Wasteload Allocation Study Tampa Bay, Florida Vol. II Water Quality Model Doc.

B. E. Ross  
*University of South Florida*

M. A. Ross  
*University of South Florida*

P. D. Jerkins  
*University of South Florida*

Follow this and additional works at: [http://scholarcommons.usf.edu/basgp_report](http://scholarcommons.usf.edu/basgp_report)

Part of the [Environmental Indicators and Impact Assessment Commons](http://scholarcommons.usf.edu/)

**Scholar Commons Citation**
[http://scholarcommons.usf.edu/basgp_report/166](http://scholarcommons.usf.edu/basgp_report/166)

This Statistical Report is brought to you for free and open access by the Tampa Bay Area Study Group Project at Scholar Commons. It has been accepted for inclusion in Reports by an authorized administrator of Scholar Commons. For more information, please contact scholarcommons@usf.edu.
Wasteload Allocation Study

VOLUME II:
DOCUMENTATION OF THE TWO-DIMENSIONAL
TRANSPORT AND ECOLOGIC MODELS

by

B. E. Ross, M. A. Ross, P. D. Jerkins
Center for Mathematical Models
Civil Engineering and Mechanics
University of South Florida

for

Florida Department of Environmental Regulation
Grant No: DER WM#54 12-2104-059
Tallahassee, Florida

March 1984
ACKNOWLEDGMENTS

Credit for authorship of this report is of a much expanded list than that of the principal authors shown. Too many students have contributed countless hours to vital components of the models and to this report to individually list here. Many are credited in the historic review (1.2) included in the first chapter. No less reward should be given to all of the research team (more than twenty qualified professionals and students) compiled during this contract period, their dedication is sincerely appreciated.

Special note must be made of the following individuals: Ms. Paula Jerkins for her continued dedication; she remains the single most valuable research assistant in the 15 plus years of modeling bay area waters. Special thanks go to Henry Dorzback, Louis McTague, and Larry Sanders for their contributions. Diane Goble was invaluable in the overall preparation of this document. Jim Devine contributed greatly to the hydraulic model documentation and numerical considerations. Scott McClelland's, Ray Walton's, and John Hartigan's review and comments of the rough draft helped locate so many of the tedious typographical and oversight errors. And, last, the inspiration, experience, and support given by my father, Bernard E. Ross, over so many years has made co-authorship of this report something very special.
## TABLE OF CONTENTS

**VOLUMES I-V WASTELoad ALLOCATION DOCUMENTATION**

### Vol I - The Hydraulic Model

<table>
<thead>
<tr>
<th>Preliminaries</th>
<th>page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acknowledgments</td>
<td>i</td>
</tr>
<tr>
<td>Table of Contents</td>
<td>ii</td>
</tr>
<tr>
<td>Preface</td>
<td>iii</td>
</tr>
<tr>
<td>List of Figures</td>
<td>v</td>
</tr>
<tr>
<td>List of Tables</td>
<td>vi</td>
</tr>
<tr>
<td>List of Symbols</td>
<td>vii</td>
</tr>
</tbody>
</table>

| Chapter 1 - Background               | 1    |
| 1.1 Introduction                     | 1    |
| 1.2 Philosophy (behind model development) | 4  |
| 1.3 History                          | 6    |

| Chapter 2 - Hydraulic Model: Formulation | 8 |
| 2.1 Equations of Motion               | 8 |
| 2.2 Boundary Conditions               | 12 |
| 2.3 Program Input                     | 18 |
| 2.3.1 Input Variable Definition      | 24 |
| 2.4 Program Output                   | 30 |
| 2.4.1 Mapped Output                  | 30 |
| 2.4.2 Other Output                   | 31 |

| Chapter 3 - Numerical Considerations | 33 |
| 3.1 Discussion                       | 33 |
| 3.2 Solution Technique               | 35 |
| 3.3 Stability Criteria               | 38 |

| List of References                   | 39 |

<table>
<thead>
<tr>
<th>Appendices</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>A. Theory, Derivation of Equations</td>
<td>43</td>
</tr>
<tr>
<td>B. Program Listing - Hydraulic Model</td>
<td>53</td>
</tr>
</tbody>
</table>

### Vol II - The Transport Models

| Introduction to Volume II            | 1   |

| Chapter 4 - Particle Transport Models | 2   |
| 4.1 Non-Bouyant Particles            | 2   |
| 4.1.1 Suspension                     | 3   |
| 4.1.2 Scour & Deposition             | 5   |
| 4.2 Bouyant Particle (Oilspill) Modeling | 11 |

| Chapter 5 - Thermal Transport Model  | 12  |
| 5.1 The Energy Budget                | 12  |
| 5.2 Input Variable Definition - Thermal Model | 27 |
| 5.3 Output - Thermal Bay Model       | 31  |
8.3 - Nutrient Fluxes
   8.3.1 - The Flux of Nutrients Through the Top Surface of Tampa Bay 15
   8.3.2 - The Flux of Nutrients Through the Edges of Tampa Bay 16
   8.3.3 - The Exchange of Nutrients Through the Bottom of Tampa Bay 17
   8.3.4 - Light Flux 18
8.4 - The Box Model vs. the 2-D Ecological Model of Tampa Bay 22
8.5 - Summary 22

