ABSTRACT

SRIVASTAVA, RAVI K. An Adaptive Grid Algorithm for Air Quality Modeling.
(Under the direction of Dr. D. Scott McRae.)

The physical and chemical processes responsible for air pollution span a wide range of spatial scales. For example, there may be point sources, such as power plants that are characterized by relatively small spatial scales compared to the size of the region that may be impacted by such sources. To obtain accurate distributions of pollutants in an air quality simulation, the pertinent spatial scales can be resolved by varying the physical grid node spacing.

A new dynamic adaptive grid algorithm, the *Dynamic Solution Adaptive Grid Algorithm - PPM* (DSAGA-PPM), is developed for use in air quality modeling. Given a fixed number of grid nodes, DSAGA-PPM distributes these nodes in response to spatial resolution requirements of the solution field and then updates the solution field based on the resulting distribution of nodes. DSAGA-PPM is implemented dynamically to resolve any evolving solution features.

Tests with model problems demonstrate that DSAGA-PPM calculates advection much more accurately than the corresponding static grid algorithm (SGA-PPM) and, therefore, would assure more accurate starting concentrations for chemistry calculations. For example, after one revolution of four rotating cones, 87% of each of the cone peaks is

retained using DSAGA-PPM while only 63% is retained using SGA-PPM. The root-mean-square errors in DSAGA-PPM results are about 4-5 times lower than those in the corresponding SGA-PPM results.

Tests with reacting species and sources demonstrate that DSAGA-PPM provides the needed solution resolution. In simulations of a rotating and reacting conical puff, the root-mean-square errors in DSAGA-PPM results are about 4-6 times lower than those in the corresponding SGA-PPM results. In simulations of a power plant plume, the DSAGA-PPM solution reflects the early, the intermediate, and the mature stages of plume development; these stages are not seen in the corresponding SGA-PPM solution. Finally, it is demonstrated that DSAGA-PPM provides an accurate description of the ozone production resulting due to dynamic interactions between emissions from two power plants and an urban area. In general, these results reflect that DSAGA-PPM is able to provide accurate spatial and temporal resolution of rapidly changing and complex concentration fields.

Performance achieved by DSAGA-PPM in model problem simulations indicates that it can provide accurate air quality modeling solutions at costs 10 times less than those incurred in obtaining equivalent static grid solutions.

AN ADAPTIVE GRID ALGORITHM FOR AIR QUALITY MODELING

by

Ravi. K. Srivastava

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Department of Mechanical and Aerospace Engineering

Raleigh

1998

Approved By:

Dr. Robert White

Dr. F. DeJarnette

Dr. M. Talat Odman

Dr. D. Scott McRae, Chair

DEDICATION

In memory of my father:

Krishna Kant Srivastava

(December 4, 1926 - January 18, 1995)

BIOGRAPHY

Ravi K. Srivastava was born in Varanasi, India, on March 26, 1955. After graduating from Methodist High School in 1974, he attended Indian Institute of Technology, Kanpur (IIT Kanpur). Ravi received a Bachelor of Science in Mechanical Engineering from IIT Kanpur in 1979. From 1979 to 1982, Ravi worked with several companies in India including Larsen & Toubro Ltd., Perfect Machine Tools Co., and Crompton and Greaves Ltd. In September 1982, he began graduate study at Washington State University (WSU). Ravi received his Master of Science in Mechanical Engineering from WSU in December 1984. From 1985 through 1994, he worked as a combustion engineer with Acurex Environmental Corporation. During this period, Ravi published numerous papers on air pollution control and coinvented a sorbent injection process designed to control combustion generated metal aerosol. For this process, Ravi and other coinventors obtained a U.S. patent. In 1989, he entered in the Ph.D. program in aerospace engineering at North Carolina State University. Since 1994, Ravi is employed with the Acid Rain Division of the U.S. Environmental Protection Agency in Washington, D.C. In his current responsibilities, he reviews and conducts technical and policy analyses required in developing air pollution control regulations and programs.

ACKNOWLEDGEMENTS

I would like to acknowledge and thank everyone who made the completion of this dissertation possible. I would especially like to thank my advisor, Professor D. Scott McRae, for his encouragement, guidance, and patience through the years. My appreciation and thanks to Dr. M. Talat Odman for his numerous helpful insights and suggestions that guided the development of this dissertation. Thanks also to Dr. F.R. DeJarnette and Dr. R.E. White, my other committee members. Many thanks to Mr. Larry Kertcher, my supervisor, for allowing me time away from work at critical junctures.

Finally, I would like to thank my family and friends who have been a constant source of support for me through the years of this research. I especially thank my wife Jenny, my daughter Katie, and my mother Kusum for their patience, love, and encouragement. Thanks are also extended to Mr. Dominic Mancini and Mr. Peter Tsirigotis for providing me patient ears every now and then.

This research was supported by the U.S. Environmental Protection Agency under Cooperative Agreement Number CR822053-01, Task 4.2.3. The computing resources were provided by the North Carolina Supercomputing Center.

TABLE OF CONTENTS

List of Tables.....	vii
List of Figures	viii
List of Symbols	xiv
1 Introduction	1
2 Governing Equations and Boundary Conditions	7
2.1 Derivation of the Governing Equations.....	9
2.2 Initial and Boundary Conditions	19
2.3 Coordinate Systems.....	21
2.3.1 Horizontal Coordinate Systems.....	22
2.3.2 Vertical Coordinate Systems.....	25
2.3.3 Coordinate Transformations.....	25
3 Elements of DSAGA-PPM.....	27
3.1 Time Advancement of the Governing System	27
3.2 Treatment of Two - Dimensional Advection	33
3.3 Treatment of Two - Dimensional Turbulent Diffusion	40
3.4 Boundary Conditions for Transport.....	46
3.5 Treatment of Emission Sources.....	48
3.6 Computation of Chemistry	51
3.7 Grid Adaptation and Solution Correction	55
3.7.1 SIERRA Weight Function.....	55
3.7.2 Repositioning of Grid Nodes.....	59
3.7.3 Solution Redistribution	60
3.7.4 Grid Convergence.....	64
3.7.5 Preadaptation	65
4 Model Advection Problems.....	67
4.1 One-dimensional Tests	69
4.2 A Rotating Conical Distribution	76
4.3 Four Rotating Conical Distributions	83

5	Weight Function for Flows with Reacting Species	91
6	A Rotating and Reacting Conical Pollutant Puff	98
6.1	Chemical Mechanism	98
6.2	Description of the Model Problem	99
6.3	Reference Solution	102
6.4	Simulation for 150 Seconds	105
6.5	Simulation for 86,400 Seconds	121
7	Dispersion of a Power Plant Plume.....	135
7.1	Description of the Model Problem	136
7.2	Simulation for 40,000 Seconds	141
7.3	Assessment of the Accuracy of DSAGA-PPM Predictions	153
8	Simulation with Multiple Sources.....	159
8.1	Description of the Model Problem	159
8.2	Simulation for 40,000 Seconds	163
8.3	Simulation Without the Upwind Point Source	175
8.4	Evolution of the Ozone Field	178
8.5	Assessment of the Accuracy of DSAGA-PPM Predictions	189
9	Computational Performance	195
10	Summary and Conclusions	200
	References	205
	Appendix A - Coordinate Systems	213
A.1	Tangential Cartesian Coordinates	214
A.2	Spherical Coordinates.....	216
A.3	Generalized Vertical Coordinate	219
A.4	Summary	223
A.5	References	227
	Appendix B - Grid Metric Terms	228
	Appendix C - The Piecewise Parabolic Method	231
	Appendix D - The Asymptotic Integration Method	238

List of Tables

2.1	Some Generalized Vertical Coordinates	26
4.1	Values of Adaptation Parameters Used in One-Dimensional Tests.....	70
4.2	Error Characteristics for the One-Dimensional Tests	71
4.3	Error Characteristics for the Simulations of a Rotating Cone.....	82
4.4	Error Characteristics for the Simulations of Multiple Rotating Cones	88
6.1	A Simplified Chemical Mechanism for Photochemical Production of Ozone.....	99
6.2	Initial Concentrations of Species in the Pollutant Puff	101
6.3	EVALLEY and EPEAK for the 150 Seconds Simulation of the Puff.....	120
6.4	EMAS, EVAR, and ERMS for the 150 Seconds Simulation of the Puff.....	120
6.5	EVALLEY and EPEAK for the 86,400 Seconds Simulation of the Puff.....	134
6.6	EMAS, EVAR, and ERMS for the 86,400 Seconds Simulation of the Puff	134
7.1	Initial Concentrations and Source Terms for the Power Plant Plume Model Problem.....	139
8.1	VOC and NO _x Initial Concentrations and Source Terms for the Multiple Sources Model Problem	162
8.2	Cell Volumes in the Vicinity of the Downwind Point Source	176
8.3	O ₃ Metrics for Each of the Simulation Results	193
9.1	CPU Times Associated With Rotating Cone Simulations	196
9.2	CPU Times Associated With Simulations of Multiple Sources.....	198
A.1	Some Generalized Vertical Coordinates	226

List of Figures

2.1	Tangential Cartesian Coordinates	23
2.2	Spherical Coordinates.....	24
3.1	The five-point stencil for computing discretized Laplacian.....	57
3.2	Grid node o and neighboring cells used in the <i>center-of-mass</i> scheme....	60
4.1	Rectangular waveform solution obtained with (a) SGA-PPM (b) DSAGA-PPM	73
4.2	Triangular waveform solution obtained with (a) SGA-PPM (b) DSAGA-PPM	74
4.3	Gaussian waveform solution obtained with (a) SGA-PPM (b) DSAGA-PPM	75
4.4	Preadapted grid showing clustering of nodes near the cone.....	78
4.5	Solution field (a) with oscillations (b) without oscillations	79
4.6	Rotating cone results obtained with (a) SGA-PPM (b) DSAGA-PPM	80
4.7	Adaptive grid after one revolution of the cone.....	81
4.8	Preadapted grid showing clustering of nodes near the cones.....	84
4.9	Results for four cones obtained with (a) SGA-PPM (b) DSAGA-PPM	86
4.10	Adaptive grid after one revolution of the cones.....	87
4.11	DSAGA-PPM solution obtained using a refined grid with 85 x 85 nodes	89
4.12	Adaptive refined (85 x 85 nodes) grid after one revolution of the cones.....	90

6.1	Changes in concentrations of selected species over 150 seconds (a) peak concentrations (b) background concentrations.....	103
6.2	Changes in concentrations of selected species over 86,400 seconds (a) peak concentrations (b) background concentrations.....	104
6.3	Preadapted grid reflecting clustering of nodes in and around the pollutant puff	106
6.4	NO distribution after 150 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution.....	108
6.5	NO distribution after 150 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA solution.....	109
6.6	NO ₂ distribution after 150 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution.....	110
6.7	NO ₂ distribution after 150 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA solution.....	111
6.8	O ₃ distribution after 150 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution.....	112
6.9	O ₃ distribution after 150 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA solution.....	113
6.10	HCHO distribution after 150 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution.....	114
6.11	HCHO distribution after 150 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA solution.....	115
6.12	Adaptive grid after 150 seconds reflecting clustering of nodes in and around the pollutant puff	116
6.13	NO distribution after 86,400 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution.....	123
6.14	NO distribution after 86,400 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA-PPM solution.....	124

6.15	NO ₂ distribution after 86,400 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution.....	125
6.16	NO ₂ distribution after 86,400 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA-PPM solution.....	126
6.17	O ₃ distribution after 86,400 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution.....	127
6.18	O ₃ distribution after 86,400 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA-PPM solution.....	128
6.19	HCHO distribution after 86,400 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution.....	129
6.20	HCHO distribution after 86,400 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DASGA-PPM solution.....	130
6.21	Adaptive grid after 86,400 seconds reflecting clustering of nodes in and around the pollutant puff.....	131
7.1	Preadapted grid reflecting clustering of nodes near the point source.....	140
7.2	SGA-PPM prediction of NO concentration after 40,000 seconds.....	142
7.3	DSAGA-PPM prediction of NO concentration after 40,000 seconds.....	142
7.4	SGA-PPM prediction of O ₃ concentration after 40,000 seconds	143
7.5	DSAGA-PPM prediction of O ₃ concentration after 40,000 seconds	143
7.6	Three-dimensional depiction of SGA-PPM results for O ₃ concentration at 40,000 seconds.....	144
7.7	Three-dimensional depiction of DSAGA-PPM results for O ₃ concentration at 40,000 seconds.....	144
7.8(a)	SGA-PPM and DSAGA-PPM predictions of O ₃ concentration in plume cross-section at a distance of 10 km downwind of the source. SGA concentrations are depicted by the dotted curve	147

7.8(b)	SGA-PPM and DSAGA-PPM predictions of O ₃ concentration in plume cross-section at a distance of (b) 30 km downwind of the source. SGA concentrations are depicted by the dotted curve	148
7.8(c)	SGA-PPM and DSAGA-PPM predictions of O ₃ concentration in plume cross-section at a distance of 135 km downwind of the source. SGA concentrations are depicted by the dotted curve	149
7.9	DSAGA-PPM prediction of O ₃ concentration field at 3600 seconds	151
7.10	DSAGA-PPM prediction of O ₃ concentration field at 10,800 seconds ..	151
7.11	Adaptive grid after 40,000 seconds reflecting clustering of nodes near the source and in the plume	152
7.12	Refined grid SGA-PPM prediction of O ₃ concentration field at 40,000 seconds	155
7.13(a)	Refined grid SGA-PPM and DSAGA-PPM predictions of O ₃ concentration in plume cross-section at a distance of 10 km downwind of the source. Refined grid SGA concentrations are depicted by the dotted curve.....	156
7.13(b)	Refined grid SGA-PPM and DSAGA-PPM predictions of O ₃ concentration in plume cross-section at a distance of 30 km downwind of the source. Refined grid SGA concentrations are depicted by the dotted curve.....	157
7.13(c)	Refined grid SGA-PPM and DSAGA-PPM predictions of O ₃ concentration in plume cross-section at a distance of 135 km downwind of the source. Refined grid SGA concentrations are depicted by the dotted curve.....	158
8.1	Preadapted grid reflecting clustering of nodes near the point sources and in and around the urban area source.....	165
8.2	SGA-PPM prediction of NO concentration after 40,000 seconds.....	166
8.3	DSAGA-PPM prediction of NO concentration after 40,000 seconds.....	166
8.4	SGA-PPM prediction of O ₃ concentration after 40,000 seconds	167

8.5	DSAGA-PPM prediction of O ₃ concentration after 40,000 seconds	167
8.6	Three - dimensional depiction of the O ₃ concentration field resulting from the SGA-PPM simulation for 40,000 seconds.....	168
8.7	Three - dimensional depiction of the O ₃ concentration field resulting from the DSAGA-PPM simulation for 40,000 seconds.....	168
8.8	Adaptive grid after 40,000 seconds reflecting clustering of nodes around the upwind point source and in and around the urban area.....	172
8.9(a)	OH weight function at 40,000 seconds	173
8.9(b)	HO ₂ weight function at 40,000 seconds	173
8.9(c)	O ₃ weight function at 40,000 seconds	174
8.9(d)	Final smoothed weight function at 40,000 seconds	174
8.10	SGA-PPM prediction of O ₃ concentration field at 40,000 seconds in simulation without the upwind point source	177
8.11	DSAGA-PPM prediction of O ₃ concentration field at 40,000 seconds in simulation without the upwind point source	177
8.12	DSAGA-PPM prediction of NO concentration field at 3,600 seconds	179
8.13	DSAGA-PPM prediction of OH concentration field at 3,600 seconds	179
8.14	DSAGA-PPM prediction of HO ₂ concentration field at 3,600 seconds	180
8.15	DSAGA-PPM prediction of O ₃ concentration field at 3,600 seconds	180
8.16	Adaptive grid at 3,600 seconds reflecting clustering of nodes at the point sources and in and around the urban area.....	183
8.17	DSAGA-PPM prediction of NO concentration field at 25,200 seconds	184

8.18	DSAGA-PPM prediction of OH concentration field at 25,200 seconds	184
8.19	DSAGA-PPM prediction of HO ₂ concentration field at 25,200 seconds	185
8.20	DSAGA-PPM prediction of O ₃ concentration field at 25,200 seconds	185
8.21	Adaptive grid at 25,200 seconds reflecting clustering of nodes around the upwind point source and in and around the urban area.....	187
8.22	DSAGA-PPM prediction of NO concentration field at 39,600 seconds	189
8.23	DSAGA-PPM prediction of OH concentration field at 39,600 seconds	189
8.24	DSAGA-PPM prediction of HO ₂ concentration field at 39,600 seconds	190
8.25	DSAGA-PPM prediction of O ₃ concentration field at 39,600 seconds	190
8.26	Adaptive grid at 39,600 seconds reflecting clustering of nodes around the upwind point source and in and around the urban area.....	191
8.27	Refined grid SGA-PPM prediction of NO concentration field at 40,000 seconds	194
8.28	Refined grid SGA-PPM prediction of O ₃ concentration field at 40,000 seconds	194
A.1	Tangential Cartesian Coordinates	214
A.2	Spherical Coordinates.....	217

List of Symbols

Uppercase

A, B, and C	coordinate systems in E^3
A_b	base area of a three dimensional cell
D_l	molecular diffusivity of species l
E^3	the Euclidean space
E, F, G	mass fluxes in x , y , and z directions, respectively
$E_l^P(\vec{X}_p, t)$	mass emission rate of the point source located at $\vec{X}_p$
E_l^A	mass emission flux from an area source
$H(x, y, t)$	top of the modeling domain in Cartesian space
J	Jacobian of transformation
K, K^{mn}	symmetric turbulent diffusivity tensor, $m = 1, 2, 3$ and $n = 1, 2, 3$
K_{xx}, K_{yy}	horizontal turbulent diffusivity components
K_{zz}	vertical turbulent diffusivity
L	three-dimensional, semi-linear differential operator
M	number of interim steps
M_l	molecular weight of species l
N	number of species
N_k	number of cells used to compute the discrete Laplacian

$\vec{P}_i$	the position coordinates in physical space of the centers of the cells that are local to the grid node o ; $i = 1, \dots, 4$.
R_l	net generation of chemical species l by chemical reactions
R_s	stiffness ratio
$\Re$	Universal gas constant
S_l	rate of source addition for the chemical species l
S_l^p	rate of change of concentration due to point source
S_l^A	rate of change of concentration due to area source
T	temperature of air
V	cell volume in physical space
$\vec{V}(\vec{X}, t)$	wind field in $\Omega(t)$
V_l^g	pollutant deposition velocity
$\vec{X}$	position vector in $\Omega(t)$
$\vec{X}_1, \vec{X}_2$	physical locations in Cartesian coordinates

Lowercase

$a(\vec{X}, t), b(\vec{X}, t)$	coefficients associated with boundary conditions
c_l	concentration of l^{th} species
$c_l _{in}, c_l _{out}$	species concentrations at inflow and outflow boundaries, respectively
$c_l^H(\vec{X}, t), c_l^T(\vec{X}, t)$	species concentrations at the horizontal boundaries and the top surface of the modeled domain, respectively

e_1, e_2	volume weighting parameters
$g_{i\ j}$	the metric tensor for the coordinate system x^m
$g_l(\vec{X}, t)$	species specific boundary condition coefficient
$h(t)$	mixing layer height
$\vec{i}, \vec{j}, \vec{k}$	unit vectors along x, y, z directions, respectively
k_j	temperature dependent rate constant for j^{th} reaction
$\vec{n}$	the unit outward normal vector to bounding surface dS
$\vec{n}_H, \vec{n}_T$	unit normals to the horizontal boundary and the top surface of the modeled domain, respectively
p	pressure
q_l	mass mixing ratio for species l
r_j	the reaction rates of the m individual reactions, $j = 1, \dots, m$
t	time
u, v, w	velocity vector components along x, y, z directions, respectively
$\vec{u}_H, \vec{u}_T$	advective velocity in the horizontal plane and with respect to the top surface of the modeled domain, respectively
$\hat{u} _{boundary}$	velocity at the boundary
v^m	contravariant velocity components in x^m .
$\vec{v}_b$	velocity of an element of the bounding surface dS
$w(x, y, t)$	weight function

$w_{\min}$	minimum allowable weight function
$\mathbf{x}$	Cartesian coordinate system
x_t, y_t, z_t	tangential Cartesian coordinates
x^m	a coordinate system (x^1, x^2, x^3)
x^α	a coordinate system $(x^1, x^2, x^3, x^4 = t)$
x, y, z	Cartesian space vector components
<u>Greek</u>	
α_k, α_o	coefficients defining discrete Laplacian
α_{jl}, β_{jl}	the reactant and the product stoichiometry, respectively, in the reaction step j ; $j = 1, \dots, m$
β_m	interim step iteration count
ρ	density of air
γ	determinant of metric tensor
θ	solar zenith angle
λ_l	eigenvalues of the Jacobian Matrix $J_c = \frac{\partial R_l}{\partial c_j} ; l = 1, \dots, N; \quad j = 1, \dots, N .$
Ψ	any property (e.g., concentration)
$\Omega(t)$	time varying, spatial region
Ω^{cv}	moving control volume

$\partial \Omega(t)$	boundary surface of $\Omega(t)$
δt	a time period
ξ, η, ζ	general curvilinear coordinates
σ	direction normal to $\partial \Omega(t)$.
Δt	basic time step of atmospheric diffusion equation
$\Delta \vec{X}_{i,j}$	change in position coordinates of the node i, j
$\Delta \xi, \Delta \eta$	cell spacing in ξ and η directions, respectively
Δ^2	discrete approximation of Laplacian

Subscripts and Superscripts

adv	pertaining to advection
c	pertaining to chemistry
$diff$	pertaining to diffusion
e	exact or reference
g	grid time
i, j, k	array positions, values associated with grid cell
l	species index, $l = 1, \dots, N$
min, max	minimum and maximum values
$n, (n+1), n+1$	time level indicators
0	initial conditions

o	grid node o
p	values associated with cell side
s	pertaining to source terms
x, y, z, t	partial derivatives with respect to given Cartesian variable
ξ, η, ζ	partial derivatives with respect to given transformation space variable

Accents

$-$	denotes Reynolds averaged or cell averaged value
$\wedge$	denotes a transformed quantity
$\langle \rangle$	denotes an ensemble averaged quantity
$\sim$	denotes cell side sweep volume-averaged value

Abbreviations / Chemical Names

AQM	air quality model
CFL	Courant-Friedrichs-Levy number
CIE	Conservative Interpolation Equation
CO	carbon monoxide
CO ₂	carbon dioxide
DSAGA	Dynamic Solution Adaptive Grid Algorithm
HC	lumped hydrocarbons
HCHO	formaldehyde

HNO_3	nitric acid
H_2O	water
HO_2	hydroperoxyl radical
Mflops	million floating point operations per second
NO	nitric oxide
NO_2	nitrogen dioxide
NO_x	nitrogen oxides
OH	hydroxyl radical
O_3	ozone
<i>ppb</i>	parts per billion
PPM	Piecewise Parabolic Method
<i>ppmv</i>	parts per million by volume
rad	radians
RMS	root-mean-square
RO_2	alkylperoxy radical
SIERRA	Solver Independent, Efficient r-Refinement Algorithm
VOC	volatile organic compound

It is quite a three-pipe problem.

Sir Arthur Conan Doyle

1 Introduction

One of the major concerns facing the world currently is the impact of anthropogenic activities on the environment. Detrimental impacts on the environment resulting from acid deposition; increase in concentrations of greenhouse gases, fine particles, and tropospheric ozone (O_3); and depletion of stratospheric O_3 have been the focus of many research and regulatory activities. Over the next two decades, many far-reaching policy decisions will have to be made to address these impacts. Since these decisions will have a direct bearing on our health and lifestyle, it is imperative that research programs continue to provide, or improve, the scientific foundations for these decisions.

The complexity of atmospheric physical and chemical processes, in conjunction with the enormity of the atmosphere, makes it difficult to develop a relatively complete understanding of environmental impacts from relatively few measurements of atmospheric phenomena. The results of numerical simulations of the atmosphere play a key role in arriving at a more complete understanding of environmental impacts. These simulations utilize detailed numerical models in which atmospheric measurements are used as input data.

Numerical simulations conducted using air quality models (AQMs) are used to determine the impacts of air pollution on the environment. Specifically, the predictions of AQMs are used in establishing ambient air quality standards aimed at protecting public health with an adequate margin of safety. Additionally, AQM predictions are also used to develop effective emission reduction strategies focussed on attainment of ambient air quality standards. Since the predictions of AQMs have, and will continue to have, a direct bearing on public health and lifestyle, it is quite important that efforts continue to be made to improve the accuracy of these predictions.

The physical and chemical processes responsible for air pollution span a wide range of spatial scales. For example, there may be point sources, such as power plants, that are characterized by relatively small spatial scales compared to regional-scale pollutant plumes from such sources. Therefore, in order to accurately model transport and chemistry of air pollutants, an AQM has to be able to adequately resolve the pertinent spatial scales. This can be achieved by varying the physical grid node spacing in an AQM to provide resolution where needed.

One approach to achieving increased local solution resolution involves using embedded Cartesian grids, or static nested grids, such as those described by Odman and Russell [1] and Odman *et al.* [2]. This approach may be limited by: (1) the uncertainty in nested grid(s) placement(s) since pertinent locations may not be known *a-priori*, (2) the loss in solution accuracy resulting from grid boundary interface problems, and (3) the

inability to adjust to dynamic changes in solution resolution requirements. Another approach to achieving local solution resolution involves using dynamic adaptive grids. In principle, such grids would be continuous and would dynamically adjust to changing solution resolution requirements. Therefore, use of such grids would not be constrained by the limitations associated with use of nested grids.

Recently, ideas promoting use of dynamic adaptive grids in atmospheric modeling are gaining in popularity. Dietachmayer and Droegemeier [3] use a variational formulation of adaptive grid generation equations to compute solutions to test problems. Almgren *et al.* [4] have used a nested hierarchy of grids, with simultaneous refinement of grids in both space and time, to compute the release of hot gas into the atmosphere. Skamarock and Klemp [5] have used a hierarchical grid approach to model a compressible formulation of atmospheric flow equations. Recently, Tomlin *et al.* [6] have investigated the use of an adaptive unstructured grid method to obtain solutions of test problems of interest in air quality modeling.

A grid adaptation algorithm suitable for use in aerospace applications has been developed by Benson and McRae [7-9]. This algorithm, called the *Dynamic Solution Adaptive Grid Algorithm* (DSAGA), uses weight functions constructed from absolute values of gradients of solution variables, along with a *center-of-mass* scheme, to move grid nodes. These nodes are repositioned in a parametric space to reduce the potential for grid-line cross-over. After repositioning of grid nodes, the values of solution variables

are corrected. This correction is completed using an equation obtained by splitting the time-dependent terms from the steady terms in the general conservation laws for moving control volumes. DSAGA has been used to compute unsteady, three-dimensional, turbulent, viscous flows [10-12]. Additionally, Srivastava *et al.* [13] have used DSAGA to compute test cases of interest in air quality modeling. Recently, Laflin and McRae [14] and Laflin [15] have developed the *Solver-Independent, Efficient r-Refinement Algorithm* (SIERRA) that provides robust and computationally efficient implementation strategies for grid adaptation algorithms. SIERRA incorporates a solver-independent weight function and a solver-independent solution field redistribution procedure.

Developed in this work is a new dynamic adaptive grid algorithm that is suitable for use in AQMs. This algorithm was built on the foundation provided by the DSAGA algorithm of Benson and McRae [7-9]. However, the new algorithm extends the DSAGA framework by utilizing more accurate solution procedures. Specifically, the new algorithm uses the *Piecewise Parabolic Method* (PPM) developed by Collela *et al.* [16] for computing the advective fluxes and the fluxes resulting from grid movement. Moreover, the new algorithm incorporates the SIERRA weight function and solution field redistribution procedure. This new algorithm is called the *Dynamic Solution Adaptive Grid Algorithm - PPM* (DSAGA-PPM). The subsequent chapters describe the development of DSAGA-PPM and its use in model problems. A perspective on these chapters is provided below.

Using tensor calculus, the governing equations for the current Eulerian air quality models are derived in Chapter 2. In addition, the expressions of the associated initial and boundary conditions are specified. Also presented in this chapter are brief descriptions of the popular coordinate systems being used in air quality modeling. A cohesive treatment of the properties of some of the common coordinate systems, coordinate transformations, and the expressions of the governing equations in these common coordinates is presented in Appendix A.

In general, DSAGA-PPM includes procedures for advancing the solution field in time on a non-uniform grid, for moving the grid nodes to regions requiring solution refinement, and for conservatively redistributing the solution field over the resulting adapted grid. The mathematical-numerical formulation of these procedures is developed in Chapter 3.

In this work, idealized one- and two-dimensional test problems were chosen to examine the performance of DSAGA-PPM. For each of the test problems, results were obtained using DSAGA-PPM and the corresponding *Static Grid Algorithm - PPM* (SGA-PPM). These results are compared to reveal the advantages of using DSAGA-PPM. Note that SGA-PPM is obtained by deactivating the grid adaptation procedures in DSAGA-PPM. An SGA-PPM application utilizes a uniformly-spaced grid that is

identical to the starting uniformly-spaced grid used in the corresponding DSAGA-PPM application.

Presented in Chapter 4 are the SGA-PPM and DSAGA-PPM simulations of model test problems with advection. These simulations test the capability of DSAGA-PPM to provide accurate advection solutions. Developed in Chapter 5 is a weight function that is suitable for use in flows with reacting species. This weight function is used in the DSAGA-PPM simulations of model test problems with chemistry, described in Chapters 6, 7, and 8. The simulations in Chapters 6, 7, and 8 test the ability of DSAGA-PPM to provide the needed solution resolution in flows with reacting species and sources. Evaluated in Chapter 9 is the practicality of using DSAGA-PPM in AQMs by analyzing the computational performance it achieved in model test problems. Finally, a summary of this work and related conclusions are presented in Chapter 10.

