector has one element for for each allomorph that is reacting.

t – psi – calculator a function that evaluates, for each morphecule, a vector of allomorph zeroth-order productions, where the vector has one element for for each allomorph that is reacting.

To support these functions we will need to create a high-level, run-time library that implements *Morpho*'s built-in functions, e.g., TROE() and the functions produced by the compiler to efficiently implement the calculators above, e.g., a mole fraction calculator.

Once these are finished we can begin testing the mechanism evaluator code with the mechanism solver codes in the Eulerian model.

We will begin by replicating an existing molecule only reaction mechanism such as the Carbon Bond Four to assure that we can obtain the same results as the currently hard-wired version.

Once this is accomplished, we will begin with a single molecule/morphecule mechanism over the whole domain in both the fine grid and the coarse grid portions of the model.

While this testing is occurring, we will design two molecule/morphecule mechanisms that have the same number of morphecules and allomorphs, but will have the allomorphs allocated among the morphecules differently. One will be designed to provide for more morphecules to represent the fast reacting organics and a much smaller number of morphecules will represent the slower reacting organics. The other mechanism will be the opposite.

Note that the only differences in these two mechanisms will be the values of ρ, k, and the two functions $\mathcal{R}$ and $\widetilde{\mathcal{L}}$ in Figure 1.

Once the model is functioning with a single preliminary molecule/morphecule mechanism over the whole domain, we will implement two matched mechanisms over the two different scale domains of the model and we will add "interface" code that will match a set of process rates across the transistion from one grid resolution/mechanism to the other.