List of References 24

Appendix

F. Seagrass/Macroalgae Sampling Study 26

Vol V

Results - Predictive Simulation Output

Additional Documents

"Users' Guide - WLA Models"
"A Statistical Study on Factors Affecting Chlorophyll a Levels in the Tampa Bay Area"
This five volume text represents a complete documentation of the modeling portion of a wasteload allocation (WLA) study of Tampa Bay, Florida. The study was administered by the Florida Department of Environmental Regulation (FDER) using federal Environmental Protection Agency (EPA) section 205j funds. The wasteload allocations are to be used for 201 facilities planning and permitting purposes. Modeling was conducted by the University of South Florida (USF) under research grant number DER# WM-54, USF# 12-2104-059. Hydrodynamic and water quality field data collection was performed by Continental Shelf Associates, Inc. (CSA). Engineering work, point and nonpoint source evaluation, was contracted with Camp, Dresser and McKee, Inc. (CDM). Seagrassbed analysis was conducted by CSA with assistance from Mangrove Systems, Inc.

Included in these four volumes is a brief summary of the functions and purpose of various hybrid two-dimensional models that have evolved from the initial 2-D hydraulic model developed at USF in 1970.

It must be emphasized that the scope and underlying intention of this report has been two-fold. The first purpose has been to fully document USF's involvement in the WLA study of Tampa Bay. The other purpose for this report has been to provide, perhaps for the first time in one set of documents, a comprehensive survey of philosophy and developmental extent behind fifteen years of directed research aimed at obtaining a valuable, general-purpose modeling tool to be used in decision making of this type.
LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>I - 2</td>
</tr>
<tr>
<td>1.2</td>
<td>I - 3</td>
</tr>
<tr>
<td>2.1</td>
<td>I - 19</td>
</tr>
<tr>
<td>2.2</td>
<td>I - 21</td>
</tr>
<tr>
<td>2.3</td>
<td>I - 23</td>
</tr>
<tr>
<td>2.4</td>
<td>I - 32</td>
</tr>
<tr>
<td>3.1</td>
<td>I - 34</td>
</tr>
<tr>
<td>4.1</td>
<td>II - 4</td>
</tr>
<tr>
<td>4.2</td>
<td>II - 6</td>
</tr>
<tr>
<td>4.3</td>
<td>II - 7</td>
</tr>
<tr>
<td>4.4</td>
<td>II - 8</td>
</tr>
<tr>
<td>4.5</td>
<td>II - 9</td>
</tr>
<tr>
<td>4.6</td>
<td>II - 10</td>
</tr>
<tr>
<td>6.1</td>
<td>II - 36</td>
</tr>
<tr>
<td>6.2</td>
<td>II - 55</td>
</tr>
<tr>
<td>C.1</td>
<td>II - 58</td>
</tr>
<tr>
<td>C.2</td>
<td>II - 65</td>
</tr>
<tr>
<td>C.3</td>
<td>II - 72</td>
</tr>
<tr>
<td>C.4</td>
<td>II - 76</td>
</tr>
<tr>
<td>7.1</td>
<td>III - 4</td>
</tr>
<tr>
<td>7.2</td>
<td>III - 5</td>
</tr>
<tr>
<td>7.3</td>
<td>III - 6</td>
</tr>
<tr>
<td>7.4</td>
<td>III - 7</td>
</tr>
<tr>
<td>7.5</td>
<td>III - 8</td>
</tr>
<tr>
<td>7.6</td>
<td>III - 10</td>
</tr>
<tr>
<td>7.7</td>
<td>III - 11</td>
</tr>
<tr>
<td>7.8</td>
<td>III - 12</td>
</tr>
<tr>
<td>7.9</td>
<td>III - 13</td>
</tr>
<tr>
<td>7.10</td>
<td>III - 14</td>
</tr>
<tr>
<td>7.11</td>
<td>III - 15</td>
</tr>
<tr>
<td>7.12</td>
<td>III - 16</td>
</tr>
<tr>
<td>7.13</td>
<td>III - 17</td>
</tr>
<tr>
<td>7.14</td>
<td>III - 19</td>
</tr>
<tr>
<td>7.15</td>
<td>III - 21</td>
</tr>
<tr>
<td>7.16</td>
<td>III - 22</td>
</tr>
<tr>
<td>7.17</td>
<td>III - 23</td>
</tr>
<tr>
<td>7.18</td>
<td>III - 24</td>
</tr>
<tr>
<td>7.19</td>
<td>III - 25</td>
</tr>
<tr>
<td>7.20</td>
<td>III - 26</td>
</tr>
</tbody>
</table>
### LIST OF FIGURES (continued)