2 Governing Equations and Boundary Conditions

Currently, atmospheric pollutant transport models are expressed in various coordinate systems. Considerations such as terrain-following capability, solution resolution, simplifications of mathematical and numerical procedures, etc. dictate selection of a particular coordinate system for use in an AQM. In general, implementing atmospheric pollutant transport governing equations in more than one coordinate system is a nontrivial task. However, such an implementation may be attractive for several reasons. First, it could reveal benefits of using various coordinate systems. Second it could facilitate linking of the resulting models to other existing models and databases. Therefore, an objective in this work was to investigate formulation of a generalized air pollution transport model that could be readily implemented in most of the common coordinate systems.

In order to characterize grid movement in an adaptive grid procedure, a complete description of the pertinent coordinate system is needed. Further, in light of the smaller spatial scales attainable with grid adaptation, the assumptions made during the formulation of the governing equations need a careful examination. Hence, another objective was to construct a rigorous formulation of the governing equations, embodying pertinent physics and mathematical rigor, so that a high resolution adaptive grid algorithm could be developed for use in AQMs.

Note that if the vectors in physical space are always expressed in Cartesian coordinates, then no derivatives of unit vectors are needed in expressions of these vectors in some other coordinate system. Therefore it is convenient to use Cartesian coordinates in a grid adaptation algorithm. The results using grid adaptation could be obtained in Cartesian coordinates and expressed in any other coordinate system of interest using relatively simple transformation relations. Description of grid adaptation in Cartesian coordinates is provided in Chapter 3.

The *atmospheric diffusion equation* [17] constitutes the system of governing equations for the current Eulerian air quality models (AQMs). Using the tools of tensor calculus, this equation is derived in Section 2.1. During this derivation, limitations of the atmospheric diffusion equation are also noted. In Section 2.2, expressions of the initial and boundary conditions associated with the atmospheric diffusion equation are specified. In Section 2.3, a brief discussion of the various coordinate systems being used in air quality modeling is presented. A cohesive treatment of the properties of some of the common coordinate systems, coordinate transformations, and the expressions of the atmospheric diffusion equation in these coordinates are presented in Appendix A.

2.1 Derivation of the Governing Equations

In an arbitrary, time varying, spatial region $\Omega(t)$, located in the Euclidean space E^3 , and bounded by $\partial\Omega(t)$, a spatial point is given by $\vec{X} \in \Omega(t)$. In $\Omega(t)$ the conservation of mass for each of N chemical species $c_l(\vec{X}, t)$, $l = 1, \dots, N$ is expressed as

$$\frac{\partial c_l}{\partial t} + \nabla \cdot (\vec{V} c_l) = \nabla \cdot (D_l \cdot \nabla c_l) + R_l(c_1, \dots, c_N, T) + S_l(\vec{X}, t); \quad l = 1, \dots, N. \quad (2.1)$$

In Equations (2.1), c_l is the mass concentration of pollutant l (i.e., mass of pollutant l per unit volume of air); $\vec{V}(\vec{X}, t)$ is the specified wind field; D_l is the molecular diffusivity of species l in air; R_l is the net generation of chemical species l by chemical reactions; T is the temperature of air; and S_l is the rate of source addition for the chemical species l .

In order for Equation (2.1) to contain the correct physics, $\vec{V}$ and T must satisfy the Navier-Stokes and energy equations, respectively. In addition closure must be achieved using an appropriate equation of state. Thus, in general, one must solve a system of coupled partial differential equations to properly account for the changes and interactions in c_l , $\vec{V}$, and T .

In general, pollutant concentrations are small (on the order of $\mu\text{g}/\text{m}^3$) and the presence of such pollutants does not affect the meteorology to any detectable extent. Hence, the species conservation equations, i.e., Equation (2.1), can be solved independently of the momentum (Navier-Stokes) and energy equations and $\vec{V}$ and T can be considered to be independent of c_i . Hereafter, the explicit dependence of R_i on T will not be indicated. In some circumstances, however, a change in c_i may affect meteorology. In such cases, meteorology may not be independent of the pollutant solution field. Jacobson [18], in his simulations of aerosol and gas pollution buildup in the Los Angeles basin, has reported that aerosol concentrations had a detectable impact on day and night temperatures. Such changes in temperatures were attributed to the complex absorption, re-emission, and attenuation of solar radiation by aerosol concentrations. Thus, in modeling elevated aerosol levels the assumption of decoupling of Navier-Stokes and energy equations may require careful consideration. For example, if the effects on meteorology from particle and gas interactions are expected to be small then the measured changes in velocities and temperatures may be included in solutions of species equations alone, as explained in Reynolds *et al.* [19]. If, however, such effects are not expected to be small, then the decoupling of equations may not be valid.

The atmospheric flow is turbulent and therefore contains eddies which cause turbulent diffusion of the species present. In general, the effect of this turbulent diffusion on a specie is much greater than that due to molecular diffusion. Therefore, molecular

diffusion in Equation (2.1) can, in general, be ignored. With this assumption, Equation (2.1) becomes

$$\frac{\partial c_l}{\partial t} + \nabla \cdot (\vec{V} c_l) = R_l(c_1, \dots, c_N, T) + S_l(\vec{X}, t); \quad l = 1, \dots, N. \quad (2.2)$$

At this point in our development, we digress briefly and present some expressions from tensor calculus. The use of tensor calculus is quite attractive in the study of atmospheric motion for several reasons. First, many manipulations of vector quantities can be performed rather easily using tensor calculus. Second, true tensors do appear in the study of fluid motion. Finally, the use of tensor analysis is necessary while using nonorthogonal coordinate systems. We shall use these tensor expressions in the derivation that follows.

The coordinate systems of interest are located in an Euclidean space E^3 . This is a curvature free space in which rectangular Cartesian coordinates can be introduced on a global scale. Let **A**, **B**, and **C** be three different coordinate systems defined and located in this Euclidean space. Each of these systems must span the Euclidean space. Properties of **C** can be derived either by using properties of **A** and transformation rules between **A** and **C**, or by using properties of **B** and transformation rules between **B** and **C**. However, in an Euclidean space all coordinate systems with removable singularities can be related to a global rectangular Cartesian coordinate system. Using this property and the chain rule of

calculus, it can be shown that certain fundamental elements of $\mathbf{C}$, such as the metric tensor, are independent of the transformation route followed. Thus, in the expressions given below, $\mathbf{x}$ represents a convenient selected coordinate system and $\hat{\mathbf{x}}$ a desired system. Unless stated otherwise, $\mathbf{x}$ represents a Cartesian coordinate system. The properties of $\hat{\mathbf{x}}$ are expressed in terms of those of $\mathbf{x}$.

The components of a contravariant vector (e.g., velocity vector) are transformed as

$$\hat{v}^k = \frac{\partial \hat{x}^k}{\partial x^m} v^m \quad (2.3)$$

The components of a second order covariant tensor (e.g., metric tensor) are transformed as

$$\hat{g}_{i\ j} = \frac{\partial x^k}{\partial \hat{x}^i} \frac{\partial x^m}{\partial \hat{x}^j} g_{k\ m} \quad (2.4)$$

The divergence of a vector (e.g., advective flux) can be expressed as

$$\nabla \cdot (c_l \vec{V}) = \frac{1}{\sqrt{\gamma}} \frac{\partial}{\partial x^m} (\sqrt{\gamma} c_l v^m) \quad (2.5)$$

In the expressions (2.3), (2.4), and (2.5), $\gamma = \det(g_{ij})$, g_{ij} is the metric tensor for the coordinate system x^m , and v^m are contravariant velocity components in x^m .

As shown in Defrise [20], using Equation (2.5), Equation (2.2) can be expressed for any coordinate system x^α ($x^1, x^2, x^3, x^4 = t$) as:

$$\frac{1}{\sqrt{\gamma}} \frac{\partial}{\partial x^\alpha} (\sqrt{\gamma} c_l v^\alpha) = R_l(c_1, \dots, c_N) + S_l(x^\alpha); \quad \alpha = 1, 2, 3, 4 \quad (2.6)$$

Note that $v^4 = \frac{dx^4}{dt} = \frac{dt}{dt} = 1$ in the above.

The atmospheric flow is turbulent so the velocities v^α can be expressed as $v^\alpha = \overline{v^\alpha} + v^{\alpha'}$ where $\overline{v^\alpha}$ and $v^{\alpha'}$ are the mean and random components, respectively. Since the velocities are temporally varying about a mean, the c_l resulting from the solution of Equation (2.6) must also vary similarly. Since the actual variation of $v^{\alpha'}$ about the mean

$\overline{v^\alpha}$ is not precisely obtainable by either calculation or experiment, it becomes necessary to seek a statistical description of turbulence, such as that provided by a probability density function. However, for a complex random process such as atmospheric turbulence, this is almost never possible. Hence, as explained in Seinfeld [21], one resorts to the next best approach and specifies: $c_i = \langle c_i \rangle + c_i'$, where $\langle c_i \rangle$ is the mean of an ensemble of infinite realizations and c_i' denotes a random component.

The atmospheric diffusion equation results after using the above expressions for v^α and c_i in Equation (2.6), performing an ensemble average of the resulting equation, and making the following assumptions: (1) the ergodic hypothesis holds for the ensemble averaging process, i.e., $\langle \Psi \rangle = \overline{\Psi}$ for any property Ψ (specifically, the ergodic hypothesis says that the mean of the ensemble of infinite realizations of any property is the same as the time average of that property for a single realization); (2) the flow field has stationary turbulence, i.e., $\overline{\Psi'} = 0$ for any property Ψ ; (3) the source function is deterministic for all practical purposes, i.e., $\langle S_i(x^\alpha) \rangle = S_i(x^\alpha)$; and (4) $\langle R_i(c_1, \dots, c_N) \rangle$ is approximately the same as $R_i(\langle c_1 \rangle, \dots, \langle c_N \rangle)$. As explained in Seinfeld [21], the approximation of $\langle R_i(c_1, \dots, c_N) \rangle$ to $R_i(\langle c_1 \rangle, \dots, \langle c_N \rangle)$ assumes that the effect of concentration fluctuations on the rates of reactions is negligible.

With the above assumptions, Equation (2.6) becomes

$$\frac{\partial}{\partial x^\alpha} \left(\sqrt{\gamma} \overline{c_l} \overline{v^\alpha} \right) + \frac{\partial}{\partial x^m} \left(\sqrt{\gamma} \overline{c_l'} \overline{v^m'} \right) = \sqrt{\gamma} R_l(\overline{c_1}, \dots, \overline{c_N}) + \sqrt{\gamma} S_l(x^\alpha) \quad (2.7)$$

where $\alpha = 1, 2, 3, 4$, and $m = 1, 2, 3$. Note $x^4 = t$, $\overline{v^4} = 1$, and $\overline{v^4'} = 0$.

A first order closure utilizing the K-theory (or the mixing-length theory) is adopted for modeling turbulent diffusion. Venkatram [22] has argued that the K-theory can be applied to those variables that are conserved along the path of a closed parcel or to variables whose material derivative is zero. He has shown that $\frac{dq_l}{dt}$ is zero, where

$q_l = \frac{c_l}{\rho}$ is the mass mixing ratio for species l . However, according to Venkatram $\frac{dc_l}{dt}$ is

not zero. Then, the application of K-theory to q_l results in

$$\overline{q_l' v^m'} = -K^{mn} \frac{\partial}{\partial x^n} (\overline{q_l}) ; m = 1, 2, 3 \text{ and } n = 1, 2, 3 \quad (2.8)$$

In the expressions above, ρ is the density of air, and K^{mn} is the symmetric turbulent diffusivity tensor.

Note that in introducing the K-theory for turbulent diffusion, the three unknowns $\overline{q_l' v^m'}$; $m = 1, 2, 3$ have been replaced with six unknown components of the symmetric turbulent diffusivity tensor K^{mn} . To achieve closure, the turbulent diffusivity tensor is assumed to be a diagonal tensor. This can happen because either the coordinate axes of the physical system described by Equation (2.7) are aligned with the principal axes of the turbulent diffusivity tensor or the off-diagonal terms of K^{mn} are not important. The diagonal diffusivity tensor K^{mn} and its ramifications are discussed in McRae et al. [17] and Toon et al. [23].

With the assumption of diagonal K^{mn} , Equation (2.8) becomes

$$\overline{q_l' v^m'} = -K^{mn} \frac{\partial}{\partial x^m} (\overline{q_l}) \quad (2.9)$$

Note that no summation is implied in the above representation. Thus, for example,

$$\overline{q_l' v^{1'}} = -K^{11} \frac{\partial}{\partial x^1} (\overline{q_l})$$

As shown in Van Dop [24],

$$\overline{c_l' v^m'} \approx \bar{\rho} \overline{q_l' v^m'} \quad (2.10)$$

Now using Equations (2.9) and (2.10),

$$\overline{c_l' v^m'} \approx \bar{\rho} \left[-K^{mm} \frac{\partial}{\partial x^m} (\bar{q}_l) \right] \quad (2.11)$$

Substitution of Equation (2.11) in Equation (2.7) yields:

$$\begin{aligned} \frac{\partial}{\partial x^\alpha} \left(\sqrt{\gamma} \bar{c}_l \bar{v}^\alpha \right) + \frac{\partial}{\partial x^m} \left[-\sqrt{\gamma} \bar{\rho} K^{mm} \frac{\partial}{\partial x^m} \left(\frac{\bar{c}_l}{\bar{\rho}} \right) \right] = \\ \sqrt{\gamma} R_l \left(\bar{c}_1, \dots, \bar{c}_N \right) + \sqrt{\gamma} S_l \left(x^\alpha \right) \end{aligned} \quad (2.12)$$

In air pollution modeling applications, the system of equations (2.12) is referred to as *the atmospheric diffusion equation* [17]. This system constitutes the governing equations for an AQM.

In vector notation, the system of Equations (2.12), with air density ρ being constant in space, can be expressed as:

$$\frac{\partial \bar{c}_l}{\partial t} + \nabla \cdot (\vec{V} \bar{c}_l) = \nabla \cdot (K \nabla \bar{c}_l) + R_l(\bar{c}_1, \dots, \bar{c}_N) + S_l(\vec{X}, t); \quad l = 1, \dots, N. \quad (2.13)$$

In this work, we focus on the form of the atmospheric diffusion equation expressed by Equation (2.13). Also, for convenience and clarity, henceforth we dispense with the overbars on the concentrations.

It is worth noting that Equation (2.13) is not an exact representation of the fundamental conservation law governing the dynamic behavior of pollutants in the atmosphere. Instead because of the numerous assumptions made in arriving at Equation (2.13), it is a *model* of the pollutant dynamics in the atmosphere. The studies conducted by Lamb [25], Lamb and Seinfeld [26], and Reynolds *et al.* [19] provide insights into the validity of this model. However, it is generally accepted that Equation (2.13) is a correct description of transport, mixing, and chemical reaction processes being experienced by pollutant species in the atmosphere. Possibly the major factor influencing the validity of this equation is the first order closure of turbulence; however, as long as the eddy diffusivity functions used in the model provide correct turbulent mixing, the approximation should be considered valid.

2.2 Initial and Boundary Conditions

The atmospheric diffusion equation is solved with specified initial and boundary conditions. For Equation (2.13), the initial conditions, $c_l(\vec{X}, 0)$, are given by

$$c_l(\vec{X}, 0) = c_l^0(\vec{X}), \quad l = 1, \dots, N; \quad \vec{X} \in \Omega(t=0); \quad (2.14)$$

where $c_l^0(\vec{X})$ are the specified initial concentration data. Boundary conditions are statements of mass continuity across the boundary surface $\partial\Omega(t)$. For the system (2.13), most practical situations are described by the inhomogeneous, mixed Neumann and Dirichlet boundary conditions given by

$$a(\vec{X}, t) c_l + b(\vec{X}, t) \frac{\partial c_l}{\partial \sigma} = g_l(\vec{X}, t), \quad l = 1, \dots, N; \quad (2.15)$$

where σ indicates the direction normal to $\partial\Omega(t)$. Expressions of (2.15) for specific cases are provided below.

The horizontal boundary conditions are given by

$$\left[\vec{u}_H c_l - K \nabla c_l \right] \cdot \vec{n}_H = \vec{u}_H c_l^H \cdot \vec{n}_H \quad ; \quad \vec{u}_H \cdot \vec{n}_H \leq 0 \quad (2.16)$$

$$-K \nabla c_l \cdot \vec{n}_H = 0 \quad ; \quad \vec{u}_H \cdot \vec{n}_H > 0 \quad (2.17)$$

where $\vec{n}_H$ is the unit normal to the horizontal boundary, $c_l^H(\vec{X}, t)$ are the species concentrations at the horizontal boundary, and $\vec{u}_H$ is the advective velocity in the horizontal plane. Equation (2.17) reflects that turbulent transport in outflow is zero, an approximation that is usually satisfied due to horizontal advection being the dominant physical transport process.

The boundary conditions at the top of the modeling domain $z = H(x, y, t)$ are similar to those for the horizontal plane. These are given by:

$$\left[\vec{u}_T c_l - K \nabla c_l \right] \cdot \vec{n}_T = \vec{u}_T c_l^T \cdot \vec{n}_T \quad ; \quad \vec{u}_T \cdot \vec{n}_T \leq 0 \quad (2.16)$$

$$-K \nabla c_l \cdot \vec{n}_T = 0 \quad ; \quad \vec{u}_T \cdot \vec{n}_T > 0 \quad (2.17)$$

where $\vec{n}_T$ is the unit normal to the top surface of the modeling domain, $c_l^T(\vec{X}, t)$ are the species concentrations at the top boundary, and $\vec{u}_T$ is the advective velocity with respect to the top surface. This velocity is given by

$$\vec{u}_T = u \vec{i} + v \vec{j} + \left(w - \frac{\partial H}{\partial t} \right) \vec{k} \quad (2.18)$$

Note that in the above equation the term $\frac{\partial H}{\partial t}$ describes the changes in the top surface with time.

The ground level boundary conditions are given by:

$$\left[V_l^g c_l - K_{zz} \frac{\partial c_l}{\partial z} \right] = E_l(\vec{X}, t) \quad (2.19)$$

where E_l is the mass flux of species c_l , K_{zz} the vertical diffusivity, and V_l^g the pollutant deposition velocity.

2.3 Coordinate Systems

Air quality models use a variety of vertical and horizontal coordinate systems. Considerations such as terrain - following capability, solution resolution, simplifications of mathematical and numerical procedures, etc. dictate the selection of a particular

coordinate system for use in an AQM. Since the large-scale motions of the atmosphere are quasi-horizontal with respect to the earth's surface, horizontal and vertical coordinates are considered separately. Provided below are descriptions of the popular coordinate systems used in air quality modeling.

2.3.1 Horizontal Coordinate Systems

Accuracy of results of an air quality simulation will be limited by the quality of the input data. Consequently, the horizontal coordinate system used should be closely related to that used in the observation database (concentration, deposition, land use). The following are the popular horizontal coordinate systems used in AQMs.

Tangential Cartesian Coordinates

The tangential Cartesian coordinates can be established on the surface of the earth, as shown in Figure 2.1. The origin of this coordinate system is located at a position O on the earth's surface. The x_t and y_t axes form a plane tangent to the earth's surface and point towards the east and the north, respectively. The z_t axis points in the radial direction making the coordinate system a right-handed one. The use of this coordinate system is limited to small-scale local studies because of curvature associated with the spherical geometry of the earth.

Figure 2.1. Tangential Cartesian Coordinates

Spherical Coordinates

The geometry of the earth is nearly spherical with slight ellipticity. In common practice, the earth is taken to be a sphere with associated spherical coordinates. As discussed in Haltiner and Williams [27], use of these coordinates in global models of atmospheric transport of chemical species has gained acceptance. These coordinates (r, λ, ϕ) are shown in Figure 2.2.

Figure 2.2. Spherical Coordinates

Map Projections

It is useful to project the surface of the earth onto a plane for analysis and visualization. The most popular projections are Polar Stereographic at high latitudes, Lambert Conformal at mid-latitudes, and Universal Transverse Mercator (UTM) near the equator. When doing a projection, new horizontal coordinates are set up on the projected plane and the governing equations and associated initial and boundary conditions are transformed accordingly. A good description of map projections is provided in Haltiner and Williams [27].

2.3.2 Vertical Coordinate Systems

The vertical structure of the atmosphere suggests that it might be useful to replace the geometrical vertical coordinate by one of the flow variables and, thus, use a generalized vertical coordinate in meteorological modeling. This generalization of the vertical structure of the atmosphere can provide simplifications in modeling procedures and allow terrain following. Several vertical coordinates in use have been discussed in Kasahara [28]. Since a meteorological model provides the meteorological inputs (boundary layer characteristics, cloud structure, wind patterns, etc.) for an AQM, it is important that an AQM have a coordinate system that is compatible with the meteorological model. Otherwise errors in meteorological input data for AQM will result from transforming this data from the coordinate system of the meteorological model to that of the AQM. Some popular vertical coordinates are summarized in Table 2.1 given below.

2.3.3 Coordinate Transformations

In air pollution modeling, the atmospheric diffusion equation and the associated initial and boundary conditions are appropriately transformed from a reference system (e.g., Cartesian) to the desired coordinate system selected to accommodate input data and resolution of meteorological (boundary layer, cloud) and geographical (topography) features. Discussions of the various transformations used can be found in Kasahara [28],

Toon *et al.* [23], and McRae *et al.* [17]. In this work, a cohesive treatment of some of the coordinate systems used in AQMs is provided in Appendix A.

Table 2-1. Some Generalized Vertical Coordinates

Vertical Coordinate	Reference	Remarks
Altitude	Toon (1988)	
Pressure	Toon (1988)	
Sigma-p: $\sigma_p = \frac{p - p_T}{p_s - p_T}$	Chang (1990)	$p_s(x, y)$ is surface pressure, p_T is top layer pressure
Sigma: $\sigma = \frac{p_0 - p_T}{p_s - p_T}$	Grell (1993)	$p_0(z)$ is reference pressure, $\rho_0(z)$ is reference air density
Sigma-z: $\sigma_z = \frac{z - h}{H - h}$	McRae (1982)	$h(x, y)$ is lower surface, $H(x, y, t)$ is height of upper surface
Scaled height: $\bar{z} = \frac{H(z - z_s)}{H - z_s}$	Gal-Chen (1975)	$z_s(x, y)$ is the topography

3 Elements of DSAGA-PPM

DSAGA-PPM includes the finite volume procedures for advancing the governing system (i.e., Equation (2.13) and the associated initial and boundary conditions) in time using a non-uniform grid, for moving the grid nodes to region(s) requiring solution refinement, and for conservatively redistributing the solution field over the resulting adapted grid. These procedures are described below.

3.1 Time Advancement of the Governing System

In the AQMs based on finite volume methods, the governing system, with transformed vertical coordinate, is usually advanced in time on static Cartesian grids with uniform spacing in the horizontal plane and non-uniform spacing in the vertical direction. However, in general, an adapted grid will not be uniform. Therefore in DSAGA-PPM, the solutions of the governing system need to be obtained on non-uniform grids. To facilitate this, a coordinate transformation is applied to the governing system to move from the physical domain expressed in Cartesian coordinates (x, y, z) to a computational domain expressed in general curvilinear coordinates (ξ, η, ζ) . The grid in these general curvilinear coordinates is assumed to be uniform and unit-spaced. This procedure allows for solution of the governing system to be obtained on arbitrary physical grids (i.e., grids that are curved in space and are aligned to shapes of existing solution features).

In physical space, Equation (2.13) can be expressed in Cartesian coordinates as

$$\frac{\partial c_l}{\partial t} + \frac{\partial E_l}{\partial x} + \frac{\partial F_l}{\partial y} + \frac{\partial G_l}{\partial z} = R_l + S_l, \quad l = 1, \dots, N; \quad (3.1)$$

where:

$$E_l = c_l u - K_{xx} \frac{\partial c_l}{\partial x} - K_{xy} \frac{\partial c_l}{\partial y} - K_{xz} \frac{\partial c_l}{\partial z}, \quad l = 1, \dots, N; \quad (3.2)$$

$$F_l = c_l v - K_{yx} \frac{\partial c_l}{\partial x} - K_{yy} \frac{\partial c_l}{\partial y} - K_{yz} \frac{\partial c_l}{\partial z}, \quad l = 1, \dots, N; \quad (3.3)$$

and

$$G_l = c_l w - K_{zx} \frac{\partial c_l}{\partial x} - K_{zy} \frac{\partial c_l}{\partial y} - K_{zz} \frac{\partial c_l}{\partial z}, \quad l = 1, \dots, N. \quad (3.4)$$

With the assumption of diagonal K (see Section 2.1), the above expressions reduce to

$$E_l = c_l u - K_{xx} \frac{\partial c_l}{\partial x}, \quad l = 1, \dots, N; \quad (3.5)$$

$$F_l = c_l v - K_{yy} \frac{\partial c_l}{\partial y}, \quad l = 1, \dots, N; \quad (3.6)$$

and

$$G_l = c_l w - K_{zz} \frac{\partial c_l}{\partial z}, \quad l = 1, \dots, N. \quad (3.7)$$

The mapping between the physical space expressed in Cartesian coordinates and the computational space expressed in general curvilinear coordinates (ξ, η, ζ) is given by

$$\xi = \xi(x, y, z), \quad \eta = \eta(x, y, z), \quad \zeta = \zeta(x, y, z). \quad (3.8)$$

As discussed in Roache [29], in general, the most accurate numerical results are obtained if numerical differencing is based on the conservative form of the governing equations. Methods for manipulating partial differential equations that preserve conservative properties are described in Anderson *et al.* [30], Oberkampf [31], and Vinokur [32]. Using these methods, it is possible to arrive at the conservative form of the transformed atmospheric diffusion equation given by

$$\frac{\partial \hat{c}_l}{\partial t} + \frac{\partial \hat{E}_l}{\partial \xi} + \frac{\partial \hat{F}_l}{\partial \eta} + \frac{\partial \hat{G}_l}{\partial \zeta} = \hat{R}_l + \hat{S}_l, \quad l = 1, \dots, N; \quad (3.9)$$

where:

$$\hat{c}_l = c_l / J, \quad l = 1, \dots, N; \quad (3.10)$$

$$\hat{E}_l = (\xi_x E + \xi_y F + \xi_z G) / J, \quad l = 1, \dots, N; \quad (3.11)$$

$$\hat{F}_l = (\eta_x E + \eta_y G + \eta_z G) / J, \quad l = 1, \dots, N; \quad (3.12)$$

$$\hat{G}_l = (\zeta_x E + \zeta_y F + \zeta_z G) / J, \quad l = 1, \dots, N; \quad (3.13)$$

$$\hat{R}_l = R_l(c_1, \dots, c_N) / J, \quad l = 1, \dots, N; \quad (3.14)$$

and

$$\hat{S}_l = S_l(\vec{X}, t) / J, \quad l = 1, \dots, N. \quad (3.15)$$

The expressions for the Jacobian J and the metrics of the transformation are available in Anderson *et al.* [30] and are also provided in Appendix B.

In general, Equation (3.9) represents a system of stiff partial differential equations since the time scales associated with typical chemical transformations are far smaller than those associated with transport due to advection and turbulent diffusion. Therefore,

decoupling transport and chemistry and solving for these processes in sequential steps results in more efficient computation of transport. As discussed in [17], Equation (3.1) (or Equation (3.9)) is operator - (or time -) split [33] to compute transport, chemistry, and source processes in sequential steps. The splitting of Equation (3.9) is not unique and differs between AQMs. One possible splitting sequence is

$$c_l(t + \Delta t) = L_{c_l} L_{s_l} L_{diff} L_{\zeta,adv} L_{\eta,adv} L_{\xi,adv} c_l(t), \quad l = 1, \dots, N; \quad (3.16)$$

$$c_l(t + 2\Delta t) = L_{\xi,adv} L_{\eta,adv} L_{\zeta,adv} L_{diff} L_{s_l} L_{c_l} c_l(t + \Delta t), \quad l = 1, \dots, N; \quad (3.17)$$

where:

$$L_{\xi,adv} = J \frac{\partial}{\partial \xi} \left(\frac{\xi_x(-u) + \xi_y(-v) + \xi_z(-w)}{J} \right); \quad (3.18)$$

$$L_{\eta,adv} = J \frac{\partial}{\partial \eta} \left(\frac{\eta_x(-u) + \eta_y(-v) + \eta_z(-w)}{J} \right); \quad (3.19)$$

$$L_{\zeta,adv} = J \frac{\partial}{\partial \zeta} \left(\frac{\zeta_x(-u) + \zeta_y(-v) + \zeta_z(-w)}{J} \right); \quad (3.20)$$

$$\begin{aligned}
L_{diff} = & J \frac{\partial}{\partial \xi} \left(\frac{\xi_x (K_{xx} \frac{\partial c_l}{\partial x}) + \xi_y (K_{yy} \frac{\partial c_l}{\partial y}) + \xi_z (K_{zz} \frac{\partial c_l}{\partial z})}{J} \right) + \\
& J \frac{\partial}{\partial \eta} \left(\frac{\eta_x (K_{xx} \frac{\partial c_l}{\partial x}) + \eta_y (K_{yy} \frac{\partial c_l}{\partial y}) + \eta_z (K_{zz} \frac{\partial c_l}{\partial z})}{J} \right) + \\
& J \frac{\partial}{\partial \zeta} \left(\frac{\zeta_x (K_{xx} \frac{\partial c_l}{\partial x}) + \zeta_y (K_{yy} \frac{\partial c_l}{\partial y}) + \zeta_z (K_{zz} \frac{\partial c_l}{\partial z})}{J} \right)
\end{aligned} \tag{3.21}$$

$$L_{s_l} = S_l(\bar{X}, t), \quad l = 1, \dots, N; \tag{3.22}$$

and

$$L_{c_l} = R_l(c_1, \dots, c_N), \quad l = 1, \dots, N. \tag{3.23}$$

The splitting defined by the relations (3.16) and (3.17) is symmetric and, as explained in [17], is second order accurate. Note that the relations (3.16) and (3.17) indicate that the transport processes in the atmospheric diffusion equation are decomposed into one-dimensional advection and three-dimensional diffusion problems.

With this splitting, high resolution advection solvers, developed for one-dimensional Euler equations, can be used to compute time-split three-dimensional advection.

The remainder of this chapter develops application of DSAGA-PPM to the atmospheric diffusion equation in two-dimensions only. Application of this algorithm to the atmospheric diffusion equation in three-dimensions be a logical extension of the procedures presented in the following sections.

3.2 Treatment of Two - Dimensional Advection

The finite volume representation of the advection component of (3.9), with splitting (3.16) and (3.17), can be used to compute two-dimensional advection. Accordingly, the advection solution at $t + \Delta t$ is given by

$$(\bar{c}_l^{(n+1)} V)_{i,j} - (\bar{c}_l^n V)_{i,j} + \Delta t_{adv} (\hat{E}_l^{adv} \Big|_{i+1/2,j} - \hat{E}_l^{adv} \Big|_{i-1/2,j})^n = 0, \quad l = 1, \dots, N; \quad (3.24)$$

$$(\bar{c}_l^{n+1} V)_{i,j} - (\bar{c}_l^{(n+1)} V)_{i,j} + \Delta t_{adv} (\hat{F}_l^{adv} \Big|_{i,j+1/2} - \hat{F}_l^{adv} \Big|_{i,j-1/2})^{(n+1)} = 0, \quad l = 1, \dots, N; \quad (3.25)$$

with

$$\hat{E}_l^{adv} \Big|_{i+1/2,j} = \left(\frac{\xi_x}{J} u c_l + \frac{\xi_y}{J} v c_l \right) \Big|_{i+1/2,j} ; \quad (3.26)$$

and

$$\hat{F}_l^{adv} \Big|_{i,j+1/2} = \left(\frac{\eta_x}{J} u c_l + \frac{\eta_y}{J} v c_l \right) \Big|_{i,j+1/2} . \quad (3.27)$$

In Equations (3.24) through (3.27), $\bar{c}_l$ is the average concentration of chemical species l in cell volume $V_{i,j}$; $\hat{E}_l^{adv}$ and $\hat{F}_l^{adv}$ are the net mass effluxes due to advection at cell sides $i \pm 1/2, j$ and $i, j \pm 1/2$, respectively; and $(n+1)$ is an intermediate time level between the levels n and $n+1$. Note that the concentrations appearing in Equations (3.26) and (3.27) have to be evaluated at the appropriate cell sides. For example, c_l appearing in Equation (3.26) has to be evaluated at the cell side $i+1/2, j$. In a finite volume formulation, the Jacobian J is simply the ratio of the volume of the computational cell to the volume, V , of the corresponding physical cell. In the computational domain, the grid is chosen to be uniform and unit-spaced for convenience. Consequently, the volume of each computational cell is unity and the Jacobian for each cell becomes $\frac{1}{V}$. Note that in two-dimensions, $V_{i,j}$, is simply the area of the cell i, j .

In air quality simulations, mass conservation of species, monotonicity of the solution fields, and high order of accuracy need to be maintained during numerical computations. The *Piecewise Parabolic Method* (PPM) - a numerical scheme for computing advection - is monotonic, is third order accurate for variable grid spacing, and has excellent mass conservation property [34]. Therefore, a time split version of the PPM scheme is used to compute the fluxes $\hat{E}_l^{adv}$ and $\hat{F}_l^{adv}$ required in (3.24) and (3.25). The PPM scheme, developed for modeling fluid flows with strong shocks and discontinuities, is a higher-order extension of Godunov's method and uses a parabola as the basic interpolation function in a finite volume cell [16]. A description of the PPM scheme is presented in Appendix C.

In any two- or three-dimensional finite volume scheme, numerical sources may result from errors in differencing of the metrics. As discussed in Hindman [35], such sources could affect solution quality in a significant way. To determine the presence of such sources, the *uniform flow test* has been used by Pulliam and Steger, Thomas and Lombard, and others [36-38]. In this test, each of the dependent variables is initialized with a constant value over the entire grid and then the solution is advanced through one time step with the boundary values fixed. If the numerics are consistently formulated, then the solution after the end of the time step must equal the initial conditions. As discussed in Gielda and McRae [39], an analytical analysis, in which uniform flow conditions are applied to the discretized difference equations, yields any numerical

sources resulting from the difference algorithm. Subsequently, these sources are eliminated by adding terms equal to negative of numerical source terms to the difference equations. In DSAGA-PPM, this procedure was used to eliminate the advective numerical sources, as described below.

Equation (3.24) is rewritten to yield

$$\bar{c}_l^{(n+1)}\Big|_{i,j} = \bar{c}_l^n\Big|_{i,j} - \frac{\Delta t_{adv}}{V_{i,j}}(\hat{E}_l^{adv}\Big|_{i+1/2,j} - \hat{E}_l^{adv}\Big|_{i-1/2,j})^n, \quad l = 1, \dots, N. \quad (3.28)$$

In the above equations, $\hat{E}_l^{adv}\Big|_{i+1/2,j}$ and $\hat{E}_l^{adv}\Big|_{i-1/2,j}$ are the advective mass fluxes at the cell edges $i + 1/2, j$ and $i - 1/2, j$, respectively. As described before, in DSAGA-PPM these fluxes are computed using the PPM scheme. For a flow with uniform (or constant) dependent variables $\bar{c}_l$, the PPM scheme reduces to the first order upwind scheme since each cell now has constant (instead of parabolic) profiles of the dependent variables.

Expanding Equation (3.28) for the uniform dependent variable flow,

$$\bar{c}_l^{(n+1)} \Big|_{i,j} = \bar{c}_l^n \Big|_{i,j} \quad (3.29)$$

$$- \frac{\Delta t_{adv} \cdot \bar{c}_l^n \Big|_{i,j}}{V_{i,j}} \left[\left(\frac{\xi_x}{J} u + \frac{\xi_y}{J} v \right)_{i+1/2,j} - \left(\frac{\xi_x}{J} u + \frac{\xi_y}{J} v \right)_{i-1/2,j} \right]^n, \quad l = 1, \dots, N.$$

Upon inspection of the above equations for a constant dependent variable flow, the numerical source term is identified to be

$$- \frac{\Delta t_{adv} \cdot \bar{c}_l^n \Big|_{i,j}}{V_{i,j}} \left[\left(\frac{\xi_x}{J} u + \frac{\xi_y}{J} v \right)_{i+1/2,j} - \left(\frac{\xi_x}{J} u + \frac{\xi_y}{J} v \right)_{i-1/2,j} \right]^n, \quad l = 1, \dots, N. \quad (3.30)$$

Subsequently the negative of this term is added to (3.24) to eliminate numerical sources.

A similar source correction is applied to (3.25).

The time step for advection is calculated as follows. Considering Equation (3.9), one dimensional advection is expressed as:

$$\frac{\partial \hat{c}_l}{\partial t} + \frac{\partial \hat{u}}{\partial \xi} \hat{c}_l = 0 \quad ; \quad \hat{c}_l = \frac{c_l}{J} \quad , \quad \hat{u} = u \xi_x + v \xi_y. \quad (3.31)$$

As shown in Anderson *et al.* [30], for the explicit upwind method being used to solve the above equation, stability consideration requires that

$$\frac{(\hat{u}\Delta t_{adv})|_{i+1/2,j}}{\Delta\xi} \leq 1. \quad (3.32)$$

Rearranging the above relation,

$$\Delta t_{adv}|_{i+1/2,j} \leq \frac{\Delta\xi}{\hat{u}_{i+1/2,j}} ; \Delta\xi = 1. \quad (3.33)$$

Then,

$$\Delta t_{adv}|_{i+1/2,j} \leq \frac{1}{\left[\left(\frac{\xi_x}{J} u + \frac{\xi_y}{J} v \right) J \right]_{i+1/2,j}}. \quad (3.34)$$

The metrics $\frac{\xi_x}{J}|_{i+1/2,j}$ and $\frac{\xi_y}{J}|_{i+1/2,j}$ are obtained using the relations given in Appendix B.

The jacobian $J|_{i+1/2,j}$ is computed using

$$J|_{i+1/2,j} = \frac{1}{\left(x_\xi y_\eta - y_\xi x_\eta\right)|_{i+1/2,j}} = \frac{1}{\left(\frac{\xi_x}{J} \frac{\eta_y}{J} - \frac{\xi_y}{J} \frac{\eta_x}{J}\right)|_{i+1/2,j}}; \quad (3.35)$$

where

$$\frac{\eta_x}{J}\bigg|_{i+1/2,j} = 0.25 \cdot \left(\frac{\eta_x}{J}\bigg|_{i,j+1/2} + \frac{\eta_x}{J}\bigg|_{i,j-1/2} + \frac{\eta_x}{J}\bigg|_{i+1,j+1/2} + \frac{\eta_x}{J}\bigg|_{i+1,j-1/2} \right); \quad (3.36)$$

and

$$\frac{\eta_y}{J}\bigg|_{i+1/2,j} = 0.25 \cdot \left(\frac{\eta_y}{J}\bigg|_{i,j+1/2} + \frac{\eta_y}{J}\bigg|_{i,j-1/2} + \frac{\eta_y}{J}\bigg|_{i+1,j+1/2} + \frac{\eta_y}{J}\bigg|_{i+1,j-1/2} \right). \quad (3.37)$$

Then using Equation (3.34), the time step for each of the cell edges $i \pm 1/2, j$ is obtained. In a similar fashion, the time step for each of the cell edges $i, j \pm 1/2$ is obtained. Finally,

$$\Delta t_{adv} = \min(\Delta t_{adv}|_{i+1/2,j}, \Delta t_{adv}|_{i,j+1/2}) \quad \forall i, j. \quad (3.38)$$

The advection time step obtained using (3.38) is used in Equations (3.24) and (3.25), corrected for numerical sources, to compute advection in two dimensions.

3.3 Treatment of Two - Dimensional Turbulent Diffusion

The finite volume representation of the turbulent diffusion component of (3.9), with splitting (3.16) and (3.17), can be used to compute two-dimensional turbulent diffusion. Accordingly, the turbulent diffusion solution at $t + \Delta t$ is given by

$$(\bar{c}_l^{n+1} V)_{i,j} - (\bar{c}_l^n V)_{i,j} + \Delta t_{diff} (\hat{E}_l^{diff} \Big|_{i+1/2,j} - \hat{E}_l^{diff} \Big|_{i-1/2,j} + \hat{F}_l^{diff} \Big|_{i,j+1/2} - \hat{F}_l^{diff} \Big|_{i,j-1/2})^n = 0, \quad l = 1, \dots, N; \quad (3.39)$$

with

$$\hat{E}_l^{diff} \Big|_{i+1/2,j} = \left(-\frac{\xi_x}{J} K_{xx} \frac{\partial c_l}{\partial x} - \frac{\xi_y}{J} K_{yy} \frac{\partial c_l}{\partial y} \right)_{i+1/2,j}; \quad (3.40)$$

and

$$\hat{F}_l^{diff} \Big|_{i,j+1/2} = \left(-\frac{\eta_x}{J} K_{xx} \frac{\partial c_l}{\partial x} - \frac{\eta_y}{J} K_{yy} \frac{\partial c_l}{\partial y} \right)_{i,j+1/2}. \quad (3.41)$$

In Equations (3.39) through (3.31), $\bar{c}_l$ is the average concentration of chemical species l in cell volume $V_{i,j}$; $\hat{E}_l^{diff}$ and $\hat{F}_l^{diff}$ are the net mass effluxes due to turbulent diffusion at cell sides $i \pm 1/2, j$ and $i, j \pm 1/2$ respectively; and $(n+1)$ is an intermediate time level between the levels n and $n+1$.

In order to use Equation (3.39), the expressions for $\frac{\partial c_l}{\partial x}$ and $\frac{\partial c_l}{\partial y}$ are needed at the cell sides $i \pm 1/2, j$ and $i, j \pm 1/2$. The quantity $\left. \frac{\partial c_l}{\partial x} \right|_{i+1/2,j}$ can be computed as follows.

$$\left. \frac{\partial c_l}{\partial x} \right|_{i+1/2,j} = \frac{\partial c_l}{\partial \xi} \xi_x \bigg|_{i+1/2,j} + \frac{\partial c_l}{\partial \eta} \eta_x \bigg|_{i+1/2,j}. \quad (3.42)$$

Now

$$\left. \frac{\partial c_l}{\partial \xi} \right|_{i+1/2,j} = c_l|_{i+1,j} - c_l|_{i,j}; \quad (3.43)$$

$$\begin{aligned}
\left. \frac{\partial c_l}{\partial \eta} \right|_{i+1/2,j} &= 0.5 \cdot \left(\left. \frac{\partial c_l}{\partial \eta} \right|_{i,j} + \left. \frac{\partial c_l}{\partial \eta} \right|_{i+1,j} \right) \\
&= 0.5 \cdot \left(\frac{c_l|_{i,j+1} - c_l|_{i,j-1}}{2} + \frac{c_l|_{i+1,j+1} - c_l|_{i+1,j-1}}{2} \right) ;
\end{aligned} \tag{3.44}$$

$$\left. \xi_x \right|_{i+1/2,j} = \left. \frac{\xi_x}{J} \right|_{i+1/2,j} \cdot J|_{i+1/2,j} ; \text{ and} \tag{3.45}$$

$$\left. \eta_x \right|_{i+1/2,j} = \left. \frac{\eta_x}{J} \right|_{i+1/2,j} \cdot J|_{i+1/2,j} . \tag{3.46}$$

Note that $\left. \frac{\xi_x}{J} \right|_{i+1/2,j}$ can be computed as discussed in Appendix B. The quantities

$\left. \frac{\eta_x}{J} \right|_{i+1/2,j}$ and $J|_{i+1/2,j}$ can be computed using Equations (3.36) and (3.35), respectively,

along with the relations given in Appendix B. Thus the expressions (3.42) - (3.46) can be

used to compute $\left. \frac{\partial c_l}{\partial x} \right|_{i+1/2,j}$. Similar procedures can be used to compute $\left. \frac{\partial c_l}{\partial x} \right|_{i-1/2,j}$,

$\left. \frac{\partial c_l}{\partial y} \right|_{i\pm 1/2,j}$, $\left. \frac{\partial c_l}{\partial x} \right|_{i,j\pm 1/2}$, and $\left. \frac{\partial c_l}{\partial y} \right|_{i,j\pm 1/2}$. Finally, with specified K_{xx} and K_{yy} , Equation

(3.39) can be used to compute the solution for turbulent diffusion.

The time step used in Equation (3.39) is calculated as follows. Considering Equation (3.9), one-dimensional diffusion is expressed as:

$$\frac{\partial \frac{c_l}{J}}{\partial t} + \frac{\partial}{\partial \xi} \left(-\frac{\xi_x}{J} K_{xx} \frac{\partial c_l}{\partial x} - \frac{\xi_y}{J} K_{yy} \frac{\partial c_l}{\partial y} \right) = 0. \quad (3.47)$$

Since the grid does not move during the diffusion step, $J \neq J(t)$. Further $x = x(\xi, \eta)$ and $y = y(\xi, \eta)$. Consequently, Equation (3.47) becomes

$$\frac{1}{J} \frac{\partial c_l}{\partial t} = \frac{\partial}{\partial \xi} \left(K_{\xi \xi} \frac{\partial c_l}{\partial \xi} + K_{\xi \eta} \frac{\partial c_l}{\partial \eta} \right); \quad (3.48)$$

where

$$K_{\xi \xi} = \frac{\xi_x}{J} K_{xx} \xi_x + \frac{\xi_y}{J} K_{yy} \xi_y; \quad (3.49)$$

and

$$K_{\xi \eta} = \frac{\xi_x}{J} K_{xx} \eta_x + \frac{\xi_y}{J} K_{yy} \eta_y. \quad (3.50)$$

Assume that locally at a cell edge, $K_{\xi\xi}$ and $K_{\xi\eta}$ are constant. Then

$$K_{\xi\xi} \neq K_{\xi\xi}(\xi) \ ; \ K_{\xi\eta} \neq K_{\xi\eta}(\xi) \quad (3.51)$$

Further assume that locally at a cell edge, for ξ -direction diffusion, $\frac{\partial c_l}{\partial \eta}$ is constant.

Then, locally at a cell edge, Equation (3.48) simplifies to

$$\frac{1}{J} \frac{\partial c_l}{\partial t} = K_{\xi\xi} \frac{\partial}{\partial \xi} \left(\frac{\partial c_l}{\partial \xi} \right). \quad (3.52)$$

If Equation (3.52) is solved using an explicit method, then as shown in Anderson *et al.* [30], the stability condition is

$$\frac{\left(JK_{\xi\xi} \Delta t_{diff} \right) \Big|_{i+1/2,j}}{(\Delta \xi)^2} \leq \frac{1}{2}. \quad (3.53)$$

Now $\Delta \xi = 1$ for the computational domain. Then using Equation (3.53), the time step for diffusion at cell edge $i + 1/2$, j is given by

$$\Delta t_{diff} \Big|_{i+1/2,j} \leq \left(\frac{1}{2J \left(\frac{\xi_x}{J} K_{xx} \xi_x + \frac{\xi_y}{J} K_{yy} \xi_y \right)} \right) \Big|_{i+1/2,j} . \quad (3.54)$$

Consequently, using Equation (3.54), the diffusion time step for each of the cell edges $i+1/2, j$ is obtained. In a similar fashion, the diffusion time step for each of the cell edges $i, j+1/2$ is obtained. Finally,

$$\Delta t_{diff} = \min(\Delta t_{diff} \Big|_{i+1/2,j} , \Delta t_{diff} \Big|_{i,j+1/2}) \quad \forall \ i, j . \quad (3.55)$$

The diffusion time step obtained using (3.55) is used in Equation (3.39) to compute turbulent diffusion in two dimensions.

Note that advection and turbulent diffusion are calculated using their own time steps given by Equations (3.38) and (3.55), respectively. These processes are synchronized in time to ensure that advection and diffusion advance through the same time period in any solution step. This synchronization can be accomplished using either of the following two methods: (1) choose the smaller of Δt_{adv} or Δt_{diff} to compute transport, or (2) compute advection using Δt_{adv} and then repeatedly apply diffusion using Δt_{diff} (or a fraction of it) until the total time step for diffusion equals Δt_{adv} . Since

advection is known to be the dominant transport process and accuracy of its calculation increases with the magnitude of the time step used (within the stability constraint), the second method is used in DSAGA-PPM.

3.4 Boundary Conditions for Transport

The boundary conditions for species transport are termed either *inflow* or *outflow*, depending on the direction of the flow at the grid boundary under consideration. In fluid flow situations, generally the species concentrations at inflow boundaries are known as a function of time. The outflow boundary conditions are generally not known and, therefore, need to be computed. Some discussions of the boundary conditions are provided in literature [40 - 47].

The species transport boundary conditions used in this work can be expressed as follows.

Advection

$$\text{Inflow:} \quad (\hat{u}c_l)|_{\text{boundary}} = (\hat{u})|_{\text{boundary}} c_l|_{\text{in}} ; \quad (3.56)$$

with

$$\hat{u}|_{boundary} = \left(\frac{\xi}{J} u + \frac{\xi}{J} v \right) \Big|_{boundary} . \quad (3.57)$$

An expression similar to (3.57) can be written for $\hat{u}|_{boundary}$ in the η direction.

Note that in the above expressions, *boundary* refers to the pertinent ghost cells for the concentrations $\hat{c}_i$ and grid boundary for $\hat{u}$. At any grid boundary, determination of whether inflow boundary conditions should be applied is done based on the direction of $\hat{u}|_{boundary}$ with respect to the boundary under consideration. Note also that in the upwind scheme used in this work to compute the hyperbolic advection process, specification of outlet boundary conditions is not needed.

Diffusion

Zero flux boundary conditions are used for computing turbulent diffusion in this work. These are expressed as:

$$\text{Inflow} \quad c_i|_{boundary} = c_i|_{in} ; \text{ and} \quad (3.58)$$

$$\text{Outflow} \quad c_l|_{\text{boundary}} = c_l|_{\text{out}}. \quad (3.59)$$

In the finite volume discretization used in this work, *in* and *out* refer to the cells in the *interior* of the domain that are adjacent to the pertinent ghost cells.

3.5 Treatment of Emission Sources

The pollutant emissions into the atmosphere play a major role in determining the pollutant concentration levels in the domain being modeled. These emissions enter the model domain either through the boundary conditions or as source terms in the conservative equations. This section describes the procedures used to allocate emissions into the appropriate computational cells.

Most emissions can be considered to be released from either point locations (point sources) or area regions (area sources). Point sources, by definition, are accounted for in the atmospheric diffusion equation in a manner similar to the chemical reaction terms. If a point source emits species l at the mass rate $E_l^P(\vec{X}_p, t)$ (e.g., kg/s) from the location $\vec{X}_p$, then the contribution from this point source to the rate of mass change at any location $\vec{X}$ is given by

$$E_l(\vec{X}, t) = \begin{cases} E_l^P(\vec{X}_P, t) & ; \quad \vec{X} = \vec{X}_P \\ 0 & ; \quad \vec{X} \neq \vec{X}_P \end{cases} \quad (3.60)$$

In a typical urban domain, usually there are a large number of point sources within an area defined by a typical computational cell. Rather than considering each source separately, a common practice is to aggregate all the ground level point sources within each cell and develop a comparable source term. If there are n ground level point sources located within an area A^P , then the equivalent, mass emission rate in this area is given by

$$E_l^A = \sum_{i=1}^n E_l^P(\vec{X}_i, t) \quad ; \quad \vec{X}_i \in A^P. \quad (3.61)$$

In order to determine the incremental contribution from source emissions into a computational cell, consider a particular cell of base area A_b and uniform height $h(t)$. Given the mass emission rate E_l^P from a point source, the corresponding rate of change of concentration *in the cell containing this source* is given by

$$S_l^P = \frac{E_l^P}{h(t)A_b}. \quad (3.62)$$

Similarly, given the mass emission flux E_l^A (e.g., kg/cm^2-s) from an area source, the corresponding rate of change of concentration *in a cell in this source* is given by

$$S_l^A = \frac{E_l^A}{h(t)}. \quad (3.63)$$

To be useful in comparison against ambient air quality standards the expressions (3.62) and (3.63) need to be converted to a system of concentration units expressed in term of *parts per million by volume (ppmv)*. This is accomplished by assuming that all species can be described by ideal gas laws. Under these conditions, the volume occupied by one mole of an ideal gas is given by $\frac{\Re T}{P}$ where $\Re$ is the Universal gas constant, T the absolute temperature in 0K , and P the pressure in standard atmospheres. If M_l is the molecular weight of species l , then the conversion is given by

$$c_l(ppmv) = \frac{\Re T}{M_l} c_l(\mu g / m^3). \quad (3.64)$$

The source contribution expressed in units of *ppmv/s* is then given by

$$S_l(ppmv / s) = \frac{RT}{M_l P} 10^9 S_l(kg / m^3 - s). \quad (3.65)$$

In DSAGA-PPM, the specification of emission sources with respect to the grid may change as the grid is adapted. For example, if before grid movement the grid cell m contains a point source, after grid movement cell $n \neq m$ may contain the same point source. Moreover, if a point source was located within a cell before grid movement, after grid movement the same source may be located on a cell side or a cell corner. To ensure that locations of emission sources are tracked properly as the grid is moved, DSAGA-PPM includes appropriate logic. In this logic, the equation for each of the sides of a cell is derived using the coordinates of the pertinent grid nodes. Then starting with a side and traversing in a counter-clockwise manner, the location of a particular source is checked with respect to each side of the grid cell. If this check reveals that the source is within the cell, then the process is terminated. Otherwise such checks are performed with other cells to determine the source location. The process is repeated with any other sources present in the domain being modeled.

3.6 Computation of Chemistry

In the atmospheric diffusion equation, Equation (3.1), the terms R_l , $l = 1, \dots, N$, describe the contributions to the rates of change of N chemical species concentrations, $c_1, c_2, \dots, c_N$, due to chemical reactions. At any time, for any spatial point, the rate of change of each species concentration due to chemical reactions can be described by a set of coupled, nonlinear ordinary differential equations,

$$\frac{dc_l}{dt} = R_l(c_1, c_2, \dots, c_N); \quad l = 1, \dots, N; \quad (3.66)$$

and the associated initial conditions $c_l(0) = c_l^0$; $l = 1, \dots, N$.

In this section general formulae are developed to calculate the chemical source terms for the Equations (3.66). Following the development of these formulae, a solution scheme for the system (3.66) is presented. Note that as shown in Equations (3.16), (3.17), and (3.23), the system (3.66) is solved in the operator split method employed in DSAGA-PPM.

The functional form of the terms R_l , $l = 1, \dots, N$ can be developed by considering a homogeneous system in which N single phase species participate in m reaction steps of the form

where the coefficients α_{jl} , β_{jl} denote the reactant and the product stoichiometry, respectively, in the reaction step j . As explained in Gawalas [48], if the reaction rates, r_j , of the m individual reactions are described by the mass action law

$$r_j = k_j \prod_{l=1}^N c_l^{\alpha_{jl}}, \quad j = 1, \dots, m; \quad (3.68)$$

where k_j is a temperature dependent rate constant, then the chemistry source terms are given by

$$\frac{dc_l}{dt} = R_l = \sum_{j=1}^m (\beta_{jl} - \alpha_{jl}) r_j, \quad l = 1, \dots, N. \quad (3.69)$$

Atmospheric chemistry mechanisms include reactions with characteristic time scales that differ by orders of magnitude. For such mechanisms, Equations (3.69) lead to “stiff” systems of differential equations. According to a common definition of stiffness [17], the system (3.69) is stiff if for $l = 1, \dots, N$, (a) $\text{Re}(\lambda_l) < 0$; and (b)

$$\frac{\left(\text{Max} |\text{Re} \lambda_l| \right)}{\left(\text{Min} |\text{Re} \lambda_l| \right)} = R_s \gg 1, \text{ where } R_s \text{ is the stiffness ratio and } \lambda_l \text{ are the eigenvalues of}$$

the Jacobian Matrix $J_c = \frac{\partial R_l}{\partial c_j}$; $l = 1, \dots, N$; $j = 1, \dots, N$. In general, stiff differential

equation systems would dictate that solution advancement in time be done in prohibitively small time steps.

In the last few years extensive research has been conducted to develop algorithms for solving stiff ordinary differential equations [49-54]. In numerical implementation of a solution procedure for the atmospheric diffusion equation, a large number of reaction rate equations must be integrated at a large number of grid points for relatively short periods of time between transport steps. Consequently, chemistry solvers with low start up costs, low memory requirements, and high computational speed are used in AQMs.

In this work *the asymptotic integration method* of Young and Boris [55] is used to solve the system (3.69). This method is self starting, very fast, and requires minimal storage [17]. In this method, a second order predictor-corrector scheme that takes into account the stiffness of equations is employed to integrate the system (3.69). The method has been tested against the very accurate chemistry solver EPISODE [56] and has been found to provide comparable results at significantly reduced costs [17]. A description of the asymptotic integration method is provided in Appendix D.

In atmospheric chemistry, reaction rates are often expressed in units of *molecules/cm³-s*, whereas concentrations are often expressed in units of *parts per million by volume (ppmv)*. Hence a conversion between the units of *molecules/cm³* and *ppmv* is required. This is accomplished by considering that one mole of a gas contains Avogadro number of molecules (i.e., 6.02×10^{23} molecules) and that air is a perfect gas. Then the conversion is given by

$$c(\text{molecules} / \text{cm}^3) = \frac{6.02 \times 10^{17} \times P(\text{atm})}{82.06 \times T(^{\circ}\text{K})} c(\text{ppmv}) \quad (3.70)$$

Then under standard conditions of $P = 1 \text{ atm.}$ and $T = 298^{\circ}\text{K}$,

$$c(\text{ppmv}) = \frac{c(\text{molecules} / \text{cm}^3)}{2.46 \times 10^{13}} \quad (3.71)$$

3.7 Grid Adaptation and Solution Correction

In the following subsections, procedures for moving the grid nodes to region(s) requiring solution refinement, and for conservatively redistributing the solution field over the resulting adapted grid are described.

3.7.1 SIERRA Weight Function

DSAGA-PPM requires determination of a weight function $w(x, y, t)$, which will be large in regions where grid clustering is necessary to achieve desired solution accuracy. The SIERRA weight function, developed by Laflin and McRae [14] and Laflin [15], is easy to compute and promotes both grid node clustering and grid alignment adaptation processes. For these reasons, this weight function is used in DSAGA-PPM.

As discussed in [15], the SIERRA weight function is a measure of how well the grid supports accurate interpolations of the discrete solution vector of N species in the vicinity of any cell center. For example, a relatively small value of w at a cell center would indicate that the grid can support relatively accurate interpolations of the solution in the neighborhood of that cell center. A brief description of the SIERRA weight function is provided below.

As explained in [15], at any finite volume cell center o , the expression of the SIERRA weight function, for the l^{th} chemical species is given by:

$$w_o^l = V_o^{1+e_1} \left| \Delta^2 \left(\phi_l \right)_o + w_{\min} \right|; \quad \phi_l = c_l \left(\frac{V}{V_o} \right)^{e_2}; \quad (3.72)$$

where Δ^2 is the discrete approximation of the Laplacian operator and $\left(\Delta^2 \phi_l \right)_o$ represents the error between the computed value of c_l at cell center o and the value at o obtained by interpolating the c_l values in the neighboring cells. Considering one species only, for the cell-centered finite volume scheme being used in DSAGA-PPM this error is evaluated by

$$\Delta^2 \left(\phi \right)_o = \sum_{k=1}^{N_k} (\alpha_k \phi_k) - \alpha_o \phi_o; \quad (3.73)$$

where:

$$\alpha_o = \sum_{k=1}^{N_k} (\alpha_k) \quad (3.74)$$

In (3.73), N_k is the number of distinct discrete values $\phi_k \neq \phi_o$ used in the discrete approximation of the Laplacian and α_k are constant coefficients of the values ϕ_k that define the discrete approximation. As shown in (3.74), the coefficient of ϕ_o is α_o and α_o depends on the values α_k . In this work, a five-point stencil, shown in Figure 3.1, is used to approximate $\Delta^2 \phi$. The boxes in this figure represent the values ϕ_k and the number in a box is the value of the coefficient α associated with that box.