<table>
<thead>
<tr>
<th>Figure</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.21</td>
<td>Courtley Campbell Velocities - 7/7-7/8 (Magnitude)</td>
</tr>
<tr>
<td>7.22</td>
<td>Courtley Campbell Velocities - 7/7-7/8 (Direction)</td>
</tr>
<tr>
<td>7.23</td>
<td>Courtley Campbell Velocities - 7/20-7/21 (Magnitude)</td>
</tr>
<tr>
<td>7.24</td>
<td>Courtley Campbell Velocities - 7/20-7/21 (Direction)</td>
</tr>
<tr>
<td>7.25</td>
<td>Courtley Campbell Velocities - 8/3-8/4 (Magnitude)</td>
</tr>
<tr>
<td>7.26</td>
<td>Courtley Campbell Velocities - 8/3-8/4 (Direction)</td>
</tr>
<tr>
<td>7.27</td>
<td>Howard Franklin Velocities - 7/7-7/8 (Magnitude)</td>
</tr>
<tr>
<td>7.28</td>
<td>Howard Franklin Velocities - 7/7-7/8 (Direction)</td>
</tr>
<tr>
<td>7.29</td>
<td>Howard Franklin Velocities - 7/20-7/21 (Magnitude)</td>
</tr>
<tr>
<td>7.30</td>
<td>Howard Franklin Velocities - 7/20-7/21 (Direction)</td>
</tr>
<tr>
<td>7.31</td>
<td>Howard Franklin Velocities - 8/3-8/4 (Magnitude)</td>
</tr>
<tr>
<td>7.32</td>
<td>Howard Franklin Velocities - 8/3-8/4 (Direction)</td>
</tr>
<tr>
<td>7.33</td>
<td>Hillsborough Bay No. Velocities - 7/7-7/8 (Magnitude)</td>
</tr>
<tr>
<td>7.34</td>
<td>Hillsborough Bay No. Velocities - 7/7-7/8 (Direction)</td>
</tr>
<tr>
<td>7.35</td>
<td>Hillsborough Bay No. Velocities - 7/20-7/21 (Magnitude)</td>
</tr>
<tr>
<td>7.36</td>
<td>Hillsborough Bay No. Velocities - 7/20-7/21 (Direction)</td>
</tr>
<tr>
<td>7.37</td>
<td>Hillsborough Bay No. Velocities - 8/3-8/4 (Magnitude)</td>
</tr>
<tr>
<td>7.38</td>
<td>Hillsborough Bay No. Velocities - 8/3-8/4 (Direction)</td>
</tr>
<tr>
<td>7.39</td>
<td>Hillsborough Bay No. Central Vel. - 7/7-7/8 (Magnitude)</td>
</tr>
<tr>
<td>7.40</td>
<td>Hillsborough Bay No. Central Vel. - 7/7-7/8 (Direction)</td>
</tr>
<tr>
<td>7.41</td>
<td>Hillsborough Bay No. Central Vel. - 7/20-7/21 (Magnitude)</td>
</tr>
<tr>
<td>7.42</td>
<td>Hillsborough Bay No. Central Vel. - 7/20-7/21 (Direction)</td>
</tr>
<tr>
<td>7.43</td>
<td>Hillsborough Bay No. Central Vel. - 8/3-8/4 (Magnitude)</td>
</tr>
<tr>
<td>7.44</td>
<td>Hillsborough Bay No. Central Vel. - 8/3-8/4 (Direction)</td>
</tr>
<tr>
<td>7.45</td>
<td>Hillsborough Bay So. Central Vel. - 7/7-7/8 (Magnitude)</td>
</tr>
<tr>
<td>7.46</td>
<td>Hillsborough Bay So. Central Vel. - 7/7-7/8 (Direction)</td>
</tr>
<tr>
<td>7.47</td>
<td>Hillsborough Bay So. Central Vel. - 7/20-7/21 (Magnitude)</td>
</tr>
<tr>
<td>7.48</td>
<td>Hillsborough Bay So. Central Vel. - 7/20-7/21 (Direction)</td>
</tr>
<tr>