	1	
1	- 4	1
	1	

Figure 3.1. The five-point stencil for computing discretized Laplacian.

In (3.72), the parameter e_1 controls weighting of each cell volume V in relation to its size. If e_1 is negative then smaller cells will be weighted more than larger cells and vice-versa. Thus, choosing e_1 to be negative may cause smooth flow features to be under-resolved. The parameter $e_2 \geq 0$ provides control over the rate of change of the cell volumes in the grid. If $e_2 > 0$ then evacuation of grid nodes from regions of uniform concentration will be inhibited and grid orthogonality will be promoted. The parameter w_{min} is the minimum allowable weight function value and is typically set such that $10 \times \text{machine zero} \leq w_{min} \leq 1$. Larger values of w_{min} are chosen if grid adaptation is needed only in regions with prominent solution features.

Note that (3.72) provides an expression of the weight function for the chemical species l . To form a weight function from N chemical species a linear combination of (3.72) is taken. The development of a weight function suitable for use in reacting flows is described in Chapter 5.

The weight function resulting from Equation (3-72) may result in highly sheared or skewed grids in which large cell volumes may exist next to small cell volumes. These large volumes can adversely affect the accuracy of solution calculations in the next time step. Hence, it is desirable to obtain smooth grids without highly sheared or skewed cells. In order to ensure that such grids result from the adaptation process, the weight function resulting from Equation (3-72) is smoothed by applying the discretized Laplacian to it [57].

3.7.2 Repositioning of Grid Nodes

In the adaptive grid procedure described here, repositioning of the grid nodes is accomplished by a *center-of-mass* scheme, originally proposed by Eiseman [58]. In this scheme, a grid node is repositioned such that its position coincides with the center-of-mass of a local cluster of cells, with mass distribution over the grid being defined by the weight function. Benson and McRae [8] have noted that if the grid cells have concave boundaries, then the new grid-node position provided by Eiseman's *center-of-mass* scheme may lie outside of the local cluster of cells and thereby result in grid-line cross-over. To reduce the potential for grid-line cross-over, they recommend that the *center-of-mass* scheme be applied in a parametric space and the grid nodes be moved in this space [9-12]. In air quality modeling, problems of grid crossover during adaptation can be avoided by initially choosing the modeled region to not have concave cell boundaries. Hence, in DSAGA-PPM the original *center-of-mass* scheme of Eiseman is used. Using this scheme, the new position coordinates of grid node o are given by

$$\vec{P}_o^{new} = \frac{\sum_{i=1}^4 w_i \vec{P}_i}{\sum_{i=1}^4 w_i}. \quad (3.75)$$

In (3.75), $\vec{P}_i$, $i = 1, \dots, 4$, are the position coordinates in physical space of the centers of the cells that are local to the grid node o and w_i , $i = 1, \dots, 4$, are the weights associated with these cells. The grid node o and the local cells are shown in Figure 3.2.

Figure 3.2. Grid node o and neighboring cells used in the *center-of-mass* scheme.

3.7.3 Solution Redistribution

In DSAGA-PPM, a solver-independent solution field redistribution procedure developed by Laflin [15] is employed. In this procedure, the solution field is fixed with respect to an inertial frame while a control volume, Ω^{cv} , is allowed to move arbitrarily through the spatial domain. To describe the motion of Ω^{cv} through the inertially fixed solution field, a fictitious grid time, t_g , is introduced. The moving control volume is a function of grid time (i.e., $\Omega^{cv} = \Omega^{cv}(t_g)$) but is not a function of the physical time

(i.e., $\Omega^{cv} \neq \Omega^{cv}(t)$). Conversely, the solution field is a function of the physical time, (i.e., $c_l = c_l(t)$) but is not a function of grid time (i.e., $c_l \neq c_l(t_g)$). Using these concepts, the Reynolds Transport Theorem reduces to a *Conservative Interpolation Equation* (CIE) given by

$$\frac{d}{dt_g} \int_{\Omega^{cv}(t_g)} c_l dV = \int_{\partial \Omega^{cv}(t_g)} c_l \vec{v}_b \cdot \vec{n} dS, \quad l = 1, \dots, N; \quad (3.76)$$

where $\partial \Omega^{cv}(t_g)$ is the bounding surface of $\Omega^{cv}(t_g)$, $\vec{v}_b$ is the velocity of an element of the bounding surface dS , and $\vec{n}$ is the unit outward normal vector to dS . Note that the CIE assures that control volume movement does not change the inertially-fixed solution field.

Applied to a two-dimensional grid-cell with four sides, the CIE becomes

$$\frac{d}{dt_g} (\bar{c}_l V) = \sum_{p=1}^4 \int_{\partial \Omega_p^{cv}(t_g)} c_l \vec{v}_b \cdot \vec{n} dS, \quad l = 1, \dots, N; \quad (3.77)$$

where $\bar{c}_l = \bar{c}_l(t_g)$ is the average value of c_l over the cell volume $V = V(t_g)$ and $\vec{v}_b$ and $\vec{n}$ are the velocity and unit outward normal vector of cell surface p , respectively. Integrating (3.77) over grid time levels n_g and $n_g + 1$, a form of (3.77) suitable for numerical evaluation is obtained [15]. This form is

$$\left(\bar{c}_l V\right)^{n_g+1} = \left(\bar{c}_l V\right)^{n_g} + \sum_{p=1}^4 \left(\widehat{V}_p \Big|_{n_g}^{n_g+1} \right) \left(\widehat{c}_{lp} \Big|_{n_g}^{n_g+1} \right), \quad l = 1, \dots, N; \quad (3.78)$$

where $\widehat{V}_p \Big|_{n_g}^{n_g+1}$ is the volume swept by the cell surface p during movement between grid time levels n_g and $n_g + 1$ and $\widehat{c}_{lp} \Big|_{n_g}^{n_g+1}$ is the average value of c_l in $\widehat{V}_p \Big|_{n_g}^{n_g+1}$. Note that the grid time step $\Delta t_g = (t_g^{n_g+1} - t_g^{n_g})$ does not appear in (3.78) and is not required in the solution interpolation procedure.

Equations (3.78) are used to compute the interpolated cell-averaged values for the dependent variables, $\bar{c}_l^{n_g+1}$, $l = 1, \dots, N$. To use (3.78), the values of the volumes V^{n_g} , V^{n_g+1} , and $\widehat{V}_p \Big|_{n_g}^{n_g+1}$; values of cell-averaged dependent variables before grid movement, $\bar{c}_l^{n_g}$, $l = 1, \dots, N$; and values of $\widehat{c}_{lp} \Big|_{n_g}^{n_g+1}$, $p = 1, \dots, 4$, $l = 1, \dots, N$ are needed.

The volumes V^{n_g} , V^{n_g+1} , and $\widehat{V}_p \Big|_{n_g}^{n_g+1}$ are computed using the grid node coordinates before and after grid movement. Values $\bar{c}_l^{n_g}$, $l = 1, \dots, N$ are available either as initial conditions or as a result of the time advancement of the solution discussed in Section 3.1.

However, the values $\widehat{c}_{lp} \Big|_{n_g}^{n_g+1}$, $p = 1, \dots, 4$, $l = 1, \dots, N$ need to be determined. In two-

dimensional calculations, the cell and sweep volumes can be calculated exactly and, therefore, the accuracy of determination of $\bar{c}_l^{n_g+1}$ is entirely dependent on the accuracy of

determination of $\hat{c}_{lp} \Big|_{n_g}^{n_g+1}, p=1,\dots,4, l=1,\dots,N$. An error analysis of

$\bar{c}_l^{n_g+1}, l=1,\dots,N$, conducted by Laflin [15] shows that to assure accurate determinations

of $\hat{c}_{lp} \Big|_{n_g}^{n_g+1}, p=1,\dots,4, l=1,\dots,N$: (1) grid node movements need to be restricted, and

(2) a higher-order scheme needs to be used for computing

$\hat{c}_{lp} \Big|_{n_g}^{n_g+1}, p=1,\dots,4, l=1,\dots,N$. Accordingly in DSAGA-PPM, an interim-step

procedure [15] is used to increase the accuracy of solution interpolation while allowing

for an arbitrary amount of grid movement and a higher-order scheme - PPM- is used to

compute $\hat{c}_{lp} \Big|_{n_g}^{n_g+1}, p=1,\dots,4, l=1,\dots,N$.

In the interim-step procedure, the grid time step, $\Delta t_g = (t_g^{n_g+1} - t_g^{n_g})$, is divided into M smaller interim steps, δt_g . If the change in position coordinates of a grid node o

over Δt_g is $\Delta \vec{X}_o = \vec{X}_o^{n_g+1} - \vec{X}_o^{n_g}$ then the change in position coordinates of this grid

node over δt_g is given by $\delta \vec{X}_o = (\Delta \vec{X}_o) / M$. In DSAGA-PPM, the value of M is

chosen such that during any interim step, the movement of any cell side is restricted to

one-half of the corresponding cell length. The interpolated cell-averaged values for the dependent variables, $\bar{c}_l^{n_g+1}$, $l = 1, \dots, N$, are obtained by M iterative applications of

$$\bar{c}_l^{n_g+\beta_m} = \frac{1}{V^{n_g+\beta_m}} \left((\bar{c}_l V)^{n_g+\beta_{m-1}} + \sum_{p=1}^4 \left(\widehat{V}_p \left| \begin{matrix} n_g+\beta_m \\ n_g+\beta_{m-1} \end{matrix} \right. \right) \left(\widehat{c}_{lp} \left| \begin{matrix} n_g+\beta_m \\ n_g+\beta_{m-1} \end{matrix} \right. \right) \right), \quad l = 1, \dots, N. \quad (3.79)$$

In (3.79), m is the interim-step counter such that $m = 1, \dots, M$; $\beta_m = m / M$ with $\beta_0 = 0$.

3.7.4 Grid Convergence

The *center-of-mass* scheme, depicted by (3.75), is a discrete approximation of the elliptic Poisson's equation [15]. Therefore, an iterative process is used to obtain adequately converged grid node positions. At any time, the following sequence of steps is taken: (1) weights are computed using the SIERRA approach described in Section 3.7.1; (2) grid nodes are moved using (3.75); (3) solution is redistributed as described in Section 3.7.3; and (4) a check is made on grid convergence. If the grid is converged within a specified tolerance, then solution is advanced through a new time step, otherwise weights are recomputed and grid movement - solution redistribution procedure is repeated. In DSAGA-PPM, a limit, δ , is set on the maximum movement of grid nodes relative to largest cell side in the starting Cartesian grid such that

$$\frac{MAX|\Delta\vec{X}_{i,j}|}{MAX(\Delta x, \Delta y)_{starting\ Cartesian\ grid}} \leq \delta \quad \forall \quad i, j; \quad (3.80)$$

where $\Delta\vec{X}_{i,j}$ is the change in position coordinates of the node i, j . If (3.80) was satisfied, then the grid was considered to be converged.

3.7.5 Preadaptation

In an air quality simulation, the domain being modeled will, in general, contain regions with relatively large gradients in species concentrations. Such gradients may result from complex interactions between emissions from sources, meteorological conditions, and atmospheric chemistry. A simulation using DSAGA-PPM would start with a uniform distribution of grid nodes and would modify this distribution based on the spatial resolution requirements of the various species. Consequently, using the starting uniform grid with initial gradients in species concentration can result in an inaccurate calculation of the solution field in the first time step. In order to remedy this potentiality, a preadaptation step has been included in the DSAGA-PPM algorithm. In this step, the starting uniform grid is preadapted to regions with initial concentration gradients before calculating the solution field in the first time step. In general, this preadaptation is accomplished by computing weights based on the initial solution field, moving the grid nodes, checking grid convergence, and stopping the preadaptation process once the grid is

converged, as described in the previous sections. There could, however, be situations in which the initial solution field may not contain any concentration gradients but sources in the domain start emitting at the beginning of the first time step. In such situations, emissions from sources are assumed to occur before the first time step, preadaptation process is completed based on the gradients resulting from these emissions, and then the solution field is reinitialized with background concentrations.

4 Model Advection Problems

The accuracy of results produced by an AQM depends to a large measure on the accuracy of computation of the advection process. This is because of two reasons. First, horizontal transport of pollutant species is dominated by advection. Second, advection affects the accuracy of the species concentration data available at the beginning of each chemistry integration step and these data can have a significant influence on the results of atmospheric chemistry calculations. Thus, it is very important that the advection component of an AQM be computed accurately.

Tests with model advection problems were used in this work to characterize the improvements in accuracy achieved by using DSAGA-PPM. These problems have known analytical solutions. The tests were conducted on a CRAY T90 vector processing machine. For each of the model problems, DSAGA-PPM and corresponding SGA-PPM results were obtained and compared.

Following Smolarkiewicz and Rasch [59], the errors in the numerical solutions were characterized by the following normalized measures:

$$EMIN = \frac{c^{\min} - c_e^{\min}}{c_e^{\max}}, \quad (4.1)$$

$$EMAX = \frac{c^{\max} - c_e^{\max}}{c_e^{\max}}, \quad (4.2)$$

$$EMAS = \frac{\sum cV - \sum c_e V}{\sum c_e V}, \quad (4.3)$$

$$EVAR = \frac{\sum c^2 V - \sum c_e^2 V}{\sum c_e^2 V}, \quad (4.4)$$

and

$$ERMS = \sqrt{\sum \left(\frac{c - c_e}{c_e} \right)^2 V / \sum V}. \quad (4.5)$$

In the expressions above, *min* and *max* refer to the global minimum and maximum values, respectively in the solution field; the subscript *e* stands for the analytical solution; and $\sum$ is the discrete integral over the modeled region. As defined above, the quantities EMIN, EMAX, EMAS, EVAR, and ERMS represent, respectively, the normalized error in the minimum value of the solution field, the normalized error in the maximum value of the solution field, the normalized error of the total mass, the normalized error of the variance of the solution field, and the normalized root-mean-square error of the field.

Thus, EMIN indicates the maximum undershoot in the solution field; EMAX measures the damping or overshoot of the initial solution peak; EMAS indicates the extent of mass conservation; EVAR is a measure of numerical diffusion; and ERMS indicates overall error in the solution field. In the definitions of EMAS, EVAR, and ERMS, the integrals are weighted by cell volume to account for the significant cell volume variation resulting from grid adaptation.

4.1 One-dimensional Tests

In these tests, one-dimensional advection of three different waveforms is examined in a region of 32 km, discretized with 32 cells. The waveforms selected for these tests are a rectangular wave, a triangular wave, and a gaussian distribution. Initially, each of these waveforms has a base of 8 km and a peak concentration of 100 units and is centered at a distance of 8.5 km from the left edge of the region. The background concentration in each test is chosen to be 5 units. Each of the waveforms is advected from left to right through a distance of 16 km by a wind with a constant speed of 0.25 km/h. The analytical solution for each of these tests is translation of the initial waveform through the simulation time period. Similar tests with waveform amplitudes of unity and a background concentration of zero have been conducted by others [17, 33]. However, to rigorously test the ability of DSAGA-PPM to resolve steep gradients and to produce monotonic solution fields, for each of the waveforms the peak concentration was increased to 100 units and the minimum concentration was selected to be greater than

zero. The resulting rectangular and triangular waveforms represent pollutant distributions with sharp gradients and discontinuities while the gaussian waveform represents a well-resolved, smoothly varying distribution.

For each of the waveforms, the simulations were conducted using a CFL of 0.8. The values of the adaptation parameters used in these simulations were determined through numerical experiments conducted to obtain good preadaptation (see Section 3.7.5). These values are shown in Table 4.1 below. The parameters e_1 , e_2 , w_{min} , and the smoothing iterations, related to the SIERRA weight function, are described in Section 3.7.1. The grid convergence parameter, δ , is described in Section 3.7.4.

Table 4.1. Values of Adaptation Parameters Used in One-Dimensional Tests

Waveform - Solution	e_1	e_2	w_{min}	Smoothing Iterations	δ
Rectangle - DSAGA-PPM	-0.7	0	1.E-05	9	1.E-02
Triangle - DSAGA-PPM	-0.7	0	1.E-04	5	1.E-04
Gaussian - DSAGA-PPM	-0.7	0	1.E-04	5	1.E-04

Note also that in DSAGA-PPM simulations, the movement of any cell side was restricted to one-half of the corresponding cell length in the interim step procedure described in Section 3.7.3.

The results of the computations are shown in Figures 4-1, 4-2, and 4-3. Note that in the DSAGA-PPM result obtained for each of the waveforms, the grid nodes are clustered in the areas with large changes in solution values. The error characteristics, defined in at the beginning of this chapter, are shown in Table 4.2 for each of the tests.

Table4.2. Error Characteristics for the One-Dimensional Tests

Waveform - Solution	EMIN	EMAX	EMAS	EVAR	ERMS
Rectangle - SGA-PPM	0	-1.9E-06	-4.9E-06	-8.5E-02	1.2E00
Rectangle - DSAGA-PPM	4.1E-04	-5.5E-04	-7.4E-05	-3.9E-02	5.0E-01
Triangle - SGA-PPM	0	-1.4E-01	0	-1.8E-02	2.0E-01
Triangle - DSAGA-PPM	1.9E-04	-8.9E-02	-1.2E-04	2.0E-03	1.8E-02
Gaussian - SGA-PPM	0	-1.1E-01	-1.8E-07	-2.0E-02	6.3E-02
Gaussian - DSAGA-PPM	3.0E-04	-4.3E-02	-1.5E-04	6.7E-03	4.1E-02

A comparison of the error characteristics yields the following: 1) the adaptive grid solutions do not increase spurious oscillations significantly, as seen by the very low values of EMIN for each DSAGA-PPM result; 2) For the triangle and the gaussian waveforms, the adaptive grid maintains waveform peaks better than the static grid (compare the values of EMAX); 3) the adaptive grid solutions lose slightly more global

mass than the corresponding static grid solutions, but in each case the global mass balance is still quite reasonable (indicated by the low values of EMAS); 4) the adaptive grid solutions show reduced numerical diffusion (compare magnitudes of EVAR); and (5) for each of the waveforms, the adaptive grid solution has less overall error than the corresponding static grid solution (compare the ERMS values). Note that for the rectangular waveform, the EMAX values in simulations with SGA-PPM and DSAGA-PPM are quite low, indicating good peak maintenance by both of these algorithms. The effect of numerical diffusion on this waveform would tend to smooth the shoulders of the wave (see Figures 4-1(a) and 4-1(b)). This smoothing occurring symmetrically on either side of the centerline of the waveform would not have a large impact on the “peak” waveform value at the centerline. Thus, the “peak” value of the waveform is fairly well maintained in each of the SGA-PPM and DSAGA-PPM simulations.

The one-dimensional tests discussed above reflect that DSAGA-PPM provided significantly improved advection results compared to results obtained using SGA-PPM. Additional advection tests with more complicated two-dimensional model problems were conducted to corroborate the findings of this section. These two-dimensional tests are described in the next two sections.

(a)

(b)

Figure 4-1. Rectangular waveform solution obtained with (a) SGA-PPM
(b) DSAGA-PPM

(a)

(b)

Figure 4-2. Triangular waveform solution obtained with (a) SGA-PPM (b) DSAGA-PPM

(a)

(b)

Figure 4.3. Gaussian waveform solution obtained with (a) SGA-PPM (b) DSAGA-PPM

4.2 A Rotating Conical Distribution

In this test, the solid body rotation of a conical distribution is examined in a two-dimensional region that is 42 km long in each of x - and y -directions. The region is initially discretized with 43×43 grid nodes, spaced uniformly. The initial conditions consist of a conical concentration distribution (cone) with a base radius of 4 km and a peak concentration of 100 units, centered at coordinates (26.5 km, 21.5 km) in the region. The background concentration in the region is 5 units. This cone is advected in the counterclockwise direction around the center of the region by a wind with a constant angular velocity of 0.1 rad/h. The analytical solution of the above problem is a solid body rotation of the cone [17]. Note that the cone presents a relatively complex solution feature with a steep gradient at the side and discontinuities at the apex and at the juncture with the background. As in the one-dimensional tests, in order to rigorously test DSAGA-PPM, the peak concentration was increased over those used in other works [17, 60, 61] and the background concentration was selected to be greater than zero.

In each of the simulations the CFL was set at 0.8. In the DSAGA-PPM simulation, the values of the SIERRA weight function related parameters e_1 , e_2 , w_{min} , and smoothing iterations, described in Section 3.7.1, were set at -1.0, 0, 8.E-03, and 15, respectively; and the value of the grid convergence parameter δ , described in Section 3.7.4, was set at 3.E-02. Additionally, in the interim step procedure described in Section

3.7.3, the movement of any cell side was restricted to one-half of the corresponding cell length. The values of the adaptation parameters were determined through numerical experiments and were selected to obtain a good preadapted grid (see Section 3.7.5). This grid is shown in Figure 4.4.

Initially, a version of the adaptive grid code without the correction for numerical sources, described in Section 3.1, was used. Shown in Figure 4.5(a) is the solution after two time steps, obtained with this code. This solution clearly reflects oscillations in the concentration field. The code was then modified to incorporate the correction for numerical sources described in Section 3.1. Shown in Figure 4.5(b) is the solution after two time steps, obtained with the modified code. This solution is free of oscillations.

Finally, one revolution of the cone is followed using SGA-PPM and DSAGA-PPM. The results shown in Figures 4.6(b) and 4.6(a) reflect that 87% of the peak is retained using DSAGA-PPM while only 63% of the peak concentration is retained using SGA-PPM. This illustrates that the adaptive grid locally reduces numerical diffusion and thereby provides better peak concentration maintenance. As shown in Figure 4.7, the nodes of the adaptive grid at the end of the simulation are clustered in the cone and have tended to align with the discontinuity at the background juncture.

Figure 4.4. Preadapted grid showing clustering of nodes near the cone

(a)

(b)

Figure 4.5. Solution field (a) with oscillations (b) without oscillations

Peak concentration = 63
Minimum concentration = 5

(a)

Peak concentration = 87
Minimum concentration = 5

(b)

Figure 4.6. Rotating cone results obtained with (a) SGA-PPM (b) DSAGA-PPM

Figure 4.7. Adaptive grid after one revolution of the cone

The error measures introduced before were used to further compare the static and adaptive grid solutions. The values of these measures are shown in Table 4.3. The small value of EMIN for the DSAGA-PPM solution indicates that this algorithm did not introduce appreciable undershoots as a consequence of improving resolution. Further, the value of EMAS for the DSAGA-PPM solution reflects that the algorithm has maintained global mass with reasonable accuracy (about 0.45 % gain). The values of EMAX for the SGA-PPM and DSAGA-PPM solutions reflect that the adaptive grid maintains solution features better than the static grid (compare peak values of 87 and 63, seen in Figures 4.6(b) and 4.6(a)). Finally, the magnitudes of EVAR and ERMS reflect that compared to the static grid solution, the adaptive grid solution undergoes less numerical diffusion and has less overall (root-mean-square) error.

Table 4.3. Error Characteristics for the Simulations of a Rotating Cone

Solution	EMIN	EMAX	EMAS	EVAR	ERMS
SGA-PPM	0	-3.7E-01	2.8E-14	-1.1E-01	9.3E-02
DSAGA-PPM	6.6E-09	-1.3E-01	4.5E-03	4.8E-02	2.6E-02

4.3 Four Rotating Conical Distributions

In the second two-dimensional test, the ability of DSAGA-PPM to resolve multiple concentration distributions is examined. In this test, four conical distributions (cones), located in a 42 km x 42 km two-dimensional region, are advected in the counterclockwise direction by a wind with a constant angular velocity of 0.1 rad/hr. The region is initially discretized with 43 x 43 grid nodes, spaced uniformly. The initial conditions comprise of four pollutant cones, each with a base radius of 4 km and a peak concentration of 100 units, centered at coordinates (30.5, 21.5), (20.5, 30.5), (11.5, 20.5), and (21.5, 11.5) in the two-dimensional region. The background concentration in the region is 5 units.

In each of the simulations the CFL was set at 0.8. In the DSAGA-PPM simulation, the values of the SIERRA weight function related parameters e_1 , e_2 , w_{min} , and smoothing iterations, described in Section 3.7.1, were set at -1.0, 0, 1.E-03, and 15, respectively; and the value of the grid convergence parameter δ , described in Section 3.7.4, was set at 4.E-02. Additionally, in the interim step procedure described in Section 3.7.3, the movement of any cell side was restricted to one-half of the corresponding cell length. The values of the adaptation parameters were determined through numerical experiments and were selected to obtain a good preadapted grid (see Section 3.7.5). This grid is shown in Figure 4.8.

Figure 4.8. Preadapted grid showing clustering of nodes near the cones

The results after one full revolution of the cones, obtained without and with grid adaptation, are shown in Figures 4.9(a) and 4.9(b), and the corresponding error characteristics are displayed in Table 4.4. A comparison of the error characteristics for the static and the adaptive grid solutions yields observations similar to those seen in the previous test cases: 1) the adaptive grid solution does not increase spurious oscillations significantly (indicated by the low value of EMIN); 2) DSAGA-PPM maintains solution features better than SGA-PPM (e.g., peak of 76 versus 57 as seen in Figures 4.9(b) and 4.9(a)); 3) DSAGA-PPM maintains global mass with reasonable accuracy (1.5% gain in global mass indicated by EMAS); and 4) the adaptive grid solution has less overall (root-mean-square) error than the static grid solution. The adaptive grid at the end of the computation, shown in Figure 4.10, is clustered around the solution field features. These observations reflect that DSAGA-PPM is able to resolve multiple zones of interest and provide an accurate solution field.

The accuracy of any DSAGA-PPM solution would appear to depend in part on the number of nodes available for resolving the solution field features. To examine this hypothesis, a simulation of one revolution of the cones was completed using an adaptive grid with 85 x 85 grid nodes. This refined grid provided approximately the same number of grid nodes per solution feature (cone) as the single cone test case conducted with 43 x 43 grid nodes. The grid adaptation parameters and CFL used in this simulation were identical to those used in the simulation with 43 x 43 nodes.

(a)

(b)

Figure 4.9. Results for four cones obtained with (a) SGA-PPM (b) DSAGA-PPM

Figure 4.10. Adaptive grid after one revolution of the cones

The DSAGA-PPM result after one full revolution of the cones is shown in Figure 4.11 and the corresponding error characteristics are displayed in Table 4.4. A comparison of the error characteristics for the two DSAGA-PPM solutions and Figures 4.9(b) and 4.11 reflects that solution quality improves as the number of grid nodes available for adaptation increases. It is also worth noting that by maintaining approximately the same number of grid nodes per solution feature, the values of the cone peaks in DSAGA-PPM solution for the four cones are identical to the value of the peak for the single cone, after one revolution (see Figures 4.11 and 4.6(b)). Finally, the adaptive refined grid at the end of the simulation, shown in Figure 4.12, has nodes clustered around the solution field features.

Table 4.4. Error Characteristics for the Simulations of Multiple Rotating Cones

Solution	EMIN	EMAX	EMAS	EVAR	ERMS
SGA-PPM	6.8E-14	-4.3E-01	-7.0E-09	-2.0E-01	2.7E-01
DSAGA-PPM (43x43 nodes)	1.6E-05	-2.4E-01	1.5E-02	-2.2E-02	1.5E-01
DSAGA-PPM (85 x 85 nodes)	7.1E-10	-1.3E-01	1.2E-02	1.3E-02	5.2E-02

Figure 4.11. DSAGA-PPM solution obtained using a refined grid with 85 x 85 nodes

Figure 4.12. Adaptive refined (85 x 85 nodes) grid after one revolution of the cones

5 Weight Function for Flows with Reacting Species

In the previous chapter, the capability of DSAGA-PPM to calculate accurate advection solutions was characterized. However an AQM includes coupled transport and nonlinear chemistry processes. These processes can interact and significantly alter the distributions of species in time. The ability of an adaptive grid algorithm to adapt to such changes can only be seen and evaluated in model problems including chemistry. Consequently, it is desirable to examine the capability of DSAGA-PPM to obtain accurate solutions of model problems with advection, turbulent diffusion, and chemistry. However, before conducting such an evaluation, it is necessary to develop a weight function that can be used in flows with reacting species. Described in this chapter is such a weight function.

In simulation of a flow with reacting species, these species may undergo complex dynamic transformations in time. Therefore, it may not be possible to predict *a-priori* which of the species would dictate grid adaptation requirements in such a simulation. Moreover in complex flows with reacting species, such as those in the atmosphere, the chemistry processes in one spatial region may be quite different from those in another region. As a result, some of the species may need resolution in one part of the modeled region while other species may need resolution in other part(s) of the region. Further, these resolution needs may change with time. One possible strategy to ensure adequate

grid adaptation in such a flow may be to use a weight function that takes into account the resolution requirements of each of the species included in the chemical mechanism, at each time step in the simulation.

A weight function for use in simulations of reacting flows may be constructed using a linear combination of the species-specific weights w_o^l (see Equation (3-72)) expressed as

$$w_o = V_o^{1+e_1} \left| \sum_l \Delta^2(\phi_l)_o + w_{\min} \right|; \quad \phi_l = c_l \left(\frac{V}{V_o} \right)^{e_2} \quad (5-1)$$

where $\left(\Delta^2 \phi_l \right)_o$ represents the error in computed value of c_l at cell center o with respect to the value at o obtained using interpolation of c_l values in the neighboring cells (see Section 3.7.1). Defined thus, the weight w_o at cell center o would include information on the interpolation error at o in concentration of each of the species l and, therefore, would be responsive to resolution needs of each of these species at o . However, in a flow with reacting species, the concentrations of these species may vary over large ranges and may undergo significant changes due to chemical activity. Therefore, each of the species-specific components $\left(\Delta^2 \phi_l \right)_o$ in Equation (5.1) needs to be scaled such that: (1) the interpolation error $\left(\Delta^2 \phi_l \right)_o$ is relatively independent of the magnitude of c_l , (2) the

scaled $\left(\Delta^2\phi_l\right)_o$ adequately represent the resolution requirements at o for each of the species l , and (3) the scaling process is responsive to the dynamic changes in species concentrations. Taking these requirements into consideration, a weight function suitable for use in computations involving chemistry can be developed. This development is described below.