<td>7.49</td>
<td>Hillsborough Bay So. Central Vel. - 8/3-8/4 (Magnitude)</td>
</tr>
<tr>
<td>7.50</td>
<td>Hillsborough Bay So. Central Vel. - 8/3-8/4 (Direction)</td>
</tr>
<tr>
<td>7.51</td>
<td>Hillsborough Bay So. Velocities - 7/7-7/8 (Magnitude)</td>
</tr>
<tr>
<td>7.52</td>
<td>Hillsborough Bay So. Velocities - 7/7-7/8 (Direction)</td>
</tr>
<tr>
<td>7.53</td>
<td>Hillsborough Bay So. Velocities - 7/20-7/21 (Magnitude)</td>
</tr>
<tr>
<td>7.54</td>
<td>Hillsborough Bay So. Velocities - 7/20-7/21 (Direction)</td>
</tr>
<tr>
<td>7.55</td>
<td>Hillsborough Bay So. Velocities - 8/3-8/4 (Magnitude)</td>
</tr>
<tr>
<td>7.56</td>
<td>Hillsborough Bay So. Velocities - 8/3-8/4 (Direction)</td>
</tr>
<tr>
<td>7.57</td>
<td>Tampa Bay Central Velocities - 7/7-7/8 (Magnitude)</td>
</tr>
<tr>
<td>7.58</td>
<td>Tampa Bay Central Velocities - 7/7-7/8 (Direction)</td>
</tr>
<tr>
<td>7.59</td>
<td>Tampa Bay Central Velocities - 7/20-7/21 (Magnitude)</td>
</tr>
<tr>
<td>7.60</td>
<td>Tampa Bay Central Velocities - 7/20-7/21 (Direction)</td>
</tr>
<tr>
<td>7.61</td>
<td>Tampa Bay Central Velocities - 8/3-8/4 (Magnitude)</td>
</tr>
<tr>
<td>7.62</td>
<td>Tampa Bay Central Velocities - 8/3-8/4 (Direction)</td>
</tr>
<tr>
<td>7.63</td>
<td>Skyway Bridge Velocities - 7/7-7/8 (Magnitude)</td>
</tr>
<tr>
<td>7.64</td>
<td>Skyway Bridge Velocities - 7/7-7/8 (Direction)</td>
</tr>
<tr>
<td>7.65</td>
<td>Skyway Bridge Velocities - 7/20-7/21 (Magnitude)</td>
</tr>
<tr>
<td>7.66</td>
<td>Skyway Bridge Velocities - 7/20-7/21 (Direction)</td>
</tr>
<tr>
<td>7.67</td>
<td>Skyway Bridge Velocities - 8/3-8/4 (Magnitude)</td>
</tr>
<tr>
<td>7.68</td>
<td>Skyway Bridge Velocities - 8/3-8/4 (Direction)</td>
</tr>
<tr>
<td>7.69</td>
<td>Tide Heights - 1/8-Mile Model - Ballast Point - 7/7-7/8</td>
</tr>
<tr>
<td>7.70</td>
<td>Tide Heights - 1/8-Mile Model - Ballast Point - 7/20-7/21</td>
</tr>
<tr>
<td>7.71</td>
<td>Tide Heights - 1/8-Mile Model - Ballast Point - 8/3-8/4</td>
</tr>
<tr>
<td>Figure</td>
<td>Location</td>
</tr>
<tr>
<td>--------</td>
<td>----------</td>
</tr>
<tr>
<td>7.72</td>
<td>Hillsborough Bay No. Velocities(1/8-Mile Model)-7/7-7/8</td>
</tr>
<tr>
<td>7.73</td>
<td>Hillsborough Bay No: Velocities(1/8-Mile Model)-7/7-7/8</td>
</tr>
<tr>
<td>7.74</td>
<td>Hillsborough Bay No. Cent. Vels.(1/8-Mile Model)-7/7-7/8</td>
</tr>
<tr>
<td>7.75</td>
<td>Hillsborough Bay No. Cent. Vels.(1/8-Mile Model)-7/7-7/8</td>
</tr>
<tr>
<td>7.76</td>
<td>Hillsborough Bay So. Cent. Vels.(1/8-Mile Model)-7/7-7/8</td>
</tr>
<tr>
<td>7.77</td>
<td>Hillsborough Bay So. Cent. Vels.(1/8-Mile Model)-7/7-7/8</td>
</tr>
<tr>
<td>7.78</td>
<td>Hillsborough Bay No. Velocities(1/8-Mile Model)-7/20-7/21</td>
</tr>
<tr>
<td>7.79</td>
<td>Hillsborough Bay No. Velocities(1/8-Mile Model)-7/20-7/21</td>
</tr>
<tr>
<td>7.84</td>
<td>Hillsborough Bay North Velocities(1/8-Mile Model)-8/3-8/4</td>
</tr>
<tr>
<td>7.85</td>
<td>Hillsborough Bay North Velocities(1/8-Mile Model)-8/3-8/4</td>
</tr>
<tr>
<td>7.87</td>
<td>Hillsborough Bay No. Cent. Vels.(1/8-Mile Model)-8/3-8/4</td>
</tr>
<tr>
<td>7.88</td>
<td>Hillsborough Bay So. Cent. Vels.(1/8-Mile Model)-8/3-8/4</td>
</tr>
<tr>
<td>7.89</td>
<td>Hillsborough Bay So. Cent. Vels.(1/8-Mile Model)-8/3-8/4</td>
</tr>
<tr>
<td>7.90</td>
<td>Major Nutrient Source Discharges in Tampa Bay</td>
</tr>
<tr>
<td>7.91</td>
<td>Map of Measured Benthic Oxygen Demand</td>
</tr>
<tr>
<td>7.92</td>
<td>Map of Orthophosphate Uptake</td>
</tr>
<tr>
<td>7.93</td>
<td>Map of Ammonia Uptake</td>
</tr>
<tr>
<td>7.94</td>
<td>Water Quality Parameters-Safety Harbor - 7/7-7/8</td>
</tr>
<tr>
<td>7.95</td>
<td>Water Quality Parameters-Campbell Causeway - 7/7-7/8</td>
</tr>
<tr>
<td>7.96</td>
<td>Water Quality Parameters-Franklin Bridge - 7/7-7/8</td>
</tr>
<tr>
<td>7.97</td>
<td>Water Quality Parameters-Hills. Bay North - 7/7-7/8</td>
</tr>
<tr>
<td>7.100</td>
<td>Water Quality Parameters-Hills. Bay South - 7/7-7/8</td>
</tr>
<tr>
<td>7.101</td>
<td>Water Quality Parameters-Tampa Bay Central - 7/7-7/8</td>
</tr>
<tr>
<td>7.102</td>
<td>Water Quality Parameters-Skyway Bridge - 7/7-7/8</td>
</tr>
<tr>
<td>7.103</td>
<td>Water Quality Parameters-Safety Harbor - 7/20-7/21</td>
</tr>
<tr>
<td>7.104</td>
<td>Water Quality Parameters-Campbell Causeway - 7/20-7/21</td>
</tr>
<tr>
<td>7.105</td>
<td>Water Quality Parameters-Franklin Bridge - 7/20-7/21</td>
</tr>
<tr>
<td>7.110</td>
<td>Water Quality Parameters-Tampa Bay Central - 7/20-7/21</td>
</tr>
<tr>
<td>7.111</td>
<td>Water Quality Parameters-Skyway Bridge - 7/20-7/21</td>
</tr>
<tr>
<td>7.112</td>
<td>Water Quality Parameters-Safety Harbor - 8/3-8/4</td>
</tr>
<tr>
<td>7.113</td>
<td>Water Quality Parameters-Campbell Causeway - 8/3-8/4</td>
</tr>
<tr>
<td>7.114</td>
<td>Water Quality Parameters-Franklin Bridge - 8/3-8/4</td>
</tr>
<tr>
<td>7.119</td>
<td>Water Quality Parameters-Tampa Bay Central - 8/3-8/4</td>
</tr>
<tr>
<td>7.120</td>
<td>Water Quality Parameters-Skyway Bridge - 8/3-8/4</td>
</tr>
<tr>
<td>Figure</td>
<td>Location</td>
</tr>
<tr>
<td>--------</td>