Construction of the weight function with above considerations starts with determination of the species-specific SIERRA interpolation error at cell center i, j , given by

$$\left(\Delta^2\phi_l\right)_{i,j} = \left|c_l\big|_{i-1,j} + c_l\big|_{i+1,j} + c_l\big|_{i,j-1} + c_l\big|_{i,j+1} - 4 \cdot c_l\big|_{i,j}\right| \quad (5-2)$$

Note that in this work, the value of the parameter e_2 shown in Equation (5.1) is chosen to be zero. Therefore, for example, $c_l\big|_{i+1,j}$ in Equation (5-2) is not weighted by $\left(\frac{V_{i+1,j}}{V_{i,j}}\right)^{e_2}$, as required in Equation (5-1).

The concentrations of various species can differ by several orders of magnitude and, in general, would contain computational noise. The effects of this noise would need to be removed from the SIERRA interpolation errors given by Equation (5-2) before these

errors are normalized and re-scaled. In this work, the interpolation errors are adjusted to remove the computational noise as follows:

$$c_l^{avg} = \frac{\sum_{i,j} c_l|_{i,j}}{nn} ; \quad nn = \text{number of grid cells} \quad (5-3)$$

$$\begin{aligned} \left(\Delta^2 \phi_l\right)_{i,j}^{adjusted} &= \frac{\left(\Delta^2 \phi_l\right)_{i,j}}{c_l^{avg}} \quad \forall \quad \frac{\left(\Delta^2 \phi_l\right)_{i,j}}{c_l^{avg}} > 1.E-03 \\ &= 0 \quad \forall \quad \frac{\left(\Delta^2 \phi_l\right)_{i,j}}{c_l^{avg}} \leq 1.E-03 \end{aligned} \quad (5-4)$$

The interpolation error given by Equation (5-4) is normalized using

$$\left(\Delta^2 \phi_l\right)_{i,j}^{norm} = \frac{\left(\Delta^2 \phi_l\right)_{i,j}^{adjusted}}{w_{norm}^l} ; \quad w_{norm}^l = \text{MAX}\left(\left(\Delta^2 \phi_l\right)_{i,j}^{adjusted}\right) \quad \forall \quad i, j \quad (5-5)$$

This normalization process scales the range of each of the $\left(\Delta^2 \phi_l\right)_{i,j}^{norm}$ to be between 0 and

1 and, therefore, satisfies the requirement (1) given above.

Note that the presence of a relatively large range in $\left(\Delta^2\phi_l\right)$ over the entire grid would reflect that the spatial distribution of the concentration of species l requires significant resolution. Since the process of normalization given above compresses the range of each $\left(\Delta^2\phi_l\right)$, it becomes necessary to restore each of these ranges by re-scaling.

First a linear combination of the species-specific interpolation errors is formed

$$wc_{i,j} = \sum_l \left(\Delta^2\phi_l\right)_{i,j}^{norm} \quad (5-6)$$

Then this linear combination is re-scaled using

$$wc_{i,j}^{rescaled} = \left(wc_{i,j} - wc_{\min}\right) \frac{\left(\left(\Delta^2\phi\right)_{\max} - w_{\min}\right)}{\left(wc_{\max} - wc_{\min}\right)} + w_{\min} \quad (5-7)$$

where,

$$wc_{\min} = MIN\left(wc_{i,j}\right) , \quad wc_{\max} = MAX\left(wc_{i,j}\right) \quad \forall \quad i,j \quad (5-8)$$

$$\left(\Delta^2\phi\right)_{\max} = MAX\left(w_{norm}^l\right) \quad \forall \quad l \quad (5-9)$$

and the parameter $w_{\min}$ is a user-selected minimum allowable weight function value. This value is typically set such that $10 \times \text{machine zero} \leq w_{\min} \leq 1$. Larger values of $w_{\min}$ are chosen if grid adaptation is needed only in regions with prominent solution features.

The re-scaling scheme utilizing Equations (5-7) through (5-9) adjusts the range of values (maximum to minimum) of the $wC_{i,j}^{\text{rescaled}}$ to be between $w_{\min}$ and $(\Delta^2\phi)_{\max}$. Since $(\Delta^2\phi)_{\max}$ represents the largest SIERRA interpolation error value over the entire grid, this re-scaling scheme ensures that the resolution requirements of all species over the entire grid are represented in the weight function. Thus, the requirement (2) given above is satisfied. Further, the normalization and re-scaling process depends on the time-dependent, species-specific SIERRA interpolation errors and, therefore, satisfies the requirement (3) given above.

Note that in Equation (5-7) the lower end of the range of $wC_{i,j}^{\text{rescaled}}$ is fixed at $w_{\min}$. Consequently, $w_{\min}$ can be used to control the degree of adaptation (or the amount of grid movement). Finally, note the normalizing and re-scaling procedure described above requires only the selection of a value for $w_{\min}$ by the user. This value is chosen based on the experiments conducted to obtain a good preadapted grid in a simulation.

Using the $wc_{i,j}^{rescaled}$, the weight function at a cell center i, j is given by (see Equation (5-1))

$$w_{i,j} = V_{i,j}^{1+e_1} |wc_{i,j}^{rescaled}| \quad (5-10)$$

The weight function resulting from Equation (5-10) may result in highly sheared or skewed grids in which large cell volumes may exist next to small cell volumes. These large volumes can adversely affect the accuracy of solution calculations in the next time step. Hence, it is desirable to obtain smooth grids without highly sheared or skewed cells. Therefore, the weight function resulting from Equation (5-10) is smoothed by applying the discretized Laplacian to it [57].

6 A Rotating and Reacting Conical Pollutant Puff

As mentioned before, the coupled transport and nonlinear chemistry processes occurring in the atmosphere can cause significant dynamic changes in the distributions of species in time. For example, the shape of a puff of pollutants can change constantly due to changes in the gradients and peaks of the pollutant distributions in the puff. The ability of DSAGA-PPM to adapt to such changes can only be seen and evaluated in tests with model problems incorporating chemistry. The first model problem with chemistry examined in this work is one in which a pollutant puff undergoes advection and chemistry. Such a model problem has been used by Odman and Russel [62] and Chock and Winkler [63] in their evaluations of static nested grid applications.

6.1 Chemical Mechanism

Following Hov et al. [64] as well as [62] and [63], a simple chemical mechanism incorporating production and destruction of tropospheric O_3 was used in this work. This mechanism is shown in Table 6.1. The solar zenith angle, θ , that appears in the photolysis reactions of Table 6.1, is held constant at 71.5° . This value corresponds to the average zenith angle experienced during an equinox day at the equator [62].

Table 6.1. A Simplified Chemical Mechanism for Photochemical Production of Ozone

Reaction	Rate
$\text{HC} + \text{OH} \rightarrow 4 \text{RO}_2 + 2\text{HCHO}$	$k_1 = 6.0 \times 10^{-12}$
$\text{HCHO} + h\nu \rightarrow 2 \text{HO}_2 + \text{CO}$	$J_2 = 7.8 \times 10^{-5} e^{-0.87/\cos \theta}$
$\text{RO}_2 + \text{NO} \rightarrow \text{NO}_2 + \text{HCHO} + \text{HO}_2$	$k_3 = 8.0 \times 10^{-12}$
$\text{NO} + \text{HO}_2 \rightarrow \text{NO}_2 + \text{OH}$	$k_4 = 8.3 \times 10^{-12}$
$\text{NO}_2 + h\nu \rightarrow \text{NO} + \text{O}_3$	$J_5 = 1.0 \times 10^{-2} e^{-0.39/\cos \theta}$
$\text{NO} + \text{O}_3 \rightarrow \text{NO}_2 + \text{O}_2$	$k_6 = 1.6 \times 10^{-14}$
$\text{O}_3 + h\nu \rightarrow \text{O}_2 + \text{O}(1\text{D})$	$J_7 = 1.9 \times 10^{-4} e^{-1.9/\cos \theta}$
$\text{O}(1\text{D}) + \text{H}_2\text{O} \rightarrow 2 \text{OH}$	$k_8 = 2.3 \times 10^{-11}$
$\text{NO}_2 + \text{OH} \rightarrow \text{HNO}_3$	$k_9 = 1.0 \times 10^{-11}$
$\text{CO} + \text{OH} \rightarrow \text{CO}_2 + \text{HO}_2$	$k_{10} = 2.9 \times 10^{-13}$

6.2 Description of the Model Problem

The model problem consists of advecting a pollutant puff that initially contains colocated conical distributions of nitric oxide (NO), nitrogen dioxide (NO₂), formaldehyde (HCHO), and lumped hydrocarbons (HC) with the peak concentrations shown in Table 6.2. Each of these initial distributions has a base radius of 4 km and is

centered at coordinates (26.5, 21.5) in a domain that is 42 km long in each of x - and y -directions. The initial background concentration for each of the species is also shown in Table 6.2. These concentrations are identical to those used by [62]. The puff is advected in the counterclockwise direction around the center of the two-dimensional domain by a wind with a constant angular velocity. During advection, the species in the chemical mechanism undergo chemical transformations following the reactions described in Table 6.1 and result in formation of O_3 .

The chemistry induced changes in the peak and the background values of NO , NO_2 , $HCHO$, and O_3 species are shown in Figures 6.1 and 6.2, for time periods of 150 seconds and 86,400 seconds (24 hours), respectively. From these figures it is evident that rapid transformations of the species occur in the initial 150 seconds and that by 86,400 seconds the peak concentrations of NO and NO_2 are inverted. In light of these results, two different tests were designed. In the first test, the ability of DSAGA-PPM to respond to resolution requirements generated by rapid nonlinear chemical transformations is examined by completing one revolution of the puff in 150 seconds. In the second test, the ability of DSAGA-PPM to respond to the drastic changes in the pollutant distributions (peak inversions) is examined by completing one revolution of the puff in 86,400 seconds (24 hours).

Table 6.2. Initial Concentrations of Species in the Pollutant Puff

Species	Background (molecules/cm ³)	Peak (molecules/cm ³)
CO	1.00E+12	
H ₂ O	2.50E+15	
HC	2.50E+09	1.00E+11
HCHO	1.25E+10	5.00E+11
HO ₂	1.00E+06	
NO	2.50E+09	1.00E+11
NO ₂	2.50E+09	1.00E+11
O(1d)	1.00E-03	
O ₃	5.00E+11	
OH	1.00E+05	
RO ₂	1.00E+06	

The chemistry induced changes in the peak and the background values of NO, NO₂, HCHO, and O₃ species are shown in Figures 6.1 and 6.2, for time periods of 150 seconds and 86,400 seconds (24 hours), respectively. From these figures it is evident that rapid transformations of the species occur in the initial 150 seconds and that by 86,400 seconds the peak concentrations of NO and NO₂ are inverted. In light of these results,

two different tests were designed. In the first test, the ability of DSAGA-PPM to respond to resolution requirements generated by rapid nonlinear chemical transformations is examined by completing one revolution of the puff in 150 seconds. In the second test, the ability of DSAGA-PPM to respond to the drastic changes in the pollutant distributions (peak inversions) is examined by completing one revolution of the puff in 86,400 seconds (24 hours).

6.3 Reference Solution

In the absence of diffusion, the analytically exact solution of advection and chemistry processes can be assembled as follows. If only advection is occurring, then in any time period δt the concentration c_1 of a species is moved from a location $\vec{X}_1$ to another location $\vec{X}_2$. On the other hand if only chemistry is occurring, then the initial concentration c_1 of the species at $\vec{X}_1$ is simply converted to, for instance, c_2 . Therefore, if both advection and chemistry are present then the concentration at $\vec{X}_2$ should be c_2 .

For each of the tests, a reference solution was created using the above procedure. Note that a reference solution on a finite volume grid is simply the discretized representation of the corresponding analytically exact solution on that grid. Consequently, in each of the test cases, a reference solution was used to assess the accuracy of the numerical results.

(a)

(b)

Figure 6-1. Changes in concentrations of selected species over 150 seconds
 (a) peak concentrations (b) background concentrations

(a)

(b)

Figure 6.2. Changes in concentrations of selected species over 86,400 seconds
 (a) peak concentrations (b) background concentrations

6.4 Simulation for 150 Seconds

For the numerical simulations, the domain was initially discretized with 43×43 uniformly spaced grid nodes and the CFL was set at 0.8. In DSAGA-PPM simulation, the values of the SIERRA weight function related parameters e_1 , e_2 , w_{min} , and smoothing iterations, described in Section 3.7.1, were set at -1.0, 0, 1.E-05, and 10, respectively; and the value of the grid convergence parameter δ , described in Section 3.7.4, was set at 3.E-02. Additionally, in the interim step procedure described in Section 3.7.3, the movement of any cell side was restricted to one-half of the corresponding cell length. The values of the adaptation parameters were determined through numerical experiments and were selected to obtain a good preadapted grid (see Section 3.7.5). This grid is shown in Figure 6.3.

Figure 6.3. Preadapted grid reflecting clustering of nodes in and around the pollutant puff

Presented in Figures 6.4 through 6.11 are the SGA-PPM and DSAGA-PPM solutions as well as the corresponding reference solutions, for NO, NO₂, O₃, and HCHO, after one revolution of the puff. Also presented in Figure 6.12 is the resulting adaptive grid after 150 seconds with grid nodes clustered in and around the puff.

It is interesting to qualitatively compare the results of the simulation with the chemistry results presented in Figures 6.1(a) and 6.1(b). Per Figures 6.1(a) and 6.1(b), between 0-150 seconds the peak and background NO concentrations drop continuously, and at 150 seconds the peak concentration is higher than the background concentration. The results in Figures 6.4 and 6.5 are consistent with these observations. In each of these figures, the distribution of NO concentration has a valley (formed by the folding-in of the initial peak), the floor of the valley is higher than the background, and the background is much lower than the starting value of about 4 ppb. However, while the NO valley in the DSAGA-PPM solution is as well developed as that in its reference solution, the same is not the case for the SGA-PPM solution.

Maximum concentration = 0.55 ppb
 Minimum concentration = 0.06 ppb
 Background concentration = 0.06 ppb

(a)

Maximum concentration = 0.45 ppb
 Minimum concentration = 0.06 ppb
 Background concentration = 0.06 ppb

(b)

Figure 6.4. NO distribution after 150 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution

(a)

(b)

Figure 6.5. NO distribution after 150 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA solution

(a)

(b)

Figure 6.6. NO_2 distribution after 150 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution

(a)

(b)

Figure 6.7. NO_2 distribution after 150 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA solution

(a)

(b)

Figure 6.8. O_3 distribution after 150 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution

(a)

(b)

Figure 6.9. O_3 distribution after 150 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA solution

Maximum concentration = 23.54 ppb

Minimum concentration = 0.51 ppb

Background concentration = 0.51 ppb

(a)

Maximum concentration = 14.03 ppb

Minimum concentration = 0.51 ppb

Background concentration = 0.51 ppb

(b)

Figure 6.10. HCHO distribution after 150 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution

(a)

(b)

Figure 6.11. HCHO distribution after 150 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA solution

Figure 6.12. Adaptive grid after 150 seconds reflecting clustering of nodes in and around the pollutant puff

The valley in the O_3 profile in Figure 6.1(a) reflects that the peak concentration of O_3 drops below the background concentration for about 50 seconds and then starts growing. The O_3 profile in Figure 6.1(b) indicates that the background concentration of O_3 remains virtually constant. These figures also reflect that at 150 seconds the peak concentration of O_3 is higher than the background. The results in Figures 6.8(a), 6.9(a), and 6.9(b) are consistent with these observations; each of these figures depicts a peak and a valley in concentration of O_3 . However, the SGA-PPM result in Figure 6.8(b) reflects that the peak O_3 concentration is below the background, thus indicating that this is an inaccurate result.

The concentrations of NO_2 and HCHO, shown in 6.1(a) and 6.1(b), do not undergo any peak inversions. Correspondingly, the results in Figures 6.6, 6.7, 6.10, and 6.11 do not exhibit any peak inversions.

The above qualitative observations reveal that some of the fine-scale solution structures for NO and O_3 are captured in the DSAGA-PPM simulation but not in the SGA-PPM simulation.

As discussed above, the species profiles resulting from chemical interactions can contain peaks and valleys. Thus, to obtain an accurate assessment of such profiles, it is important to be able to characterize these peaks and valleys. Accordingly, some of the error measures introduced in Chapter 4 were modified to provide such characterizations

and facilitate comparisons between the static and the adaptive grid solutions. These modified measures are

$$EVALLEY = \frac{c^{valley} - c_e^{valley}}{c_e^{valley}}, \quad (6-1)$$

$$EPEAK = \frac{c^{peak} - c_e^{peak}}{c_e^{peak}}, \quad (6-2)$$

In addition to the above measures, the errors in mass and variance as well as the root-mean-square error introduced in Chapter 4 are also used to characterize the quality of SGA-PPM and DSAGA-PPM solutions. For completeness, again these errors are presented below.

$$EMAS = \frac{\sum cV - \sum c_e V}{\sum c_e V}, \quad (6-3)$$

$$EVAR = \frac{\sum c^2 V - \sum c_e^2 V}{\sum c_e^2 V}, \quad (6-4)$$

$$ERMS = \sqrt{\sum \left(\frac{c - c_e}{c_e} \right)^2 V / \sum V}. \quad (6-5)$$

For the chemical species, initialized as conical concentration distributions, values of the above error measures are shown in Tables 6.3 and 6.4. In Table 6.3, the lower values of EVALLEY (%) for the DSAGA-PPM solution for NO and O₃ indicate that the adaptive grid algorithm predicts the valleys in the NO and O₃ concentrations better than the corresponding SGA solution (also see Figures 6.4, 6.5, 6.8, and 6.9).

The lower values of EPEAK (%) for the DSAGA-PPM compared to those for the SGA-PPM solutions for HC, HCHO, NO₂, and O₃ reflect that the adaptive grid predicts solution peaks better than the static grid. Note that a negative EPEAK (%) value indicates that the peak is lower than that in the corresponding reference solution. Note also that a large positive value of EVALLEY and a large negative value of EPEAK for NO in SGA-PPM solution reflect that the valley in this solution is significantly higher and the peak is significantly lower than these features in the corresponding reference solution. These findings are consistent with the fact that the valley is not as well developed in the SGA-PPM solution as it should be (see Figure 6.4(b)).

Table 6.3. EVALLEY and EPEAK for the 150 Seconds Simulation of the Puff

Species	SGA-PPM		DSAGA-PPM	
	EVALLEY (%)	EPEAK (%)	EVALLEY (%)	EPEAK (%)
HC	-	-35.0	-	-9.7
HCHO	-	-40.4	-	-11.3
NO	30.3	-18.2	6.2	-1.2
NO ₂	-	-37.5	-	-10.0
O ₃	0.6	-3.4	0.0	-1.1

Table 6.4. EMAS, EVAR, and ERMS for the 150 Seconds Simulation of the Puff

Species	SGA-PPM			DSAGA-PPM		
	EMAS (%)	EVAR	ERMS	EMAS (%)	EVAR	ERMS
HC	0.15	-1.3E-01	1.9E-01	0.77	1.5E-02	4.2E-02
HCHO	-0.16	-1.7E-01	1.9E-01	0.80	1.1E-02	4.2E-02
NO	1.10	-7.8E-02	2.4E-01	0.61	1.4E-02	4.3E-02
NO ₂	-0.09	-1.7E-01	2.0E-01	0.82	1.2E-02	4.2E-02
O ₃	0.0	-6.9E-05	1.4E-03	0.0	-2.8E-05	2.8E-04

As seen in Table 6.4, the low values of EMAS (%) for the DSAGA-PPM and SGA-PPM solutions reflect that the global mass in each of these solutions compares well with the global mass in the corresponding reference solution. Finally, the lower

magnitudes of EVAR and ERMS for the DSAGA-PPM solution reflect that, compared to the static grid solution, the adaptive grid solution undergoes less numerical diffusion and has less overall (root-mean-square) error.

The results of this test indicate that not only can DSAGA-PPM follow rapid chemical transformations but it does so more accurately than the corresponding static grid algorithm.

6.5 Simulation for 86,400 Seconds

For these simulations, settings identical to those used for 150 seconds test case were used. Thus, the domain was initially discretized with 43 x 43 grid nodes, spaced uniformly. In each of the simulations, the CFL was set at 0.8. In DSAGA-PPM simulation, the values of the SIERRA weight function related parameters e_1 , e_2 , w_{min} , and smoothing iterations, described in Section 3.7.1, were set at -1.0, 0, 1.E-05, and 10, respectively; and the value of the grid convergence parameter δ , described in Section 3.7.4, was set at 3.E-02. Additionally, in the interim step procedure described in Section 3.7.3, the movement of any cell side was restricted to one-half of the corresponding cell length. As discussed in the last section, the values of the adaptation parameters were determined through numerical experiments and were selected to obtain a good preadapted grid (see Figure 6.3).

Figures 6.13 through 6.20 present the SGA-PPM and DSAGA-PPM solutions and the corresponding reference solutions for NO, NO₂, O₃, and HCHO after one revolution of the puff. As explained below, in a qualitative sense the results in these figures are consistent with the species profiles shown in Figures 6.2(a) and (b). Also presented in Figure 6.21 is the resulting adaptive grid after 86,400 seconds with grid nodes clustered in and around the puff.

(a)

(b)

Figure 6.13. NO distribution after 86,400 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution

(a)

(b)

Figure 6.14. NO distribution after 86,400 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA-PPM solution

(a)

(b)

Figure 6.15. NO_2 distribution after 86,400 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution

(a)

(b)

Figure 6.16. NO_2 distribution after 86,400 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA-PPM solution

(a)

(b)

Figure 6.17. O_3 distribution after 86,400 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution

(a)

(b)

Figure 6.18. O_3 distribution after 86,400 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DSAGA-PPM solution

(a)

(b)

Figure 6.19. HCHO distribution after 86,400 seconds in the pollutant puff (a) exact solution on uniform grid (b) SGA-PPM solution

(a)

(b)

Figure 6.20. HCHO distribution after 86,400 seconds in the pollutant puff (a) exact solution on adaptive grid (b) DASGA-PPM solution

Figure 6.21. Adaptive grid after 86,400 seconds reflecting clustering of nodes in and around the pollutant puff

As seen in Figures 6.2(a) and 6.2(b), at 86400 seconds, the peaks of O_3 and HCHO concentrations are higher than the corresponding backgrounds; accordingly conic shapes are seen for these species in Figures 6.17 through 6.20. Again in Figures 6-2(a) and 6-2(b), at 86400 seconds, the peak concentrations of NO and NO_2 are below the corresponding background concentrations; accordingly valleys are seen in the Figures 6.13 through 6.16 for these species.

The error measures introduced in the last section were again used to provide quantitative comparisons between the static and the adaptive grid solutions. For the chemical species initialized as conical distributions, the values of these measures are shown in Tables 6.5 and 6.6.

In Table 6.5, the much lower values of EVALLEY (%) for the DSAGA-PPM solution for NO and NO_2 indicate that the adaptive grid algorithm predicts the valleys in the concentrations of these species far better than the corresponding SGA solution (also see Figures 6.13 through 6.16). Note that the minimum concentrations of NO and NO_2 are quite small, $O(E-07)$ ppb and $O(E-03)$ ppb, respectively. Such small concentrations are affected more by round-off errors and may be responsible for the relatively large EVALLEY(%) values.

The magnitudes of the DSAGA-PPM solution EPEAK (%) values for HC, HCHO, and O_3 are lower than the corresponding values in the SGA-PPM solution. This

reflects that the adaptive grid predicts solution peaks better than the static grid. Note that all of the EPEAK(%) values are negative and, therefore, the peaks in the numerical solutions are lower than the corresponding peaks in the reference solutions.

As seen in Table 6.6, the values of EMAS (%) for HC, HCHO, and O₃ in the DSAGA-PPM and SGA-PPM solutions are quite small and thus the global mass of each of these species in the numerical solutions compares well with the global mass in the corresponding reference solution. However, the EMAS(%) values in the SGA-PPM solution for NO and NO₂ are quite large thus indicating that the SGA-PPM solution is not a good representation of the corresponding reference solution.

Finally, the lower magnitudes of EVAR and ERMS for the DSAGA-PPM solution reflect that, compared to the static grid solution, the adaptive grid solution undergoes less numerical diffusion and has less overall (root-mean-square) error.

The above results indicate that DSAGA-PPM is capable of following the drastic changes in the pollutant distributions (peak inversions) more accurately than SGA-PPM.

Table 6.5. EVALLEY and EPEAK for the 86,400 Seconds Simulation of the Puff

Species	SGA-PPM		DSAGA-PPM	
	EVALLEY (%)	EPEAK (%)	EVALLEY (%)	EPEAK (%)
HC	-	-22.2	-	-11.1
HCHO	-	-37.5	-	-13.0
NO	141.5	-	28.0	-
NO ₂	55.9	-	11.7	-
O ₃	-	-16.7	-	-5.7

Table 6.6. EMAS, EVAR, and ERMS for the 86,400 Seconds Simulation of the Puff

Species	SGA-PPM			DSAGA-PPM		
	EMAS (%)	EVAR	ERMS	EMAS (%)	EVAR	ERMS
HC	-5.40	-1.1E-01	6.1E-02	-0.33	-9.3E-03	1.3E-02
HCHO	0.35	-1.5E-01	1.9E-01	0.90	1.7E-02	4.9E-02
NO	14.00	2.9E-01	1.5E-01	1.30	2.6E-02	2.2E-02
NO ₂	14.00	3.1E-01	1.5E-01	1.40	2.8E-02	1.5E-02
O ₃	0.41	7.3E-03	1.0E-02	0.06	1.4E-03	3.4E-03

7 Dispersion of A Power Plant Plume

In the previous chapter, the capability of DSAGA-PPM to calculate coupled processes of advection and chemistry was examined. However, in reality, distribution of pollutants in the atmosphere is affected by the non-linear interactions between emissions from a point source, advection, turbulent diffusion, and chemistry. Therefore, it is instructive to examine the ability of DSAGA-PPM to accurately simulate a problem with these interactions. Consequently, examined in this chapter is a model problem in which emissions from a power plant lead to formation of O_3 . Specifically, the power plant represents a point source of nitrogen oxides (NO_x), and its plume can travel in the atmosphere for hundreds of miles. As this plume moves through the atmosphere, it interacts with the background species to produce O_3 . Such a model problem has been described by Tomlin *et al.* [6] in their evaluation of an unstructured adaptive grid method.

The photochemical generation of O_3 is known to be a regional-scale problem. This is because the times scales of photochemical production of O_3 are such that the pollution episodes can occur at distances far downwind of the source. It is also well established that the photochemistry describing O_3 production is non-linear [65, 66]. This requires that the spatial distributions of the various species in the power plant plumes be described accurately, especially in locations of steep gradients (e.g., in the vicinity of sources). Consequently, in an adaptive grid simulation of photochemical O_3 production,

it would appear to be desirable to cluster grid nodes near the sources and around the power plant plumes, as these plume travels over regional distances. Accordingly, photochemical production of O_3 poses a multi-scale problem, the solution of which would appear to benefit from application of dynamic grid adaptation.

7.1 Description of the Model Problem

In the model problem, a point source (i.e., a power plant) is located at coordinates (55 km, 105 km) in a two-dimensional region that is 210 km long in each of x - and y - directions. The domain is initially discretized with 22×22 grid nodes, spaced uniformly. Thus, each of the cells in the initial uniform grid is a square with a side of 10 km. As discussed in Gillani [67], the current AQMs typically use grid cell sizes larger than 10 km $\times$ 10 km. Therefore, in the context of contemporary air quality modeling, the grid chosen for the test case would be considered to be a refined grid. The transport parameters consist of a constant wind speed of 5 m/s in the x -direction and a constant horizontal turbulent diffusivity of $100 \text{ m}^2/\text{s}$ for all species, in each of x - and y - directions. Note that the chosen value of diffusivity is consistent with the range of $50\text{-}100 \text{ m}^2/\text{s}$ provided in [21]. Gillani [67] has reported that plumes from large power plants achieve spreads of 30-40 km as they travel downwind. The chosen value of diffusivity ensured that the plume spread in this test case is consistent with Gillani's finding.

In this model problem, the simplified atmospheric chemistry described in Table 6.1 was used. As in the previous tests, the solar zenith angle, θ , that appears in the photolysis reactions of Table 6.1, was held constant at 71.5° . Note that for this simplified chemical mechanism, volatile organic compounds (VOC) are HC and HCHO. Also NO_x is simply the mixture of NO and NO_2 . Per Gillani [67], the regional background in the Eastern U.S., in the summer daytime mixing layer, is typically characterized by relatively high VOC/ NO_x ratios (~ 10 -100). Moreover, Sillman *et al.* [68] have used a NO_x concentration of 0.2 ppb in their simulations of urban and power plant plumes. Accordingly, for the model problem the background NO_x concentration and VOC/ NO_x ratio are chosen to be 0.2 ppb and 35, respectively. These values, correspond to a background VOC concentration of 7 ppb. For the cell containing the source (power plant), the VOC/ NO_x ratio, comprising of $\frac{\text{VOC}_{\text{background}}(\text{ppb}) + \text{VOC}_{\text{powerplant}}(\text{ppb})}{\text{NOx}_{\text{background}}(\text{ppb}) + \text{NOx}_{\text{powerplant}}(\text{ppb})}$, is chosen to be 0.3, which is less than the characteristic value of one for power plant plumes per Sillman *et al.* [68]. Further, Sillman *et al.* [68] have reported that in the St. Louis region in the Eastern U.S., on average, the ratio of VOC emissions to NO_x emissions from power plants is about 14%. This ratio is used in this model problem. Again, in accordance with [68], the ratios of $\frac{\text{NO}_{\text{background}}(\text{ppb})}{\text{NO2}_{\text{background}}(\text{ppb})}$, and $\frac{\text{HCHO}_{\text{background}}(\text{ppb})}{\text{VOC}_{\text{background}}(\text{ppb})}$ are assumed to be 0.1 and 0.05, respectively. Finally, ten percent of the power plant NO_x is assumed to be NO_2 , the remaining 90% being NO. With these assumptions, and using Equation (3.71), the

concentrations (*molecules/cm³*) of VOC and NO_x are shown in Table 7.1. Further, using the concentrations (*molecules/cm³*) of VOC and NO_x emissions from the source, the chosen grid and transport parameters, a CFL of 0.8, and a summer daytime mixing layer height of 1 km, the source terms (*molecules/s*) corresponding to NO_x and VOC emission rates from the power plant are calculated. These rates are shown in Table 7.1.