<td>----------</td>
</tr>
<tr>
<td>E.1 SOD/NER Sampling Sites - Hillsborough Bay</td>
<td>III - 154</td>
</tr>
<tr>
<td>E.2 SOD/NER Sampling Sites - Tampa Bay</td>
<td>III - 158</td>
</tr>
<tr>
<td>8.1 Ecology of Tampa Bay</td>
<td>IV - 3</td>
</tr>
<tr>
<td>8.2 Ecology of Tampa Bay (Revised)</td>
<td>IV - 4</td>
</tr>
<tr>
<td>8.3 Major Storages of Nutrients</td>
<td>IV - 7</td>
</tr>
<tr>
<td>8.4 Sewage Discharges in Tampa Bay</td>
<td>IV - 9</td>
</tr>
<tr>
<td>8.5 Schematic of Box Model</td>
<td>IV - 19</td>
</tr>
</tbody>
</table>
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Location</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.1</td>
<td>Reflected Solar Radiation Constants</td>
<td>II - 20</td>
</tr>
<tr>
<td>E.1</td>
<td>Barrier Definition for the Tampa Bay Model</td>
<td>III - 143</td>
</tr>
<tr>
<td>E.2</td>
<td>Point Source Data for Tampa Bay Model</td>
<td>III - 149</td>
</tr>
<tr>
<td>E.3</td>
<td>Point Source Data for Hillsborough Bay Model</td>
<td>III - 149</td>
</tr>
<tr>
<td>E.4</td>
<td>Nonpoint Sources for Tampa Bay (6/24-7/8)</td>
<td>III - 150</td>
</tr>
<tr>
<td>E.5</td>
<td>Nonpoint Sources for Tampa Bay (7/9-7/21)</td>
<td>III - 151</td>
</tr>
<tr>
<td>E.6</td>
<td>Nonpoint Sources for Tampa Bay (7/22-8/4)</td>
<td>III - 151</td>
</tr>
<tr>
<td>E.7</td>
<td>Nonpoint Sources for Hillsborough Bay (6/24-7/8)</td>
<td>III - 152</td>
</tr>
<tr>
<td>E.8</td>
<td>Nonpoint Sources for Hillsborough Bay (7/9-7/21)</td>
<td>III - 153</td>
</tr>
<tr>
<td>E.9</td>
<td>Nonpoint Sources for Hillsborough Bay (7/22-8/4)</td>
<td>III - 153</td>
</tr>
<tr>
<td>E.10</td>
<td>SOD Rates and Means - Hillsborough Bay</td>
<td>III - 155</td>
</tr>
<tr>
<td>E.11</td>
<td>NER Rates and Means - Hillsborough Bay</td>
<td>III - 156</td>
</tr>
<tr>
<td>E.12</td>
<td>Replicant NER Rates and Means - Hillsborough Bay</td>
<td>III - 157</td>
</tr>
<tr>
<td>E.13</td>
<td>NER Rates, Means, Ranges - Tampa Bay</td>
<td>III - 159</td>
</tr>
<tr>
<td>E.14</td>
<td>Replicant NER Rates - Tampa Bay</td>
<td>III - 160</td>
</tr>
<tr>
<td>E.15</td>
<td>Replicant SOD Rates - Tampa Bay</td>
<td>III - 161</td>
</tr>
<tr>
<td>E.16</td>
<td>Coefficients Used in the Calibration Runs</td>
<td>III - 162</td>
</tr>
<tr>
<td>8.1</td>
<td>Storage of Nutrients</td>
<td>IV - 14</td>
</tr>
<tr>
<td>8.2</td>
<td>Light Extinction for Various Stations in the Bay</td>
<td>IV - 21</td>
</tr>
<tr>
<td>8.3</td>
<td>Flux of Nutrients Through the Boundaries</td>
<td>IV - 23</td>
</tr>
</tbody>
</table>
LIST OF SYMBOLS

\( a_1, a_2 \) = Atmospheric transmission coefficient.
\( A \) = Constant dependent on the degree of cloudiness
\( A \) = Water surface area, \( \text{ft}^2 \)
\( A \) = Vertically averaged concentration of algae biomass, \( \text{mg/l} \)
\( B \) = Constant dependent on the degree of cloudiness
\( C \) = A measure of the proportional sky covered by clouds
\( C \) = Vertically averaged concentration of the constituent, \( \text{mg/l} \) or ppm
\( C \) = Vertically averaged concentration of the ultimate carbonaceous biochemical oxygen demand, BOD\(_C\), in \( \text{mg/l} \)
\( C_L \) = Concentration of chlorophyll \( a \), \( \text{mg/l} \)
\( C_P \) = Specific heat of water, 32.2 BTU/SLUG/DEG F.
\( C_S \) = Cloud cover factor
\( C_S \) = Concentration of BOD\(_C\) in source or sinks, \( \text{mg/l} \)
\( d \) = Total dust attenuation
\( D \) = The number of the day of the year
\( D \) = The water depth, \( \text{ft} \)
\( D_{\text{MAX}} \) = Maximum depth of the water
\( e_a \) = Emissivity of the atmosphere
\( e_a \) = Atmospheric vapor pressure, in mb
\( e_s \) = Vapor pressure of the water surface, mb
\( E \) = A constant = +1 for west longitude
\( E \) = -1 for east longitude
\( E_w \) = Emissivity of water, 0.97
\( E_T \) = Equation of time from a solar ephemeris
\( EX, EY \) = X and y direction dispersion coefficients, resp.
\( E_{\text{max}} \) = Maximum value of the dispersion coefficient (EX or EY)
\( E_{\text{min}} \) = Minimum value of the dispersion coefficient (EX or EY)
\( f \) = Friction factor
\( f_1 \) = Low-dissolved oxygen correction factor for the decay of BOD\(_C\)
\( f_2 \) = Low-dissolved oxygen correction factor for ammonia
\( F_r \) = Correction factor for diurnal exposure to radiation flux
\( g \) = Acceleration due to gravity, \( \text{ft/s}^2 \)
\( h \) = Height of water, \( \text{ft} \)
\( h_o \) = Extraterrestrial radiation flux
\( H \) = Height of water above the mean low water, \( \text{ft} \)
\( H_a \) = Long-wave atmospheric radiation
<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>H_{ar}</td>
<td>Reflected atmospheric radiation</td>
</tr>
<tr>
<td>H_{br}</td>
<td>Long-wave back radiation</td>
</tr>
<tr>
<td>H_{c}</td>
<td>Conduction heat transfer</td>
</tr>
<tr>
<td>H_{cs}</td>
<td>Clear sky short wave radiation received at the earth's surface</td>
</tr>
<tr>
<td>H_{e}</td>
<td>Evaporation</td>
</tr>
<tr>
<td>H_{m}</td>
<td>Heat flux terms</td>
</tr>
<tr>
<td>H_{s}</td>
<td>Short wave solar radiation</td>
</tr>
<tr>
<td>H_{sr}</td>
<td>Reflective solar radiation</td>
</tr>
<tr>
<td>H_{s}'</td>
<td>Algae-free secchi depth, ft.</td>
</tr>
<tr>
<td>H_{t}</td>
<td>Evaporation of heat transfer</td>
</tr>
<tr>
<td>I</td>
<td>Grid location in the horizontal, x-direction</td>
</tr>
<tr>
<td>I_{0}</td>
<td>Solar constant</td>
</tr>
<tr>
<td>J</td>
<td>Grid location in the vertical, y-direction</td>
</tr>
<tr>
<td>K</td>
<td>Surface heat exchange coefficient</td>
</tr>
<tr>
<td>K_{1}</td>
<td>First order decay rate of the constituents</td>
</tr>
<tr>
<td>K_{2}</td>
<td>Kinematic surface exchange coefficient</td>
</tr>
<tr>
<td>K_{n}</td>
<td>Nitrogen half-saturated constant for algae growth, mg/l</td>
</tr>
<tr>
<td>K_{p}</td>
<td>Phosphorus half-saturated constant for algae growth, mg/l</td>
</tr>
<tr>
<td>KE</td>
<td>Kinetic energy</td>
</tr>
<tr>
<td>L</td>
<td>Latent heat of vaporization, cal/gm</td>
</tr>
<tr>
<td>L_{l}</td>
<td>Latitude of local meridian</td>
</tr>
<tr>
<td>L_{s}</td>
<td>Latitude of standard meridian</td>
</tr>
<tr>
<td>M</td>
<td>Optical air mass</td>
</tr>
<tr>
<td>n</td>
<td>Manning's roughness coefficient</td>
</tr>
<tr>
<td>n_{1}</td>
<td>Outward normal to the edges of the surface</td>
</tr>
<tr>
<td>N_{1}</td>
<td>Vertically averaged concentration of organic nitrogen as nitrogen, mg/l</td>
</tr>
<tr>
<td>N_{1s}</td>
<td>Concentration of organic nitrogen in source or sink</td>
</tr>
<tr>
<td>N_{2}</td>
<td>Concentration of ammonia as nitrogen, mg/l.</td>
</tr>
<tr>
<td>N_{2s}</td>
<td>Concentration of ammonia in source or sink</td>
</tr>
<tr>
<td>N_{3}</td>
<td>Concentration of nitrogen as nitrogen, mg/l.</td>
</tr>
<tr>
<td>N_{3s}</td>
<td>Concentration of nitrogen in source or sink, mg/l.</td>
</tr>
<tr>
<td>O</td>
<td>Vertically averaged concentration of oxygen, mg/l.</td>
</tr>
<tr>
<td>O_{d}</td>
<td>Oxygen deficit concentration, mg/l.</td>
</tr>
<tr>
<td>O_{s}</td>
<td>Saturation concentration of oxygen, in mg/l, dependent on temperature and chlorine</td>
</tr>
<tr>
<td>P_{h}</td>
<td>Atmospheric pressure</td>
</tr>
<tr>
<td>P</td>
<td>Hydrostatic pressure</td>
</tr>
<tr>
<td>P_{1}</td>
<td>Vertically averaged concentration of organic phosphorus, mg/l.</td>
</tr>
</tbody>
</table>
\[ P_2 \] = Concentration of orthophosphate as phosphate, mg/l.