The initial conditions for species other than VOC and NO_x consist of the background concentrations used earlier (see Table 6.2). For completeness, the initial conditions for all species are summarized in Table 7.1. Note that in the simulation, at time = 0 seconds, the power plant starts to emit NO_x and VOC at the steady rates shown in Table 7.1.

Each of the SGA-PPM and DSAGA-PPM simulations was carried out for 40,000 seconds, i.e. about 12 hours of daytime. In these simulations the CFL was set at 0.8. In the DSAGA-PPM simulation, the values of the SIERRA weight function related parameters e_1 , e_2 , m_{in} , and number of smoothing iterations (see Section 3.7.1) were set at -1.1, 0, 1.E-01, and 18, respectively; the value of the grid convergence parameter δ , (see Section 3.7.4) was set at 1.E-02. Additionally, in the interim step procedure (see Section 3.7.3), the movement of any cell side was restricted to one-half of the corresponding cell length. As in the other test cases, the values of the adaptation parameters were determined through numerical experiments and were selected to obtain a good preadapted grid (see Section 3.7.5). This grid is shown in Figure 7.1.

Table 7.1. Initial Concentrations and Source Terms
for the Power Plant Plume Model Problem

Species	Background (molecules/cm ³)	Source (molecules/cm ³)	Source (molecules/s)
CO	1.00E+12		
H ₂ O	2.50E+15		
HC	1.64E+11	1.42E+11	8.87E+24
HCHO	8.61E+09	7.47E+09	4.67E+23
HO ₂	1.00E+06		
NO	4.47E+08	9.60E+11	6.00E+25
NO ₂	4.47E+09	1.07E+11	6.67E+24
O(1d)	1.00E-03		
O ₃	5.00E+11		
OH	1.00E+05		
RO ₂	1.00E+06

Figure 7.1. Preadapted grid reflecting clustering of nodes near the point source

7.2 Simulation for 40,000 Seconds

Shown in Figures 7.2 and 7.3 are the NO solution fields obtained using SGA-PPM and DSAGA-PPM. A comparison of these figures reveals that the DSAGA-PPM solution, as expected, is less dispersed than the corresponding SGA-PPM solution. Further, the NO contour values in the DSAGA-PPM solution are higher than those in the SGA-PPM solution. In the SGA-PPM simulation, the source emissions are dispersed over a large grid cell volume and result in lower NO contours

The regional nature of the O₃ plume can be seen in Figures 7.4, 7.5, 7.6, and 7.7. In each of these figures, starting from the source, the plume traverses regional scale distances. Further, Figures 7.4 and 7.5 depict different spatial distributions of O₃ thereby indicating that grid cell size appears to have a significant impact on the calculated O₃ field.

As seen in Figure 7.7, close to the source the concentration of O₃ is much lower than the background. In the area close to the source (*high- NO_x regime*), the chemistry is limited by the supply of odd-H radicals (OH, HO₂, and RO₂) relative to NO_x and, therefore, the oxidizing capacity of this environment is low.

Figure 7.2. SGA-PPM prediction of NO concentration after 40,000 seconds

Figure 7.3. DSAGA-PPM prediction of NO concentration after 40,000 seconds

Figure 7.4. SGA-PPM prediction of O_3 concentration after 40,000 seconds

Figure 7.5. DSAGA-PPM prediction of O_3 concentration after 40,000 seconds

Figure 7.6. Three-dimensional depiction of SGA-PPM results for O_3 concentration at 40,000 seconds

Figure 7.7. Three-dimensional depiction of DSAGA-PPM results for O_3 concentration at 40,000 seconds.

Consequently, O_3 is quickly depleted by the high levels of NO_x present near the source, according to the reaction:

As the O_3 plume travels downstream, the NO_x concentration decreases and turbulent diffusion causes the plume to gradually mix with the background HC and HCHO. Consequently, the supply of odd-H radicals, relative to NO_x , increases. At some point in time, the chemistry in the plume shifts to *low NO_x regime* in which the supply of odd-H radicals, relative to NO_x , becomes large enough to cause an oxidizing environment in which NO is oxidized to NO_2 by the odd-H radicals. This oxidation of NO competes with the reaction shown by (7-1) for the availability of NO and results in reduced destruction of O_3 . The concentrations of O_3 , therefore, rise above the background levels at quite large distances downwind of the source. This behavior is exhibited by the DSAGA-PPM solution presented in Figure 7.7 but not by the SGA-PPM solution shown in Figure 7.6. In the SGA-PPM simulation, power plant emissions are instantaneously dispersed over a large grid volume and their chemistry is artificially shifted from that of the high- NO_x regime to low- NO_x regime in a sudden fashion. Consequently, the oxidation of NO to NO_2 and production of O_3 occurs near the source.

Presented in Figures 7.8(a) through 7.8(c) are the SGA-PPM and DSAGA-PPM O_3 concentrations in the plume cross-sections at distances of 10 km, 30 km, and 135 km

from the source. The DSAGA-PPM O_3 concentrations in these three plots represent three general stages of plume chemistry, denoted as the “early”, the “intermediate”, and the “mature” stages [67].

In the “early” stage shown in Figure 7.8(a), the plume chemistry is in the high NO_x regime described above and, consequently, O_3 is depleted. As the plume travels downwind, its edges receive NO_x by diffusion from the plume core and VOC and odd-H by entrainment from the background. These conditions become favorable to generation of odd-H and, consequently, formation of O_3 . Thus O_3 bulges develop at the plume edges. This is shown in the “intermediate” stage shown in Figure 7.8(b). Finally, in the “mature” stage shown in Figure 7.8(c), the entire plume volume is highly oxidizing and substantial excess of O_3 over background occurs. Such stages were observed in the measurements made on the Labadie Power plant plume, as noted in [67]. Note that the SGA-PPM O_3 concentrations in the plume shown in these figures do not exhibit the expected near-source plume structure. This reflects that the SGA-PPM application is unable to resolve the delicate and non-linear nature of NO_x and odd-H chemistry.

Figure 7.8(a). SGA-PPM and DSAGA-PPM predictions of O_3 concentration in plume cross-section at a distance of 10 km downwind of the source. SGA concentrations are depicted by the dotted curve.

Figure 7.8(b). SGA-PPM and DSAGA-PPM predictions of O_3 concentration in plume cross-section at a distance of 30 km downwind of the source. SGA concentrations are depicted by the dotted curve.

Figure 7.8(c). SGA-PPM and DSAGA-PPM predictions of O_3 concentration in plume cross-section at a distance of 135 km downwind of the source. SGA concentrations are depicted by the dotted curve.

Presented in Figures 7.9 and 7.10 are the DSAGA-PPM solution fields for O_3 at 3600 seconds and 10,800 seconds. The results in these figures depict the evolution of the O_3 solution field in time. At 3600 seconds, the O_3 valley near the source is much deeper than that at 10,800 seconds reflecting the initial rapid depletion of background O_3 concentration. This consumption of O_3 occurs via production of NO_2 (see Reaction (7-1)) which, in turn, leads to some O_3 production through photochemistry (see, for e.g., Figures 6.1 and 6.2). Thus, in the simulation some O_3 is produced at the source cell and some is transported to the source cell by turbulent diffusion. As the simulation progresses in time, O_3 concentration in the source cell, as well as other cells, increases due to photochemistry (see, for e.g., Figures 6.1 and 6.2). Consequently, the O_3 valley near the source becomes shallower in time. These results reflect that DSAGA-PPM is able to resolve the varying temporal scales of O_3 solution field through dynamic grid adaptation.

Shown in Figure 7.11 is the grid resulting from DSAGA-PPM simulation for 40,000 seconds. As seen in this figure, the grid nodes are clustered near the source and in the plume. Thus the grid appears to have responded to the resolution needs of the solution field.

Figure 7.9. DSAGA-PPM prediction of O_3 concentration field at 3600 seconds

Figure 7.10. DSAGA-PPM prediction of O_3 concentration field at 10,800 seconds

Figure 7.11. Adaptive grid after 40,000 seconds reflecting clustering of nodes near the source and in the plume

In order to examine mass conservation capability of DSAGA-PPM, mass-balance of N atoms was computed in each of the SGA-PPM and DSAGA-PPM simulations. The values of this balance are 99.97 percent for SGA-PPM and 99.96 percent for DSAGA-PPM. These results reflect that both SGA-PPM and DSAGA-PPM are able to maintain mass with a high degree of accuracy.

7.3 Assessment of the Accuracy of DSAGA-PPM Predictions

It is important to assure that the DSAGA-PPM results obtained for this model problem are a reasonably accurate description of reality. The fact that near-source plume structure captured by the DSAGA-PPM solution is consistent with the measured findings is reassuring. A comparison of this solution with a more accurate SGA-PPM solution may provide further assessment of its accuracy. Accordingly, a refined-grid SGA-PPM solution for 40,000 seconds, utilizing 64 x 64 grid nodes, was obtained for this model problem. Shown in Figure 7.12 is the O₃ field resulting from the refined grid simulation. The O₃ valley in the vicinity of the source seen in this figure is consistent with the DSAGA-PPM result shown in Figure 7.7. Presented in Figures 7.13(a) through 7.13(c) are the O₃ concentrations in the plume cross-sections at distances of 10 km, 30 km, and 135 km from the source, obtained using refined grid SGA-PPM and DSAGA-PPM. As seen in these plots, the refined grid SGA-PPM solutions reflect the “early”, the “intermediate”, and the “mature” stages of plume development captured by the DSAGA-

PPM solutions. These results indicate that the more accurate SGA-PPM solution obtained using a refined grid is similar to DSAGA-PPM solution and, therefore, the adaptive grid solution is a reasonably accurate depiction of reality. Note that at the end of the DSAGA-PPM simulation, the minimum cell size was 7.3 km^2 . This cell size is smaller than the cell size of 11.1 km^2 used in the refined grid SGA-PPM simulation. This indicates that the solution field obtained in the refined grid SGA-PPM simulation is still not as resolved as that obtained in the DSAGA-PPM simulation.

The results of this test case reflect that DSAGA-PPM is able to resolve the multiple spatial and temporal scales associated with O_3 production and transport. Consequently, DSAGA-PPM is able to capture not only the regional O_3 plume distribution but also the small-scale plume structure near a source. Additionally, DSAGA-PPM appears to maintain mass with good accuracy.

Maximum = 89.73 ppb

Minimum = 19.21 ppb

Background = 31.40 ppb

Figure 7.12. Refined grid SGA-PPM prediction of O_3 concentration field at 40,000 seconds

Figure 7.13(a). Refined grid SGA-PPM and DSAGA-PPM predictions of O_3 concentration in plume cross-section at a distance of 10 km downwind of the source. Refined grid SGA concentrations are depicted by the dotted curve.

Figure 7.13(b). Refined grid SGA-PPM and DSAGA-PPM predictions of O_3 concentration in plume cross-section at a distance of 30 km downwind of the source. Refined grid SGA concentrations are depicted by the dotted curve.

Figure 7.13(c). Refined grid SGA-PPM and DSAGA-PPM predictions of O_3 concentration in plume cross-section at a distance of 135 km downwind of the source. Refined grid SGA concentrations are depicted by the dotted curve.

8 Simulation with Multiple Sources

Air pollution is generated as a result of complex, dynamic, physical and chemical interactions among emissions from various sources. Thus, it is desirable that an AQM be able to faithfully follow these interactions and produce accurate solution fields. Consequently, the final model problem was designed to examine the ability of DSAGA-PPM to accurately calculate solution fields resulting from such interactions. Specifically, in this problem, the O_3 production resulting from dynamic interactions between emissions from two point sources and an area source is followed in time. As discussed in the context of the previous model problem, photochemical production of O_3 poses a multi-scale problem, the solution of which would appear to benefit from application of dynamic grid adaptation.

8.1 Description of the Model Problem

In this model problem, two identical point sources (i.e., power plants) are located at coordinates (139.5 km, 139.5 km) and (319.5 km, 319.5 km) in a two-dimensional domain that is 459 km long in each of x - and y -directions. In addition, a circular area source (i.e., an urban area) is located at the center of the domain. The area source is divided into two regions. The *inner* region is circular with center at coordinates (229.5 km, 229.5 km) and has a radius of 27 km. This region represents an area with a relatively

high level of anthropogenic activity. The *outer* region is annular in shape with the same center as the first region and inner and outer radii of 27 km and 36 km, respectively. This region represents the periphery of the urban area. The domain is initially discretized with 52 x 52 grid nodes, spaced uniformly. Thus, each of the cells in the initial uniform grid is a square with a side of 9 km. In the context of contemporary air quality modeling, this grid would be considered to be a refined grid. The transport parameters consist of a constant wind speed of 5 m/s in the North-Easterly direction and a constant horizontal turbulent diffusivity of 100 m²/s for all species, in each of *x*- and *y*-directions. This value of turbulent diffusivity is identical to that used in the last test case and is consistent with the range provided in [21].

As in the previous model problems, the simplified atmospheric chemistry described in Table 6-1, along with a constant solar zenith angle θ of 71.5 °, is used in this model problem. The background NO_x concentration and VOC/ NO_x ratio are chosen to be 0.2 ppb and 10, respectively. These values, correspond to a background VOC concentration of 2 ppb. Note that for the simplified chemistry mechanism used in this model problem, VOC represents a mixture of HCHO and HC. As in the last model problem, the ratios of $\frac{NO_{background}(ppb)}{NO_{2background}(ppb)}$, and $\frac{HCHO_{background}(ppb)}{VOC_{background}(ppb)}$ are assumed to be 0.1 and 0.05, respectively. Using these values, the concentrations (*molecules/cm*³) of background VOC and NO_x are calculated. These values are shown in Table 8.1.

For the cells containing the point sources (power plants), the VOC/NO_x ratio $\frac{VOC_{background}(ppb) + VOC_{powerplant}(ppb)}{NOx_{background}(ppb) + NOx_{powerplant}(ppb)}$ is chosen to be 0.3, i.e., a value less than one characteristic of power plant plumes per Sillman *et al.* [68]. Other assumptions regarding emissions from power plants are the same as those used in the last test. Thus, the ratio of VOC emissions to NO_x emissions from power plants is taken to be 14% and ten percent of the power plant NO_x is assumed to be NO₂ with the remaining 90% being NO. Using Equation (3.71) along with these assumptions and background values of NO_x and VOC, the concentrations (*molecules/cm³*) of VOC and NO_x emissions from each of the two identical point sources are calculated. Using these concentrations along with the chosen grid and transport parameters, a CFL of 0.8, and a summer daytime mixing layer height of 1 km, Equation (3.62) provides the emission rates (*molecules/s*) for each of the power plants. These emission rates are shown in Table 8.1.

According to Sillman *et al.* [68], urban areas typically have high emission rates for both hydrocarbons and NO_x, with hydrocarbon to NO_x emission ratios of about 5:1 (carbon atoms : nitrogen atoms). Accordingly, the cells in the inner and outer regions of the urban area are assigned VOC/ NO_x emission ratios of 5 and 9, respectively. The cells in the inner region have higher VOC and NO_x emissions to reflect a higher density of traffic flow and other anthropogenic activity. The emission fluxes (*molecules/cm²-s*) corresponding to the above ratios for each of the urban regions are shown in Table 8.1.

The initial conditions for species other than VOC and NO_x consist of the background concentrations used earlier (see Table 6-2). Note that in the simulation, at time = 0 seconds, the sources start to emit NO_x and VOC at the steady rates shown in Table 8.1.

Table 8.1. VOC and NO_x Initial Concentrations and Source Terms for the Multiple Sources Model Problem

Species	Background (molecules per cm³)	Point Sources (molecules/s)	Inner Region (molecules/cm²-s)	Outer Region (molecules/cm²-s)
HC	4.67E+10	1.58E+24	1.72E+12	1.10E+11
HCHO	2.46E+09	8.30E+22	9.06E+10	6.00E+09
NO	4.47E+08	1.07E+25	5.49E+10	3.70E+09
NO ₂	4.47E+09	1.19E+24	5.49E+11	3.70E+10

8.2 Simulation for 40,000 Seconds

Each of the SGA-PPM and DSAGA-PPM simulations was carried out for 40,000 seconds, i.e., about 12 hours of daytime. In these simulations; the CFL was set at 0.8. Further in the simulation using DSAGA-PPM, the values of the SIERRA weight function related parameters e_1 , e_2 , w_{min} , and smoothing iterations, described in Section 3.7.1, were set at -1.1, 0, 7.E-02, and 16, respectively; and the value of the grid convergence parameter δ , described in Section 3.7.4, was set at 1.7E-01. As in the other simulations, these values were determined through numerical experiments and were selected to obtain good preadaptation (see Section 3.7.5). The preadapted grid is shown in Figure 8.1. Note also that in the DSAGA-PPM simulation, the movement of any cell side was restricted to one-half of the corresponding cell length in the interim step procedure described in Section 3.7.3.

Shown in Figures 8.2 and 8.3 are the predictions of NO. A comparison of these figures reveals that the NO contour values in the DSAGA-PPM solution are higher than those in the SGA-PPM solution, especially in the vicinity of the sources. Application of DSAGA-PPM reduces sizes of the two cells with point sources. Consequently, errors due to averaging of point source emissions over large grid cells are reduced. This results in relatively higher NO concentrations near these sources. Further, in the DSAGA-PPM simulation, relatively higher NO emissions from the upwind point source transport into

the urban area. These transported emissions react with the available odd-H radicals (OH , HO_2 , and RO_2) to form NO_2 and subsequently O_3 .

The regional nature of the O_3 plume, resulting from the emissions of the three sources, is evident in Figures 8.4 through 8.7. In each of these figures, the two peaks reflect the contributions from the upwind and the downwind point sources. Moreover, the absence of an O_3 peak near the upwind point source in these figures reflects a relatively mature O_3 chemistry near this source. It is also worth noting that the distribution of O_3 resulting from the interaction of the upwind source plume with the urban area plume is more defined in the DSAGA-PPM solution shown in Figure 8.5 than in the corresponding SGA-PPM solution shown in Figure 8.4.

The Figures 8.4 through 8.7 reflect that the O_3 concentrations in the upwind point source plume and in the vicinity of the urban area are higher in the DSAGA-PPM solution. Further, as seen in Figure 8.7, the O_3 peak near the urban area (the first O_3 peak) is more pronounced in the DSAGA-PPM solution. The O_3 levels in the vicinity of the urban area are affected by the transport of O_3 and NO from the upwind point source and by the O_3 chemistry in the urban area. The higher levels of transported O_3 along with more efficient O_3 production due to higher levels of NO [66] appear to cause the higher first O_3 peak seen in the DSAGA-PPM solution.

Figure 8.1. Preadapted grid reflecting clustering of nodes near the point sources and in and around the urban area source

Figure 8.2. SGA-PPM prediction of NO concentration after 40,000 seconds

Figure 8.3. DSAGA-PPM prediction of NO concentration after 40,000 seconds

Figure 8.4. SGA-PPM prediction of O_3 concentration after 40,000 seconds

Figure 8.5. DSAGA-PPM prediction of O_3 concentration after 40,000 seconds

Figure 8.6. Three -dimensional depiction of the O₃ concentration field resulting from the SGA-PPM simulation for 40,000 seconds

Figure 8.7. Three -dimensional depiction of the O₃ concentration field resulting from the DSAGA-PPM simulation for 40,000 seconds

The NO and O₃ concentrations, downwind of the first O₃ peak, were generated in the urban area before the emissions from the upwind source reached this area. As seen in Figures 8.4 through 8.7, these concentrations are of similar magnitudes in the SGA-PPM and the DSAGA-PPM solutions. Unlike the point source emissions, input as mass rates (molecules/s), the urban area emissions are input as mass fluxes (molecules/cm²-s). Thus, whereas emissions from a point source emissions are averaged over the pertinent cell volume *within* an AQM, urban area emissions are averaged over an area (or a number of cells) *outside* the AQM (or in the preprocessing of input data). Consequently, in a simulation, emissions from a point source can undergo artificial dilution due to averaging over a relatively large cell size. Urban area emissions within a simulation, however, would not suffer from these errors. Therefore, clustering of grid nodes around a point source reduces dilution of emissions but clustering in and around an urban area may reduce numerical diffusion and dispersion only. Since chemical activity depends on the local concentrations of species, diluted point source emissions could lead to erroneous chemistry results. In contrast, if changes in urban concentrations due to chemistry dominate over changes due to numerical inaccuracies, urban area concentration fields would be relatively independent of grid cell size. The similarity of the SGA-PPM and the DSAGA-PPM results for O₃ concentrations in the urban plume, downwind of the first O₃ peak, is attributed to the above effects.

Further downwind, the concentrations in the urban plume mix with emissions from the downwind point source and result in the second O₃ peak. This peak has similar

magnitudes in the SGA-PPM and the DSAGA-PPM solutions (Figures 8.6 and 8.7). As noted above, the concentrations in the urban plume, downwind of the first O_3 peak, are of similar magnitudes in the SGA-PPM and the DSAGA-PPM solutions. Similarity in magnitudes of the second O_3 peak in the SGA-PPM and the DSAGA-PPM solutions reflects that these concentrations dominate the chemical activity in the vicinity of the second point source.

Shown in Figure 8.8 is the converged grid obtained using DSAGA-PPM. Note that the nodes in this grid are clustered around the upwind point source and in and around the urban area. Downwind of the urban area the nodes are distributed to provide greater density of cells in the plume. It is interesting to note that there is no grid node clustering in the vicinity of the downwind point source. To understand the grid behavior, it is useful to examine the species specific weight functions (see Equation (5.5)) and the final composite weight function resulting after smoothing. Shown in Figures 8.9(a), 8.9(b), and 8.9(c) are the weight functions for OH, HO_2 , and O_3 at the end of the simulation (i.e., 40,000 seconds). The weight function for OH has an isolated slender peak at the downwind point source location. On the other hand, the weight function for HO_2 reflects a plume like distribution near this location. Finally, the weight function for O_3 has both a slender peak and a plume like distribution near the downwind point source location. Weight functions similar to that for O_3 were found for O^1D , RO_2 , HCHO, and CO. Isolated slender peaks at the downwind point source location were seen in the weight functions for NO_2 , and HNO_3 . How these weight functions might result may be

understood by the explanation of the evolution of O_3 , provided later in this chapter. The weight function formed from a linear combination of the species-specific weights (see Equations (5.6) through (5.10)) would still reflect a slender peak at the downwind source location. However, consequent smoothing may “round” or average out this slender peak. Shown in Figure 8.9(d) is the final weight function obtained after smoothing. As explained above, this weight function does not have a peak at the downwind point source location and, therefore, no grid clustering occurs near this location.

In order to examine the mass conservation capability of DSAGA-PPM, mass-balance of N atoms was computed in each of SGA-PPM and DSAGA-PPM simulations. The values of this balance are 99.94 for SGA-PPM and 99.58 for DSAGA-PPM. These results reflect that both SGA-PPM and DSAGA-PPM are able to maintain mass with a high degree of accuracy.

Figure 8.8. Adaptive grid after 40,000 seconds reflecting clustering of nodes around the upwind point source and in and around the urban area

Figure 8.9(a). OH weight function at 40,000 seconds

Figure 8.9(b). HO₂ weight function at 40,000 seconds

Figure 8.9(c). O_3 weight function at 40,000 seconds

Figure 8.9(d). Final smoothed weight function at 40,000 seconds

8.3 Simulation Without the Upwind Point Source

In an attempt to examine the effect of the upwind point source emissions on the urban area, SGA-PPM and DSAGA-PPM simulations without the upwind point source were completed. The resulting O_3 fields after 40,000 seconds are shown in Figures 8.10 and 8.11. A comparison of the SGA-PPM and the DSAGA-PPM results obtained with and without the upwind source (Figures 8.4 versus 8.10 and 8.5 versus 8.11) reveals that elimination of the upwind source removes the first O_3 peak. This is explained by the fact that the removal of the upwind source results in elimination of transported O_3 and NO emissions into the urban area and leads to reduced generation of O_3 . For the reasons explained before, the O_3 fields upwind of the second point source are similar in the SGA-PPM and the DSAGA-PPM results shown in Figures 8.10 and 8.11, respectively. However, the second O_3 peak is higher in the DSAGA-PPM solution without the source (Figures 8.11) than with the source (Figure 8.5).

Shown in Table 8.2 are the maximum and minimum cell volumes in a circular area of radius 27 km centered at the downwind point source, attained in the DSAGA-PPM simulations with and without the upwind point source. These cell volumes occur at the end of the simulation periods of 40,000 seconds each.

Table 8.2. Cell Volumes in the Vicinity of the Downwind Point Source

Simulation	Maximum Cell Size (km²)	Minimum Cell Size (km²)
DSAGA-PPM with Upwind point source	75.1	35.3
DSAGA-PPM without Upwind point source	42.1	19.6

As seen in the above table the cell sizes in the vicinity of the downwind point source are smaller in the DSAGA-PPM simulation without the upwind point source. This is explained by the fact that removal of the upwind point source results in more grid nodes available for resolving the emissions from the downwind point source. This, in turn, further reduces the dilution of emissions from this source. Consequently, the resulting higher NO_x emissions cause the elevated O₃ peak seen in Figure 8.11. Similar effect is not seen in the SGA-PPM solutions since in these simulations the resolution of the downwind point source emissions does not change.

Figure 8.10. SGA-PPM prediction of O₃ concentration field at 40,000 seconds in simulation without the upwind point source

Figure 8.11. DSAGA-PPM prediction of O₃ concentration field at 40,000 seconds in simulation without the upwind point source

8.4 Evolution of the Ozone Field

It is informative to follow the changes in chemistry that result in production of O₃. As discussed in the last chapter, odd-H radicals play a major role in the photochemical formation of O₃. The radical OH oxidizes CO and HC to generate HO₂ and RO₂. The radicals HO₂ and RO₂ react with NO to form NO₂, and NO₂ undergoes photochemistry to form O₃. Per Sillman *et al.* [66], the rate limiting steps in the production of O₃ are the oxidation of CO and HC and the key process that results in O₃ is

Therefore, concentrations of OH, HO₂, NO, and O₃ were examined at various times during the DSAGA-PPM simulation to follow the dynamic changes in chemistry.

Presented in Figures 8.12 through 8.15 are the DSAGA-PPM solution fields at 3600 seconds for NO, OH, HO₂, and O₃, respectively. The NO peaks at the point sources reflect the balance between emissions of point sources and consumption according to Reaction (8-1). Note that the NO emissions in the urban area being 10% of NO_x are quite small and are consumed per (8-1); thus, the remaining NO emissions are not apparent in the scale of Figure 8.12.

Figure 8.12. DSAGA-PPM prediction of NO concentration field at 3,600 seconds

Figure 8.13. DSAGA-PPM prediction of OH concentration field at 3,600 seconds

Figure 8.14. DSAGA-PPM prediction of HO_2 concentration field at 3,600 seconds

Figure 8.15. DSAGA-PPM prediction of O_3 concentration field at 3,600 seconds

The peaks in OH reflect production through

and

The valleys in HO₂ at the point sources reflect consumption per (8-1). However, in the urban area with plentiful HCHO, production of HO₂ via photochemical oxidation of HCHO, given by

dominates over consumption per (8-1) and results in a peak. The O₃ peaks downwind of the point sources and in the urban area reflect production through photochemical conversion of NO₂ in low NO_x regimes. At the point sources, in the presence of high NO_x concentrations, background O₃ is consumed according to

resulting in O_3 valleys. These valleys are evident in Figure 8.15. Note that at this early stage in the simulation, the O_3 plumes from the three sources are distinct. The adaptive grid at this stage of the simulation is shown in Figure 8.16. This grid depicts clustering of nodes at the point sources and in and around the urban area.

Presented in Figures 8.17 through 8.20 are the DSAGA-PPM solution fields at 25,200 seconds for NO, OH, HO_2 , and O_3 , respectively. An NO peak smaller than the one at 3600 seconds is evident at the upwind point source and no peak is seen at the downwind point source. Also noticeable is the downwind transport of NO emissions from the upwind point source. The NO profile near the upwind point source is consistent with the growth in OH (per Reactions (8-2) and (8-3)) and the valleys in HO_2 (per Reaction (8-1)) seen in Figures 8.18 and 8.19, respectively, in the region upstream of the urban area. These results reflect growth of the O_3 plume from the upwind point source (compare Figures 8.15 and 8.20).

Figure 8.16. Adaptive grid at 3,600 seconds reflecting clustering of nodes at the point sources and in and around the urban area

Figure 8.17. DSAGA-PPM prediction of NO concentration field at 25,200 seconds

Figure 8.18. DSAGA-PPM prediction of OH concentration field at 25,200 seconds

Figure 8.19. DSAGA-PPM prediction of HO_2 concentration field at 25,200 seconds

Figure 8.20. DSAGA-PPM prediction of O_3 concentration field at 25,200 seconds

Note that at this stage in the simulation, the plumes from the upwind point source, the urban area, and the downwind point source are in contact (see Figure 8.20). At the upwind end of the urban area plume, NO from the upwind point source has reacted with HO₂ (per Reaction (8-1)) in the urban area plume to generate the middle O₃ peak and the corresponding HO₂ valley seen in Figures 8.20 and 8.19, respectively. In accordance with Reactions (8-2) and (8-3), this peak in O₃ causes the corresponding peak in OH at the upwind end of the urban plume, seen in Figure 8.18.

The interaction between the downwind end of the urban plume and the downwind point source causes the peaks in OH and O₃ and the tiny valley in HO₂, seen in Figures 8.18, 8.20, and 8.19, respectively. Note that the urban plume supplies most of the HO₂ required for formation of O₃ and only a small amount is consumed from the background, thus explaining the tiny valley in HO₂. Also note that in the presence of plentiful HO₂, most of NO at the downwind point source is consumed and, therefore, no NO peak is evident in Figure 8.17 near this source. Finally, note that no O₃ valleys exist near the point source locations at this stage in the simulation; thus reflecting a relatively mature O₃ chemistry at these sources.

The adaptive grid at this stage of simulation shown in Figure 8.21 depicts clustering of grid nodes at the upwind point source and in and around the urban area. Downwind of the urban area the nodes are distributed to provide greater density of cells

in the plume. For the reasons given before, the grid nodes do not cluster around the downwind point source location.

Figure 8.21. Adaptive grid at 25,200 seconds reflecting clustering of nodes around the upwind point source and in and around the urban area

Finally, presented in Figures 8.22 through 8.25 are the DSAGA-PPM solution fields for NO, OH, HO₂, and O₃, respectively, at 39,600 seconds. At this stage in the simulation, the O₃ peaks resulting from the interactions of the plumes from the upwind point source, the urban area, and the downwind point source have grown. The results for OH and HO₂ are consistent with the O₃ solution field. Also the NO concentrations near the upwind point source are lower, reflecting more mature O₃ chemistry. The adaptive grid at this stage of simulation, shown in Figure 8.26, is similar to the one at 25,200 seconds.

Figure 8.22. DSAGA-PPM prediction of NO concentration field at 39,600 seconds

Figure 8.23. DSAGA-PPM prediction of OH concentration field at 39,600 seconds

Figure 8.24. DSAGA-PPM prediction of HO_2 concentration field at 39,600 seconds

Figure 8.25. DSAGA-PPM prediction of O_3 concentration field at 39,600 seconds

Figure 8.26. Adaptive grid at 39,600 seconds reflecting clustering of nodes around the upwind point source and in and around the urban area

8.5 Assessment of the Accuracy of DSAGA-PPM Predictions

It is important to assure that the DSAGA-PPM results obtained in this model problem are a reasonably accurate description of reality. This assessment may be made by comparing the DSAGA-PPM solution with a more accurate SGA-PPM solution. Accordingly, a refined-grid SGA-PPM solution for 40,000 seconds, utilizing 92 x 92 grid nodes, was obtained for this model problem. Shown in Figure 8.27 and 8.28 are the NO and O₃ fields resulting from the refined grid simulation. Comparison of these figures with the SGA-PPM results shown in Figures 8.2 and 8.4, reveals that NO and O₃ concentrations in the plume from the upwind point source and in the vicinity of the urban area are higher in the more accurate refined grid results.

To facilitate a quantitative comparison of O₃ results obtained in SGA-PPM, DSAGA-PPM, and refined grid simulations, peak O₃ concentrations were identified in: (1) a circular area of radius 27 km, centered at the upwind point source, and (2) in the urban area. In addition, the average O₃ concentration in a circular area of radius 27 km, centered at the downwind point source, was also calculated for each of the O₃ fields. These O₃ metrics are tabulated in Table 8.3.

Table 8.3. O₃ Metrics for Each of the Simulation Results

Simulation	Peak O₃ near the Upwind Point Source (ppb)	Peak O₃ in the Urban Area (ppb)	Average O₃ near the Downwind Point Source (ppb)
SGA-PPM	37	57	53
Refined Grid SGA-PPM	39	62	56
DSAGA-PPM	45	63	57

As seen in Table 8.3, each of the O₃ metrics increases in value as solution resolution is increased from SGA-PPM to refined grid SGA-PPM to DSAGA-PPM. This reflects that grid refinement causes changes in the O₃ solution field in a direction consistent with reaching the DSAGA-PPM solution. This trend, in turn, reflects that the adaptive grid solution does represent a reasonably accurate depiction of reality.

The simulations of this model problem reflect that the distribution of O₃ concentration resulting from dynamic interactions amongst plumes from various sources may not be accurately predicted unless a sufficiently refined grid is used. DSAGA-PPM is able to follow and resolve these interactions while maintaining mass with good accuracy. Therefore, DSAGA-PPM offers an attractive approach for obtaining accurate solutions in air quality simulations.

Figure 8.27. Refined grid SGA-PPM prediction of NO concentration field at 40,000 seconds

Figure 8.28. Refined grid SGA-PPM prediction of O₃ concentration field at 40,000 seconds

9 Computational Performance

As discussed in the previous chapters, a solution with a specific level of accuracy can be obtained using either DSAGA - PPM with relatively few grid nodes or a refined grid SGA-PPM with many more nodes. However, to assess the practicality of using DSAGA-PPM in air quality simulations, it is important to characterize its computational performance.

A projection of the performance that DSAGA-PPM may potentially achieve in air quality simulations can be made as follows. Shown in Table 9.1 is the CPU time associated with the DSAGA-PPM simulation of the single rotating cone model problem discussed in Chapter 4. Also shown are results for an additional simulation in which a refined grid (203 x 203 nodes) SGA-PPM was used. This refined grid was chosen because it gives approximately the same resolution of concentration gradients for the single cone model problem as that achieved by DSAGA-PPM applied to a 43 x 43 nodes grid. Note that both of these simulations were conducted on a CRAY Y-MP T90 machine and the code used for the two-dimensional advection experiments performs floating point operations at ~ 300 Mflops for an average vector length of about 35 units.

Table 9.1. CPU Times Associated With Rotating Cone Simulations

Simulation	Grid Nodes	CFL	CPU Time (Seconds)
Cone / DSAGA-PPM	43 x 43	0.8	19.35
Cone / SGA-PPM	203 x 203	0.8	15.55

It is well known that the chemistry calculations consume the majority of the CPU time used in an air quality simulation. The time associated with these calculations depends on the number of species and reactions modeled and the number of cells in the domain under consideration. According to Peters *et al.* [69], the integration of the chemistry rate equations at each grid point within the simulation region can consume as much as 98 percent of the total CPU time. Consider a static grid AQM utilizing SGA-PPM in which 5% of the run time is spent in computing transport and 95% in computing chemistry. Choosing the 203 x 203 static grid solution as a reference, assign 5 units to transport and 95 units to chemistry, adding up to a total run time of 100 units. As seen in Table 9.1, the time required by DSAGA-PPM for obtaining transport solution of equivalent accuracy on the 43 x 43 grid is approximately 1.24 times the time required on the 203 x 203 mesh. Thus, in an air quality simulation, a DSAGA-PPM application on a 43 x 43 grid will require about 6.2 units of time for transport. The 43 x 43 grid has 23 times fewer grid cells than the 203 x 203 grid and, therefore, will require 23 times less computational effort related to chemistry. Consequently, a DSAGA-PPM application on

the 43 x 43 grid will require chemistry work of 4.1 (95/23) units compared to chemistry work of 95 units needed in an SGA-PPM application on the 203 x 203 grid. Thus, for comparable resolution of concentration gradients, a DSAGA-PPM application will need a total of 10.3 units of time compared to 100 units for the corresponding refined grid SGA-PPM application. Finally, on the basis of this simple example, an efficiency advantage of ~ 9.7 would be expected by using DSAGA-PPM.

Further assessment of DSAGA-PPM performance may be made by examining the simulations of the multiple sources model problem described in Chapter 8. This model problem is of practical relevance; therefore an assessment of DSAGA-PPM's performance with this problem is useful. Shown in Table 9.2 are the CPU times associated with these simulations. Note that all of the model problem simulations were conducted on a CRAY Y-MP T90 machine and the code used performs floating point operations at ~ 150 million floating-point operations per second (Mflops) for an average vector length of about 50 units.

Table 9.2. CPU Times Associated With Simulations of Multiple Sources

Simulation	Grid Nodes	CFL	Minimum Cell Size (km²)	CPU Time (Seconds)
Multiple Sources using SGA-PPM	52 x 52	0.8	81	499.86
Multiple Sources using DSAGA-PPM	52 x 52	0.8	2.9	851.72
Multiple Sources using Refined grid SGA-PPM	92 x 92	0.8	25.4	1125.63

As shown in Table 9.2, the cell size in each of the static grid simulations (SGA-PPM or refined grid SGA-PPM) is substantially larger than the smallest cell size in the DSAGA-PPM simulation. This indicates that the extent of resolution of spatial distributions of species is much greater in the DSAGA-PPM simulation. Consequently, the grid in the refined grid SGA-PPM simulation would have to be refined further to obtain a solution with accuracy comparable to that achieved in the DSAGA-PPM simulation. Assuming that accuracy of a solution is proportional to the smallest cell size, an SGA-PPM simulation using about 256 x 256 grid nodes would be needed to provide a solution of comparable accuracy. Note that in the DSAGA-PPM simulation shown in Table 9.1, the smallest cell size changed with time and ranged between 1.4 to 2.9 km²; however, conservatively a value of 2.9 km² is assumed.

In the simulations represented in Table 9.2, about 4% of the run time was spent in computing transport and 96% in computing chemistry. Assume that this proportion of times remains about the same in other simulations and that time spent in calculating chemistry is proportional to the number of cells used. Then based on the CPU time associated with the 92 x 92 static grid simulation, about 8648 seconds will be spent in a simulation with a 256 x 256 static grid. Thus, for comparable resolution of concentration distributions, a DSAGA-PPM application will need a total of 851.72 seconds of time compared to 8648 seconds for the corresponding SGA-PPM solution. Finally, based on this simple analysis, an efficiency advantage of ~ 10.2 would be expected by using DSAGA-PPM. Note that this estimate of DSAGA-PPM performance is consistent with the estimate obtained using results of Table 9.1.

Based on the above discussion, DSAGA-PPM appears to offer a potential to provide accurate air quality modeling solutions at costs that may be 10 times less than those incurred in obtaining equivalent solutions with a static grid algorithm. However, a more detailed assessment of DSAGA-PPM's performance would have to be deduced from realistic air quality simulations.

10 Summary and Conclusions

A new dynamic adaptive grid algorithm for air quality modeling has been developed. This algorithm, called the *Dynamic Solution Adaptive Grid Algorithm-PPM* (DSAGA-PPM), distributes the existing grid nodes in response to spatial resolution requirements of the solution field and then updates the solution field on the resulting grid geometry. As shown in this work, this process of grid adaptation and solution update results in a more accurate solution than that achieved using a static grid with the same number of nodes.

In general, grid adaptation results in a non-orthogonal, curvilinear, structured grid. The governing equations of pollutant transport in the atmosphere (i.e., the atmospheric diffusion equation [17]) are solved on this grid. This is accomplished by transforming the governing equations, initially expressed in Cartesian space, to a computational domain expressed in general curvilinear coordinates (ξ, η, ζ) . Once the governing equations are expressed in this domain, the mapping to any coordinate system of interest in air quality modeling can be obtained by specifying appropriate values of the metric derivatives of the transformation. The generality of this mapping is demonstrated analytically.

The atmospheric diffusion equation represents a system of stiff partial differential equations since the time scales associated with typical chemical transformations are far

smaller than those associated with transport. Therefore, in numerical formulations of the current air quality models, the atmospheric diffusion equation is time-split to facilitate efficient computation of transport. In this work, time-split expressions in general curvilinear coordinates were developed for the transport and chemistry processes included in the atmospheric diffusion equation.

In order to compute the advection process accurately in an air quality simulation, a higher order scheme - *the Piecewise Parabolic Method (PPM)* [16] - is used to obtain the values of the fluxes at the cell surfaces in DSAGA-PPM. Additionally, the numerical sources resulting from implementation of PPM were determined and removed.

Based on earlier work [12, 15], a weight function representing interpolation errors in the solution field, a center-of-mass scheme for moving the nodes, and a conservative procedure for redistributing the solution field over the adapted grid have been adopted in DSAGA-PPM. However, two improvements have been made in these procedures. First, the existing weight function formulation has been extended to take into account the resolution requirements of each of the species. Second, the higher order PPM scheme has been implemented in the procedure for calculating the fluxes resulting from grid movement.

Testing of DSAGA-PPM with advection model problems, including one-dimensional waveforms (rectangular, triangular, and gaussian) and a two-dimensional

rotating cone, demonstrate that: 1) it can accurately represent dynamic solution features that have sharp gradients and discontinuities, as well as those that are smooth and well-defined and 2) it is capable of increasing spatial resolution simultaneously in multiple locations. For example, after one revolution of a rotating cone, 87% of the cone's peak is retained using DSAGA-PPM while only 63% is retained using a static grid analog of DSAGA-PPM (SGA-PPM). In another test with four rotating cones, DSAGA-PPM was able to retain 87% of each of the cone peaks, after one revolution. The root-mean-square errors in DSAGA-PPM results are about 4-5 times lower than those in the corresponding SGA-PPM results. The results with these model problems demonstrate that DSAGA-PPM can compute advection much more accurately than SGA-PPM and therefore would assure more accurate starting concentrations for chemistry calculations.

Next, DSAGA-PPM was tested with a model problem in which a conical pollutant puff moves in rotating field and undergoes chemical transformation. For this problem, exact reference solutions were computed over the adaptive and static grids. Using these reference solutions, errors in DSAGA-PPM and corresponding SGA-PPM results were determined. Comparing the values of errors in DSAGA-PPM and SGA-PPM solutions, it is shown that DSAGA-PPM results are more accurate than SGA-PPM results. For example, the root-mean-square errors in the DSAGA-PPM solutions are about 4-6 times less than those in the corresponding SGA-PPM solutions. Tests also demonstrate that DSAGA-PPM is able to respond to adaptation requirements generated by rapid nonlinear chemical

transformations and by drastic changes in the pollutant distributions. These findings support the weight function formulation developed in this work.

Consequently, DSAGA-PPM was tested with two model problems of interest in air quality modeling. In the first problem, the evolution of a power plant plume was simulated. The DSAGA-PPM solution reflects the early, the intermediate, and the mature stages of plume development reported in literature [67]. Accordingly, in this solution the O_3 concentrations rise above the background levels at quite large distances downwind of the source. In contrast, the SGA-PPM solution, obtained using a static grid with 10 km x 10 km grid cells, does not reflect the stages of plume development. In the SGA-PPM simulation, power plant emissions are instantaneously dispersed over a large grid volume and their chemistry is artificially shifted from that of the high- NO_x regime to low- NO_x regime in a sudden fashion. Consequently, the oxidation of NO to NO_2 and production of O_3 occurs near the source. Good mass conservation is exhibited in both DSAGA-PPM and SGA-PPM simulations. In general, results of this model problem reflect that even a static grid with grid cells as small as 10 km x 10 km is unable to resolve the delicate and non-linear nature of NO_x and odd-H chemistry. Consequently, the O_3 field resulting from a simulation with such a grid can be inaccurate. In contrast, a DSAGA-PPM simulation would be able to resolve the multiple spatial and temporal scales associated with O_3 production and transport and would provide assurance that the resulting O_3 field is reasonably accurate.

In the second problem, the ozone production resulting due to dynamic interactions between emissions from two power plants and an urban area is followed in time. In the DSAGA-PPM solution, peak O_3 values near the upwind point source and the urban area are 45 ppb and 63 ppb, respectively; these values in the corresponding SGA-PPM solution are 37 ppb and 57 ppb, respectively. Further, the solution obtained using a more accurate refined static grid reflects peak O_3 values near the upwind point source and the urban area of 39 ppb and 62 ppb, respectively. These results reflect that the distribution of O_3 concentration resulting from dynamic interactions amongst plumes from various sources may not be accurately predicted unless a sufficiently refined grid is used. The results demonstrate that DSAGA-PPM is able to follow and resolve these interactions while maintaining mass with good accuracy.

Performance achieved by DSAGA-PPM in model problem simulations indicates that it has the potential to provide accurate air quality modeling solutions at costs that may be 10 times less than those incurred in obtaining equivalent static grid solutions.

References

1. M.T. Odman and A.G. Russell, *Atmospheric Environment* **25A** (1991), 2385.
2. M.T. Odman, R. Mathur, K. Alapaty, R.K. Srivastava, D.S. McRae, and R.J. Yamartino, Nested and Adaptive Grids for Multiscale Air Quality Modeling, in *Next Generation Environmental Models and Computational Methods*, 1997, edited by Delic and Wheeler, SIAM, 59.
3. G.S. Dietachmayer and K.K. Droegemeier, *Monthly Weather Rev.* **120** (1992), 1675.
4. A.S. Almgren, J.B. Bell, P. Colella, L.H. Howell, and M.L. Welcome, A High Resolution Adaptive Projection Method for Regional Atmospheric Modeling, in *Next Generation Environmental Models and Computational Methods*, 1997, edited by Delic and Wheeler, SIAM, 69.
5. W.C. Skamarock and J.B. Klemp, *Monthly Weather Rev.* **121** (1993), 788.
6. A. Tomlin, M. Berzins, J. Ware, J. Smith, and M.J. Pilling, *Atmospheric Environment* **31** (1997), 2945.
7. R.A. Benson and D.S. McRae, A Three-Dimensional Dynamic Solution-Adaptive Mesh Algorithm, *AIAA Paper 90-1566*, June 1990.
8. R.A. Benson and D.S. McRae, A Solution Adaptive Mesh Algorithm for Dynamic/Static Refinement of Two and Three Dimensional Grids, in *Proc. Third International Conference on Numerical Grid Generation in Computational Field Simulations, Barcelona, Spain, June 1991*, 185.

9. R.A. Benson and D.S. McRae, Time Accurate Simulations of Unsteady Flows with a Dynamic Solution-Adaptive Algorithm, in *Proc. Fourth International Conference on Numerical Grid Generation in Computational Field Simulations, Swansea, U.K., April 1994*, 573.
10. R.A. Benson, D.S. McRae, and J.R. Edwards, Numerical Simulations Using a Dynamic Solution-Adaptive Grid Algorithm with Applications to Unsteady Internal flows, *AIAA Paper 92-2719, June 1992*.
11. R.A. Benson and D.S. McRae, Numerical Simulations of the Unstart Phenomena in a Supersonic Inlet/Diffuser, *AIAA Paper 93-2239, June 1993*.
12. R.A. Benson, "Development of a Time-Accurate Solution Algorithm Coupled to a Dynamic Solution-Adaptive Grid Algorithm with Application to Generic Inlet/Diffuser Configurations," Ph.D. Thesis, North Carolina State University, Raleigh, NC, 1994.
13. R.K. Srivastava, D.S. McRae, and M.T. Odman, Application of a Solution Adaptive Grid Technique to Air Quality Modeling, in *Proc. Fifth International Conference on Numerical Grid Generation in Computational Field Simulations, Vol. II, Starkville, Mississippi, April 1996*, edited by Soni, Thompson, Hauser, and Eiseman, 1241.
14. K.R. Laflin and D.S. McRae, Three-Dimensional Viscous Flow Computations Using Near-Optimal Grid Redistribution, in *Proc of First AFOSR Conference on Dynamic Motion CFD, Rutgers University, New Jersey, June 1996*.
15. K.R. Laflin, "Solver Independent r-Refinement Adaptation for Dynamic Numerical Simulations," Ph.D. Thesis, North Carolina State University, Raleigh, NC, 1997.
16. P. Collela and P.R. Woodward, *J. Comput. Phys* **54** (1984), 174.
17. G.J. McRae, W.R. Goodin, and J.H. Seinfeld, *J. Comput. Phys* **45** (1982), 1.

18. Mark Z. Jacobson, *Atmospheric Environment* **31** (1997), 587.
19. S.D. Reynolds, P.M. Roth, and J.H. Seinfeld, *Atmospheric Environment* **7** (1973), 1033.
20. P. Defrise, *Advances in Geophysics* **10** (1964), 261-315.
21. J.H. Seinfeld, "Atmospheric Chemistry and Physics of Air Pollution," John Wiley & Sons: New York, 1986.
22. A. Venkatram, *Atmospheric Environment* **27A** (1993), 1963-1966.
23. O.B. Toon, R.P. Turco, D. Westphal, R. Malone, and M. S. Liu, *J. Atmos. Sciences* **45** (1988), 2123.
24. Van Dop, *Atmospheric Environment* **32** (1998), 257.
25. R.G. Lamb, *Atmospheric Environment* **7** (1973), 257.
26. R.G. Lamb and J.H. Seinfeld, *Environ. Sci. Technol.* **7** (1973), 253.
27. G.J. Haltiner and R.T. Williams, "Numerical Prediction and Dynamic Meteorology," Second Edition, John Wiley & Sons: New York, 1980.
28. A. Kasahara, *Monthly Weather Rev.* **102** (1974), 509.
29. P.J. Roache, "Computational Fluid Dynamics," Second ed., Hermosa Publications, Albuquerque, 1976.

30. D.A. Anderson, J.C. Tannehill, and R.H. Pletcher, "Computational Fluid Mechanics and Heat Transfer," First ed., Hemisphere Publishing Corporation, New York, 1984.
31. W.L. Oberkampf, *Int. J. Numer. Methods Engrg.* **10** (1976), 211.
32. M. Vinokur, *J. Comput. Phys.* **14** (1974), 105.
33. N.N. Yanenko, "The Method of Fractional Steps," Springer-Verlag, New York, 1971.
34. R.L. Carpenter, Jr., K.K. Droegemeier, P.R. Woodward, and C.E. Hane, *Monthly Weather Rev.* **118** (1990), 586.
35. R.G. Hindman, *AIAA Journal* **20** (1982), 1359.
36. T.H. Pulliam and J.L. Steger, *AIAA Journal* **18** (1980), 159.
37. P.D. Thomas and C.K. Lombard, The Geometric Conservation Law - A Link Between Finite - Difference and Finite - Volume Methods for Flow Computation on Moving Grids, *AIAA Paper 78-1208*, July 1978.
38. P.D. Thomas, "Numerical Method for Predicting Flow Characteristics and Performance of Nonaxisymmetric Nozzles - Theory," NASA CR-3147, September 1979.
39. T.P. Gielda and D.S. McRae, "Explicit Parabolized Solutions for Secant-Ogive-Cylinder-Boat-Tail and Finned Projectiles," AIAA 87-2292, AIAA 5th Applied Aerodynamics Conference, Monterey, CA, August 1987.
40. T. Nitta, *J. Meteorol. Soc. Japan* **40** (1962), 13.

- 41. J.M. Varah, *SIAM J. Numer. Anal.* **8** (1971), 569.
- 42. J.M. Varah, *SIAM J. Numer. Anal.* **8** (1971), 598.
- 43. J.H. Chen, *J. Comput. Phys.* **13** (1973), 522.
- 44. B. Gustafsson, H.O. Kreiss, and A. Sundstrom, *Math. Comp.* **26** (1972), 649.
- 45. B. Gustafsson, *J. Comput. Phys.* **34** (1980), 108.
- 46. D. Gottlieb and E. Turkel, *J. Comput. Phys.* **26** (1978), 181.
- 47. C. Hirsch, "Numerical Computation of Internal and External Flows, Vol. 2" John Wiley & Sons, New York, 1990.
- 48. G.R. Gavalas, "Nonlinear Differential Equation of Chemically Reacting Systems," Springer Tracts in Natural Philosophy, **17**, Springer-Verlag, 1968.
- 49. R.A. Willoughby, "Stiff Differential Systems," Plenum, New York, 1974.
- 50. J.D. Lambert, in "Computational Techniques for Ordinary Differential Equations," Academic Press, London, 1976.
- 51. D.D. Warner, *J. Phys. Chem.* **81** (1977), 980.
- 52. A.R. Curtis, in "Numerical Software - Needs and Availability," Academic Press, London, 1978.
- 53. H.O. Kreiss, *SIAM J. Numer. Anal.* **16** (1979), 980.

- 54. L.F. Shampine and C.W. Gear, *SIAM Rev.* **21** (1979), 1.
- 55. T.R. Young and J.P. Boris, *J. Phys. Chem.* **81** (1977), 2424.
- 56. A.C. Hindmarsh and G.D. Byrne, “ EPISODE: An Experimental Program for the Integration of Systems of Ordinary Differential Equation Systems,” Lawrence Livermore Laboratory Report UCID-30112, 1975.
- 57. G.F. Carey, “Computational Grids: Generation, Adaptation, and Solution Strategies,” Taylor & Francis, Washington, D.C., 1997.
- 58. P.R. Eiseman, Adaptive Grid Generation, *Comput. Methods Appl. Mech. Engrg.* **64** (1987), 321.
- 59. P.K. Smolarkiewicz and P. J. Rasch, *J. Atmos. Sciences* **48** (1991), 793.
- 60. A. Bott, *Monthly Weather Rev.* **117** (1989), 1006.
- 61. P.K. Smolarkiewicz, *Monthly Weather Rev.* **111** (1983), 479.
- 62. Mehmet T. Odman and Armistead. G. Russel, *J. Geophys. Res.* **96** (1991), 7363.
- 63. David P. Chock and Sandra L. Winkler, *J. Geophys. Res.* **99** (1994), 1033.
- 64. O Hov, Z. Zlatev, R. Berkowicz, A. Eliassen, and L.P. Prahm, *Atmos. Environ.* **23** (1989), 967.
- 65. S.C. Liu, M. Trainer, F.C. Fehsenfeld, D.D. Parrish, E.J. Williams, D.W. Fahey, G. Hubler, and P.C. Murphey, *J. Geophys. Res.* **92** (1987), 4191.

- 66. S. Sillman, J.A. Logan, and S.C. Wofsey, *J. Geophys. Res.* **95** (1990), 1837.
- 67. Noor V. Gillani and Jonathan E. Pleim, *Atmos. Environ.* **30** (1996), 2043.
- 68. S. Sillman, J.A. Logan, and S.C. Wofsey, *J. Geophys. Res.* **95** (1990), 5731.
- 69. Leonard K. Peters, Carl M. Berkowitz, Gregory R. Carmichael, Richard C. Easter, Graeme Fairweather, Steven J. Ghan, Jeremy M. Hales, L. Ruby Leung, William R. Pennell, Florian A. Potra, Rick D. Saylor, and Tate T. Tsang, *Atmos. Environ.* **29** (1995), 189.

Appendices

Appendix A Coordinate Systems

Several different coordinate systems are in use in air pollution modeling. In this appendix, the methods of tensor calculus are used, along with the atmospheric diffusion equation derived in Chapter 2, to provide a cohesive treatment of these various coordinate systems. Specifically, expressions are derived for metric tensor and contravariant velocity components for the coordinate systems popular in air pollution modeling. Also the contravariant velocity components are related to the physical velocity components. These expressions are then substituted in Equation (2.7) to derive the atmospheric diffusion equation in the coordinate system of interest. Further, if K-theory is used for parameterizing turbulent diffusion, Equation (2.12) can be used instead of (2.7).

The coordinate systems of interest are located in an Euclidean space. In treatment of the various coordinate systems, a convenient coordinate system will be selected and, subsequently, the properties of the desired system will be derived by using the tools of tensor calculus already introduced in Section 2.1. The selected coordinate system will be denoted by $\mathbf{x}$ while the desired coordinate system will be denoted by $\hat{\mathbf{x}}$. Unless otherwise stated, $\mathbf{x}$ is a Cartesian system. Note that if the vectors are always expressed in a Cartesian frame, then no derivatives of unit vectors are needed in expressions of these vectors in some other coordinate system.

A.1 Tangential Cartesian Coordinates

The tangential Cartesian coordinates can be established on the surface of the earth, as shown in Figure A-1. The origin of this coordinate system is located at a position O on the earth's surface. The x_t and y_t axes form a plane tangent to the earth's surface and point towards the east and the north, respectively. The z_t axis points in the radial direction making the coordinate system a right-handed one. As discussed in Dutton [1], the use of this coordinate system is limited to small-scale local studies because of curvature associated with the spherical geometry of the earth.

Figure A.1. Tangential Cartesian Coordinates

Let, $\mathbf{x}$ and $\hat{\mathbf{x}}$ be the global rectangular Cartesian and tangential Cartesian coordinates, respectively. Then

$$\begin{aligned} x^1 &= x, \quad x^2 = y, \quad \text{and} \quad x^3 = z \\ \hat{x}^1 &= x_t, \quad \hat{x}^2 = y_t, \quad \text{and} \quad \hat{x}^3 = z_t \end{aligned} \tag{A.1}$$

Using Equation (2.4), the metric tensor for $\hat{\mathbf{x}}$ can be obtained as

$$\hat{g}_{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.2}$$

Note that $\hat{\mathbf{x}}$ is an orthogonal coordinate system as evidenced by $\hat{g}_{ij}$ being diagonal.

Also,

$$\hat{\gamma} = \det(\hat{g}_{ij}) = 1 \tag{A.3}$$

The contravariant velocity components may be obtained using Equation (2.3) as

$$\begin{aligned} \hat{v}^1 &= d\hat{x}^1/dt = dx_t/dt = u_t \\ \hat{v}^2 &= d\hat{x}^2/dt = dy_t/dt = v_t \\ \hat{v}^3 &= d\hat{x}^3/dt = dz_t/dt = w_t \end{aligned} \tag{A.4}$$

Using Equation (2.7), the atmospheric diffusion equation in this coordinate system is

$$\begin{aligned}
 \frac{\partial}{\partial t}(\overline{c_l}) = & -\frac{\partial}{\partial x_t}(\overline{c_l} \overline{u_t}) - \frac{\partial}{\partial y_t}(\overline{c_l} \overline{v_t}) - \frac{\partial}{\partial z_t}(\overline{c_l} \overline{w_t}) \\
 & - \frac{\partial}{\partial x_t}(\overline{c'_l} \overline{u'_t}) - \frac{\partial}{\partial y_t}(\overline{c'_l} \overline{v'_t}) - \frac{\partial}{\partial z_t}(\overline{c'_l} \overline{w'_t}) \\
 & + R_l(\overline{c_1}, \dots, \overline{c_N}) + S_l(x_t, y_t, z_t, t)
 \end{aligned} \tag{A.5}$$

A.2 Spherical Coordinates

The geometry of the earth is nearly spherical with slight ellipticity. In common practice, the earth is taken to be a sphere with associated spherical coordinates. Consider the spherical coordinate system $\hat{\mathbf{x}}$ and an associated rectangular Cartesian coordinate system $\mathbf{x}$, as shown in Figure A-2.

Let,

$$\begin{aligned}
 x^1 &= x, \quad x^2 = y, \quad x^3 = z \\
 \hat{x}^1 &= r, \quad \hat{x}^2 = \lambda, \quad \hat{x}^3 = \phi
 \end{aligned} \tag{A.6}$$

Figure A.2. Spherical Coordinates.

From the geometry shown in Figure A-2,

$$x = r \cos \phi \cos \lambda, \quad y = r \cos \phi \sin \lambda, \quad z = r \sin \phi \quad (\text{A.7})$$

and,

$$r = (x^2 + y^2 + z^2)^{1/2}, \quad \lambda = \arctan(y/x), \quad \phi = \arctan[z/(x^2 + y^2)^{1/2}] \quad (\text{A.8})$$

Using Equation (2.4), the metric tensor is given by

$$\hat{g}_{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & r^2 \cos^2 \phi & 0 \\ 0 & 0 & r^2 \end{bmatrix} \quad (\text{A.9})$$

Since the metric tensor is diagonal, the spherical coordinate system is orthogonal. Also,

$$\hat{\gamma} = \det(\hat{g}_{ij}) = r^4 \cos^2 \phi \quad (\text{A.10})$$

The contravariant velocity components may be obtained using Equation (2.3) as

$$\begin{aligned} \hat{v}^1 &= \frac{dr}{dt} = \dot{r} = (\cos \phi \cos \lambda) u + (\cos \phi \sin \lambda) v + (\sin \phi) w \\ \hat{v}^2 &= \frac{d\lambda}{dt} = \dot{\lambda} = \left(\frac{-\sin \lambda}{r \cos \phi} \right) u + \left(\frac{\cos \lambda}{r \cos \phi} \right) v \\ \hat{v}^3 &= \frac{d\phi}{dt} = \dot{\phi} = \left(\frac{-\sin \phi \cos \lambda}{r} \right) u + \left(\frac{-\sin \phi \sin \lambda}{r} \right) v + \left(\frac{\cos \phi}{r} \right) w \end{aligned} \quad (\text{A.11})$$

Now using Equation (2.7), the atmospheric diffusion equation in this coordinate system becomes

$$\begin{aligned}
\frac{\partial}{\partial t}(\overline{c_l}) = & -\frac{1}{r^2} \frac{\partial}{\partial r} (r^2 \overline{c_l} \dot{\overline{r}}) - \frac{\partial}{\partial \lambda} (\overline{c_l} \dot{\overline{\lambda}}) - \frac{1}{\cos \phi} \frac{\partial}{\partial \phi} (\overline{c_l} \dot{\overline{\phi}}) \\
& - \frac{1}{r^2} \frac{\partial}{\partial r} (r^2 \cos \phi \overline{c'_l} \dot{\overline{r'}}) - \frac{\partial}{\partial \lambda} (\overline{c'_l} \dot{\overline{\lambda'}}) - \frac{1}{\cos \phi} \frac{\partial}{\partial \phi} (\overline{c'_l} \dot{\overline{\phi''}}) \\
& + R_l(\overline{c_1}, \dots, \overline{c_N}) + S_l(r, \lambda, \phi, t)
\end{aligned} \tag{A.12}$$

A.3 Generalized Vertical Coordinate

The vertical structure of the atmosphere suggests that it might be useful to replace the geometrical vertical coordinate by one of the flow variables and, thus, use a generalized vertical coordinate in meteorological modeling. This generalization of the vertical structure of the atmosphere can provide simplifications in modeling procedures and allow terrain following. Several vertical coordinates in use have been discussed in Kasahara [2].

Consider an orthogonal coordinate system $\mathbf{x}$ and an associated generalized vertical coordinate system $\hat{\mathbf{x}}$ such that

$$\hat{x}^1 = x^1, \quad \hat{x}^2 = x^2, \quad \hat{x}^3 = \hat{x}^3(x^1, x^2, x^3, x^4 = t), \quad \hat{x}^4 = \hat{t} = t \tag{A.13}$$

The Jacobian matrix of the transformation from $\mathbf{x}$ to $\hat{\mathbf{x}}$ is given by,

$$J_{\hat{x}}^x = \frac{\partial (\hat{x}^1, \hat{x}^2, \hat{x}^3, \hat{x}^4)}{\partial (x^1, x^2, x^3, x^4)} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \frac{\partial \hat{x}^3}{\partial x^1} & \frac{\partial \hat{x}^3}{\partial x^2} & \frac{\partial \hat{x}^3}{\partial x^3} & \frac{\partial \hat{x}^3}{\partial x^4} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (\text{A.14})$$

The determinant of the Jacobian matrix is

$$\left| J_{\hat{x}}^x \right| = \frac{\partial \hat{x}^3}{\partial x^3} \quad (\text{A.15})$$

Since $\left| J_{\hat{x}}^x \right|$ is non-zero everywhere in $\mathbf{x}$, the transformation is uniquely invertible. Then

$$x^1 = \hat{x}^1, \quad x^2 = \hat{x}^2, \quad x^3 = x^3(\hat{x}^1, \hat{x}^2, \hat{x}^3, \hat{x}^4), \quad x^4 = t = \hat{t} \quad (\text{A.16})$$

The metric tensor, $\hat{g}_{ij}$, can be obtained from Equation (2.4). Since $\mathbf{x}$ is an orthogonal system, g_{kl} is diagonal. Therefore,

$$\hat{g}_{ij} = \begin{bmatrix} g_{11} + \left(\frac{\partial x^3}{\partial \hat{x}^1} \right)^2 g_{33} & \frac{\partial x^3}{\partial \hat{x}^1} \frac{\partial x^3}{\partial \hat{x}^2} g_{33} & \frac{\partial x^3}{\partial \hat{x}^1} \frac{\partial x^3}{\partial \hat{x}^3} g_{33} \\ \frac{\partial x^3}{\partial \hat{x}^1} \frac{\partial x^3}{\partial \hat{x}^2} g_{33} & g_{22} + \left(\frac{\partial x^3}{\partial \hat{x}^2} \right)^2 g_{33} & \frac{\partial x^3}{\partial \hat{x}^2} \frac{\partial x^3}{\partial \hat{x}^3} g_{33} \\ \frac{\partial x^3}{\partial \hat{x}^1} \frac{\partial x^3}{\partial \hat{x}^3} g_{33} & \frac{\partial x^3}{\partial \hat{x}^2} \frac{\partial x^3}{\partial \hat{x}^3} g_{33} & \left(\frac{\partial x^3}{\partial \hat{x}^3} \right)^2 g_{33} \end{bmatrix} \quad (\text{A.17})$$

Note that $\hat{g}_{ij}$ is not diagonal, therefore the generalized vertical coordinate system is not orthogonal. Finally,

$$\hat{\gamma} = \det(\hat{g}_{ij}) = g_{11} g_{22} g_{33} \left(\frac{\partial x^3}{\partial \hat{x}^3} \right)^2 \quad (\text{A.18})$$

Again,

$$g_{11} g_{22} g_{33} = \gamma \quad (\text{A.19})$$

Therefore,

$$\hat{\gamma} = \gamma \left(\frac{\partial x^3}{\partial \hat{x}^3} \right)^2 \quad (\text{A.20})$$

Further, it can be shown that

$$\frac{\partial x^3}{\partial \hat{x}^3} = \frac{1}{\frac{\partial \hat{x}^3}{\partial x^3}} \quad (\text{A.21})$$

The contravariant components of the velocity vector, from Equation (2.3) are

$$\begin{aligned} \hat{v}^1 &= \frac{d\hat{x}^1}{dt} = v^1 \\ \hat{v}^2 &= \frac{d\hat{x}^2}{dt} = v^2 \\ \hat{v}^3 &= \frac{d\hat{x}^3}{dt} = \dot{\hat{x}}^3 = \frac{\partial \hat{x}^3}{\partial x^1} v^1 + \frac{\partial \hat{x}^3}{\partial x^2} v^2 + \frac{\partial \hat{x}^3}{\partial x^3} v^3 + \frac{\partial \hat{x}^3}{\partial t} \end{aligned} \quad (\text{A.22})$$

The atmospheric diffusion equation in the generalized vertical coordinate system becomes

$$\begin{aligned}
& \frac{\partial}{\partial t} \left(\sqrt{\gamma} \frac{\partial x^3}{\partial \hat{x}^3} c_l \right) = \\
& - \frac{\partial}{\partial x^1} \left(\sqrt{\gamma} \frac{\partial x^3}{\partial \hat{x}^3} c_l \overline{v^1} \right) - \frac{\partial}{\partial x^2} \left(\sqrt{\gamma} \frac{\partial x^3}{\partial \hat{x}^3} c_l \overline{v^2} \right) - \frac{\partial}{\partial x^3} \left(\sqrt{\gamma} \frac{\partial x^3}{\partial \hat{x}^3} c_l \overline{\hat{x}^3} \right) \\
& - \left[\frac{\partial}{\partial x^1} \left(\sqrt{\gamma} \frac{\partial x^3}{\partial \hat{x}^3} c_l \overline{v^1}' \right) + \frac{\partial}{\partial x^2} \left(\sqrt{\gamma} \frac{\partial x^3}{\partial \hat{x}^3} c_l \overline{v^2}' \right) + \frac{\partial}{\partial x^3} \left(\sqrt{\gamma} \frac{\partial x^3}{\partial \hat{x}^3} c_l \overline{\hat{x}^3}' \right) \right] \\
& + \sqrt{\gamma} \frac{\partial x^3}{\partial \hat{x}^3} \left[R_l(\overline{c_1}, \dots, \overline{c_N}) + S_l(x^1, x^2, \hat{x}^3, t) \right]
\end{aligned} \tag{A.23}$$

A.4 Summary

The results derived above can be summarized as follows:

$$\begin{aligned}
& \frac{\partial}{\partial \hat{x}^\alpha} \left(\sqrt{\gamma} \frac{\partial x^3}{\partial \hat{x}^3} c_l \overline{v^\alpha} \right) + \frac{\partial}{\partial \hat{x}^k} \left(\sqrt{\gamma} \frac{\partial x^3}{\partial \hat{x}^3} c_l \overline{v^k}' \right) = \\
& \sqrt{\gamma} \frac{\partial x^3}{\partial \hat{x}^3} \left[R_l(\overline{c_1}, \dots, \overline{c_N}) + S_l(x^\alpha) \right]
\end{aligned} \tag{A.24}$$

where $\sqrt{\gamma}$ pertains to a certain orthogonal coordinate system $\mathbf{x}(x^1, x^2, x^3)$ and $\frac{\partial x^3}{\partial \hat{x}^3}$ to a corresponding generalized vertical coordinate. A compact form has been used in the above equation with $\alpha = 1, 2, 3, 4$ and $k = 1, 2, 3$. Note that x^α , v^α , and $\sqrt{\gamma}$ have already

been derived for tangential Cartesian and spherical coordinate systems. For various popular generalized vertical coordinates, $\frac{\partial x^3}{\partial \hat{x}^3}$ are given in Table A-1. Toon *et al.* [3] arrived at a similar formulation for their generalized coordinate aerosol model.

Equation (A.24) can be implemented into an air quality model incorporating a generalized vertical coordinate system. Such an implementation would accommodate most of the common coordinate systems used in air pollution transport modeling. For example, the horizontal coordinates can be selected from Cartesian, spherical, or conformal mapped coordinates and the vertical coordinate can be any of the coordinates shown in Table A-1. Note that in Equation (A.24) no parameterization is specified for the turbulent diffusion terms. This allows for modular model structure and flexibility in the choice of turbulent parameterizations.

When implementing Equation (A.24) into an air quality model, the determinant of the metric tensor, i.e., $\sqrt{\gamma} \frac{\partial x^3}{\partial \hat{x}^3}$, should be retained in this form. In other words, no simplifications should be made based on the functional form of this quantity since simplifications will result for specific coordinate systems, when appropriate.

The transformation noted in Section A.3 can be used with a more general transformation (see Chapter 3) to develop a single generalized coordinate air quality

model that will also permit implementation of an adaptive mesh technique. Once this general transformation is implemented, computation will take place in a uniform grid computational space and there will be no further need for changing the model itself to reflect changes in coordinate systems. This would have the benefit that one model could then be used for all popular coordinate systems which allow invertible transformations for all points in the physical domain of interest. The solution will always be available in the original physical coordinates since the original transformation will either remain fixed or, in case an adaptive mesh algorithm is used, will be continuously updated. Expressions of the results in coordinates other than the original physical will always be available through a post-processing transformation.

Table A.1. Some Generalized Vertical Coordinates

Vertical Coordinate		$\frac{\partial x^3}{\partial \hat{x}^3}$	Reference	Remarks
Altitude		1	Toon [3]	
Pressure		$\frac{1}{\rho g}$	Toon [3]	ρ is density of air
Sigma-p	$\sigma_p = \frac{p - p_T}{p_s - p_T}$	$-\frac{p_s - p_T}{\rho g}$	Chang [4]	$p_s(x, y)$ is surface pressure, p_T is top layer pressure
Sigma	$\sigma = \frac{p_0 - p_T}{p_s - p_T}$	$-\frac{p_s - p_T}{\rho_0 g}$	Grell [5]	$p_0(z)$ is reference pressure, $\rho_0(z)$ is the reference air density
Sigma-z	$\sigma_z = \frac{z - h}{H - h}$	$H - h$	McRae [6]	$h(x, y)$ is lower surface, $H(x, y, t)$ is height of upper surface
Scaled height	$\bar{z} = \frac{H(z - z_s)}{H - z_s}$	$\frac{H - z_s}{H}$	Gal-Chen [7]	$z_s(x, y)$ is the topography

A.5 References

1. Dutton, J.A.; *The Ceaseless Wind*; Dover: New York, 1986; pp 226-229.
2. Kasahara, A. Mon. Wea. Rev. **1974** 102, 509-522.
3. Toon, O.B.; Turco, R.P.; Westphal, D.; Malone, R.; Liu, M.S. Journal of Atmospheric Sciences **1988** 45, 2123-2143.
4. Chang, J.S., Brost, R.A., Isaksen, I.S.A., Madronich, S., Middleton, P., Stockwell, W.R., and Walcek, C.J., Journal of Geophysical Research **1987** 92, 14681-14700.
5. Grell, G. A., Dudhia, J. D., and Stauffer, D. R.; *A Description of the Fifth-Generation Penn State/NCAR Mesoscale Model (MM5)*, NCAR/TN-398+IA; National Center for Atmospheric Research: Boulder, Co., 1994.
6. McRae, G.J.; Goodin, W.R.; Seinfeld, J.H. J. Comput. Phys. **1982** 45, 1-42.
7. Gal-Chen, T.; Somerville, R. C. J. J. Comput. Phys. **1975** 17, 209-228.

Appendix B Grid Metric Terms

In this Appendix, the metric terms that are used to transform the governing equations from Cartesian coordinates to a generalized coordinate system are derived. The transformation is from physical space to a computational space which is more convenient for obtaining a solution. Since this work has focussed on applications in two dimensions, the metrics of the transformation are derived for two dimensions only. However, extension of the derivation procedure to three dimensions would be a straight forward extension of the procedure presented here.

In two dimensions, the computational coordinates are expressed as functions of physical space as follows:

$$\xi = \xi(x, y, z), \quad \eta = \eta(x, y, z). \quad (\text{B.1})$$

The derivatives of the generalized coordinates with respect to Cartesian space are:

$$\begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix} = T \begin{pmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{pmatrix}; \quad T = \begin{pmatrix} \xi_x & \eta_x \\ \xi_y & \eta_y \end{pmatrix} \quad (\text{B.2})$$

Again,

$$\begin{pmatrix} \frac{\partial}{\partial \xi} \\ \frac{\partial}{\partial \eta} \end{pmatrix} = T^{-1} \begin{pmatrix} \frac{\partial}{\partial x} \\ \frac{\partial}{\partial y} \end{pmatrix}; \quad T^{-1} = \begin{pmatrix} x_{\xi} & y_{\xi} \\ x_{\eta} & y_{\eta} \end{pmatrix} \quad (\text{B.3})$$

Using $T = (T^{-1})^{-1}$,

$$\begin{pmatrix} \xi_x & \eta_x \\ \xi_y & \eta_y \end{pmatrix} = J \begin{pmatrix} y_{\eta} & -y_{\xi} \\ -x_{\eta} & x_{\xi} \end{pmatrix} \quad (\text{B.4})$$

where

$$J = \frac{1}{x_{\xi} y_{\eta} - x_{\eta} y_{\xi}} \quad (\text{B.5})$$

The spatial terms are calculated on a computational grid in the following manner for a finite volume formulation:

$$\left. \frac{\partial x}{\partial \eta} \right|_{i+1/2,j} = x_{i+1,j+1} - x_{i+1,j} \quad (\text{B.6})$$

$$\left. \frac{\partial y}{\partial \eta} \right|_{i+1/2,j} = y_{i+1,j+1} - y_{i+1,j}$$

$$\left. \frac{\partial x}{\partial \xi} \right|_{i,j+1/2} = x_{i+1,j+1} - x_{i,j+1} \quad (\text{B.7})$$

$$\left. \frac{\partial y}{\partial \xi} \right|_{i,j+1/2} = y_{i+1,j+1} - y_{i,j+1}$$

The metrics in Equation (B.6) would apply at the $i \pm 1/2, j$ surface of a cell and the metrics in Equation (B.7) would apply at the $i, j \pm 1/2$ surface.

Appendix C The Piecewise Parabolic Method

As discussed in Collella and Woodward [1] and Woodward [2], *the Piecewise Parabolic Method* (PPM) represents a departure from finite difference techniques typically used to solve advection flow problems. According to Rood [3], the PPM can be classified as a finite volume scheme in which the fluid is discretized into a series of zone-averaged slabs rather than discrete points. In the PPM, the zone-average is used to construct a unique parabola for each dependent variable within a zone. Additionally, since each such parabola is constrained to be monotonic, the PPM is designed to provide non-oscillatory solutions. In this appendix, the PPM scheme for solving the advection equation is described.

The advection equation is expressed as

$$\frac{\partial a}{\partial t} + \frac{\partial ua}{\partial \xi} = 0 \quad ; \quad a(\xi, 0) = a_0(\xi) \quad (\text{C.1})$$

where a is any dependent variable, such as density.

Let $\xi_{j+1/2}$ be the boundary between the j^{th} and the $j+1^{\text{st}}$ zones on the computational grid. Then the zone average of a in the j^{th} zone at time t^n is

$$a_j^n = \frac{1}{\Delta \xi_j} \int_{\xi_{j-1/2}}^{\xi_{j+1/2}} a(\xi) d\xi \quad ; \quad \Delta \xi_j = \xi_{j+1/2} - \xi_{j-1/2} \quad (\text{C.2})$$

Let x be a normalized zone coordinate given by

$$x = \frac{\xi - \xi_{j-1/2}}{\Delta \xi_j} \quad (\text{C.3})$$

Using x , (C.2) can be expressed as

$$a_j^n = \int_0^1 a(x) dx \quad (\text{C.4})$$

The distribution of a within a zone is assumed to be parabolic, i.e., for the j^{th} zone,

$$a(x) = a_0 + a_1 x + a_2 x^2 \quad (\text{C.5})$$

PPM uses three related parameters to express the above distribution: the zone average a_j^n and the values of the parabola at the left and right zone edges, $a(\xi_{j-1/2}) = a_{L,j}$ and $a(\xi_{j+1/2}) = a_{R,j}$. This leads to the parabolic representation

$$a(x) = a_{L,j} + x(\Delta a_j + a_{6,j}(1-x)) \quad (\text{C.6})$$

where

$$\Delta a_j = a_{R,j} - a_{L,j} \quad (\text{C.7})$$

and using (C.4), (C.6), (C.7),

$$a_{6,j} = 6 \left(a_j^n - \frac{1}{2}(a_{L,j} + a_{R,j}) \right) \quad (\text{C.8})$$

Taking finite volume cell values as zone averages, the construction of parabolas (C.6) simply requires finding appropriate values for the zone edges. These are found using an interpolation process described below.

As discussed in Carpenter *et al.* [4], in the first step of the interpolation process, a piecewise linear distribution is constructed. The monotonicity constraint requires that

every portion of the distribution in a given zone must lie within the range spanned by the averages in that zone and in its two neighbors. Therefore, the slopes Δa_j are determined by taking the smallest magnitude of the three slopes: the centered difference involving both neighboring zone averages, and the two one-sided differences involving each neighbor separately. If the zone average itself represents an extremum, the slope in that zone is set to zero.

In the second step, “first guess” parabolas are constructed by assigning provisional edge values through the unique cubic curve fit involving the two zone averages and the two slopes surrounding each edge. For example, as shown in [1], the left edge in the j^{th} zone has the value

$$a_{L,j} = \frac{1}{2}(a_{j-1}^n + a_j^n) - \frac{1}{6}(\Delta a_j - \Delta a_{j-1}) \quad (\text{C.9})$$

The “first guess” interpolation may introduce an undershoot and an overshoot in the $j-1^{\text{th}}$ and $j+1^{\text{th}}$ zones.

Finally, the undershoots and overshoots in the “first guess” parabolas are monotonized. For this, two cases are considered. In the first, a parabola is completely flattened if the corresponding zone average is a local extremum. The second case involves parabolas with amplitudes extending beyond the neighboring zone averages, i.e.,

those in which $|\Delta a_j| < |a_{6,j}|$. In such cases, one of the edge values of the parabola in question is redefined such that the slope of the curve is zero at the opposite edge. Because both Δa_j and $a_{6,j}$ are known, it is easy to identify which of the edge values to change and the degree of change needed [1].

Once monotonic parabolic distributions $a(x)$ (or $a(\xi)$) are available, average values of a near cell faces are determined as follows.

$$f_{j+1/2,L}^{avg}(y) = \frac{1}{y} \int_{\xi_{j+1/2-y}}^{\xi_{j+1/2}} a(\xi) d\xi ; y > 0 \quad (C.10)$$

$$f_{j+1/2,R}^{avg}(y) = \frac{1}{y} \int_{\xi_{j+1/2}}^{\xi_{j+1/2+y}} a(\xi) d\xi ; y > 0 \quad (C.11)$$

where $f_{j+1/2,L}^{avg}$ and $f_{j+1/2,R}^{avg}$ are the average values of a on the left and the right of the cell face $j+1/2$, respectively. After substituting $a(x)$ (or $a(\xi)$) in the above expressions, integrating, and simplifying,

$$f_{j+1/2,L}^{avg}(z) = a_{R,j} - \frac{z}{2} \left(\Delta a_j - \left(1 - \frac{2}{3} z \right) a_{6,j} \right) ; z = \frac{y}{\Delta \xi_j} \quad (C.12)$$

and

$$f_{j+1/2,R}^{avg}(z) = a_{L,j+1} + \frac{z}{2} \left(\Delta a_{j+1} - \left(1 - \frac{2}{3}z\right) a_{6,j+1} \right) ; \quad z = \frac{y}{\Delta \xi_j} \quad (\text{C.13})$$

Now considering $y = u \Delta t$,

$$z = \frac{u \Delta t}{\Delta \xi_j} = CFL \quad (\text{C.14})$$

where CFL is the *Courant-Levy-Friedrich* number.

Then the flux at the cell face $j+1/2$ is $u \bar{a}_{j+1/2}$ where

$$\bar{a}_{j+1/2} = f_{j+1/2,L}^{avg}(CFL) \quad \text{if} \quad u \geq 0 \quad (\text{C.15})$$

or

$$\bar{a}_{j+1/2} = f_{j+1/2,R}^{avg}(-CFL) \quad \text{if} \quad u < 0 \quad (\text{C.16})$$

References

1. Collela, P., and Woodward, P.R., *J. Comput. Phys* **54** (1984), 174.
2. Woodward, P.R., "Piecewise Parabolic Methods for Astrophysical Fluid Dynamics," *Astrophysical Radiation Hydrodynamics* (1986), K.H. Winkler and M.L. Norman, Eds., D. Reidel, 245-326.
3. Rood, R.B., *Rev. of Geophys.* **25** (1987), 71-100.
4. Carpenter, R.L., Jr., Droegemeier, K.K., Woodward, P.R., and Hane, C.E., *Monthly Weather Rev.* **118** (1990), 586.

Appendix D The Asymptotic Integration Method

The *asymptotic integration method* developed by Young [1] and Young and Boris [2] can be used to integrate the system of equations depicted by:

$$\frac{dc_l}{dt} = R_l(c_1, c_2, \dots, c_N, t) = a_l - b_l c_l \quad ; \quad l = 1, \dots, N \quad (\text{D.1})$$

In Equations (D.1), a_l and $b_l c_l$, respectively, are the production and loss rates for species c_l . Note that if a_l and b_l are constants then (D.1) can be solved to yield

$$c_l(t) = \frac{a_l}{b_l} + \left[c_l(0) - \frac{a_l}{b_l} \right] \exp(-b_l t) \quad (\text{D.2})$$

In Equation (D.2), it can be seen that $\frac{1}{b_l}$ describes how quickly species c_l reaches its equilibrium value. Note that if $a_l \gg b_l c_l$, then it is reasonable to assume that a_l and b_l will remain relatively constant for large changes in c_l .

As explained in Chapter 3, the system of Equations (D.1) can be “stiff”. Physically, this could happen in situations where the solution is changing slowly or is nearly asymptotic yet the time constants are prohibitively small. This occurs when the

production and loss rates are large, nearly equal, and there is strong coupling between the equations. The asymptotic integration method uses a predictor-corrector scheme that takes into consideration the stiffness of the Equations (D.1) while conducting the integration process.

In the method, for efficiency, an initial time step is chosen which approximates the time step that will be determined after convergence of the predictor-corrector scheme. This initial time step is chosen without regard to the “stiffness” of the equations and is determined such that none of the variables will change by more than a prescribed amount. The initial time step is chosen while considering (D.1) and (D.2),

$$\Delta t = \varepsilon_1 \min \left(\frac{c_l}{R_l} \right) ; \quad l = 1, \dots, N \quad (\text{D.3})$$

or if $a_l \gg b_l c_l$ then

$$\Delta t = \varepsilon_1 \min \left(\frac{1}{b_l} \right) ; \quad l = 1, \dots, N \quad (\text{D.4})$$

In Equations (D.3) and (D.4), the minimum is taken over the whole set of equations and ε_1 is a scale factor. In this work, ε_1 was chosen to be 1×10^{-3} . Note that (D.4) is needed when the initial conditions for some species are unknown or are set to zero.

The time step chosen by the above equations may be larger than some or all of the equilibrium times, in which case the corresponding equations would be classified as stiff. Accordingly, all of the equations are separated into two classes, stiff and non-stiff, by comparing the values of Δt and b_l . The two types of equations are then integrated by separate predictor-corrector schemes. An asymptotic integration procedure is used for the equations that are determined to be stiff.

The predictor part of the step is performed as follows:

$$\text{Non-stiff:} \quad c_l(1) = c_l(0) + \Delta t R_l(0) \quad ; \quad l = 1, \dots, N \quad (\text{D.5})$$

and

$$\text{Stiff:} \quad c_l(1) = c_l(0) + \frac{\Delta t R_l(0)}{1 + \Delta t R_l(0)} \quad ; \quad l = 1, \dots, N \quad (\text{D.6})$$

where $R_l(0) = R_l[t(0), c_l(0)]$, $c_l(0)$ is the initial value at $t(0)$, and $c_l(1)$ is the result of the predictor step.

The corrector formulae are:

$$\text{Non-stiff:} \quad c_l(k+1) = c_l(0) + \frac{\Delta t}{2} [R_l(0) + R_l(k)] \quad ; \quad l = 1, \dots, N \quad (\text{D.7})$$

$$\text{Stiff:} \quad c_l(k+1) = c_l(0) + \frac{2\Delta t [a_l(k) - b_l(0)c_l(0) + R_l(0)]}{4 + \Delta t [b_l(k) + b_l(0)]} \quad ; \quad l = 1, \dots, N \quad (\text{D.8})$$

where $c_l(k)$ is the k^{th} iterated value of c_l , or an approximation to $c_l[t(0) + \Delta t]$ and $R_l(k) = R_l[t(0) + \Delta t, c_l(k)]$.

By comparing $c_l(k+1)$ with $c_l(k)$ on successive iterations using the relative error criterion ε ,

$$\max \left[\frac{|c_l(k+1) - c_l(k)|}{c_l(k+1)} \right] \leq \varepsilon \quad (\text{D.9})$$

the convergence of each of the individual equations can be determined. Generally, ε is $O(10^{-3})$. In this work, ε was set to be 5×10^{-3} .

In practice, c_i is constrained by a minimum value when c_i is decaying exponentially toward zero. This lower bound must be selected to ensure that its value does not affect the physics of the problem but yet decouples the equation from accurate integration. Decoupling is accomplished by avoiding the application of (D.9) to those equations that have decayed to values corresponding to their lower bounds. Convergence for these equations is then trivial and does not affect the size of the time step. In this work, the lower bound used is 1×10^{-5} .

In practical implementation, for efficient computation, the number of corrector iterations is kept small; in this work this was kept at 10. If satisfactory convergence of all equations has not been obtained, the corrector step is started over with a smaller time step. In this work, on encountering non-convergence, the time step was decreased by 30% for the next corrector iteration.

References

1. Young, T.R., "CHEMEQ - A Subroutine for Solving Stiff Ordinary Differential Equations," NRL Memorandum Report 4091, 1980, Naval Research Laboratory, Washington, D.C., 49 pp.
2. Young, T.R. and Boris, J.P., "A Numerical Technique for Solving Stiff Ordinary Differential Equations Associated with the Chemical Kinetics of Reactive-Flow Problems," J. Physical Chemistry, 81 1977, 2424-2427.