\[ P_{1s}, P_{2s} \] = Concentration of P in source or sink, mg/l.

\[ q \] = Amount of water evaporated in cm/day

\[ q_s \] = Specific discharge (discharge per unit surface area) of a given source or sink

\[ Q \] = Resultant transport, cfs-ft width

\[ Q \] = Square root \((U^2 + V^2)\)

\[ Q \] = Source of sink flows

\[ Q_c \] = Source rate of BOD_c from point and nonpoint sources and sinks

\[ Q_{n1} \] = Source rate of organic nitrogen from point and nonpoint sources and sinks

\[ Q_{n2}, Q_{n3} \] = Source rate if ammonia from point and nonpoint sources and sinks, mg/1-ft/sec.

\[ Q_{p1}, Q_{p2} \] = Source rate of \( P_2 \) from point and nonpoint sources and sinks, mg/l-ft/sec.

\[ Q_s \] = Source/sink inflows

\[ r \] = Ratio of the actual mean distance between the earth and the sun

\[ R \] = Sources and sinks

\[ R \] = Ratio of open area to total grid area

\[ R' \] = Time rate of a change of a source

\[ S \] = Salinity concentration, mg/l, ppm.

\[ S \] = Total number of sources

\[ S_{tb}, S_{te} \] = Standard times of the beginning and end, respectively of the interval chosen for temperature analysis

\[ S_{tr}, S_{ts} \] = Standard times of sunrise and sunset, respectively

\[ t_{b}, t_{e} \] = Hour angles corresponding to the beginning and end, respectively of any time interval between sunrise and sunset

\[ t' \] = naturally occurring water temperature

\[ t \] = Instantaneous time

\[ T \] = Vertically averaged water temperature

\[ T \] = Local water temperature in degrees C.

\[ T_a \] = Temperature of the air in degrees C.

\[ T_{af} \] = Temperature of the air in degrees F.

\[ T_e \] = Equilibrium temperature of the water

\[ T_s \] = Temperature of the source or sink

\[ T_{s} \] = Standard temperature in degrees C.

\[ T_w \] = Temperature of the water surface in degrees C.

\[ T_{wb} \] = Temperature of the wet bulb in degrees C.

\[ T_{wbf} \] = Temperature of the wet bulb in degrees F.

\[ u \] = Wind speed in mph

\[ U \] = Transport in the x direction

\[ U_p \] = Velocity of particle, \( p \), at time, \( t \), in the x direction
U(i,j) = Velocity at the grid location (i, j) in the x-direction
V,v = Transport in the y-direction
V' = Time rate differential of the velocity
V_0 = Average initial velocity
V_t = Velocity of particle, P, at time, t, in the y-direction
V(i,j) = Velocity at grid location (i, j) in the y-direction
w = Mean daily precipitable water content in the atmosphere
W_1 = 0.95
W_2 = 0.05
x = Horizontal cartesian coordinate, ft.
X = Grid location in the horizontal direction
X_P = Grid location of particle, P, in the horizontal direction
X_{p'} = New grid location of particle, p, in the horizontal direction
X_t = Variable dependent upon local temperature
X_{ts} = Variable dependent upon standard temperature
y = Vertical cartesian coordinate, ft.
Y = Grid location in the vertical direction
Y_P = Grid location of particle, P, in the vertical direction
Y_{p'} = New grid location of particle, p, in the vertical direction
Z = Upward coordinate with origin at mean sea level
Z,z = Altitude of the site in meters

GREEK SYMBOLS

\( \alpha_{CL} \) = Ratio of chlorophyll a to total algae biomass (mg Chla/mg A)
\( \alpha_N \) = Fraction of algae biomass which is N (mg N/mg A)
\( \alpha_{OO} \) = Oxygen production per unit of algae growth (photosynthesis) (mg O/mg A)
\( \alpha_{ON2} \) = Oxygen uptake per unit of ammonia oxidation (mg O/mg N)
\( \alpha_{OR} \) = Oxygen uptake per unit of algae respiration (mg O/mg A)
\( \alpha_P \) = Fraction of algae biomass which is P (mg P/mg A)
\( \alpha_s \) = Sun's attitude in radians (II-17), degrees
\( \beta_C \) = First-order decay rate of BOD_c due to oxidation (1/sec)
\( \beta_{N_1} \) = First order decay rate of organic nitrogen to ammonia
\( \beta_{N_2} \) = Rate of biological oxidation of ammonia to nitrate (1/sec.)
\( \beta_0 \) = Reaeration rate of oxygen (1/sec.)
\( \beta_{OW} \) = Wind coefficient for reaeration
\( \beta_{P_1} \) = First order decay rate of organic phosphorus to orthophosphate
\( \delta \) = Ammonia preferential consumption factor
$\Delta H$ = Summation of all head flux terms
$\Delta s$ = Cell size of the model
$\Delta t_{or}$ = Hour corrections for obstructionn of sunrise
$\Delta t_{os}$ = Hour corrections for obstruction of sunset
$\Delta t_s$ = Difference between standard and local civil time
$\delta$ = Declination of the Sun
$\eta_{N_2}$ = Benthic source rate for $N_2$; variable with location
$\eta_{N_3}$ = Benthic source rate of nitrate (mg/l N-ft/sec)
$\eta_o$ = Benthic demand for oxygen (mg/l O - H/day)
$\theta$ = An empirical constant for temperature dependence function
$\theta_s$ = Excess water temperature due to heat source (power plants)
$\theta_d$ = Excess temperature of the given source or sink s
$\theta_{dF}$ = Mean daily dew-point temperature in degrees C
$\lambda$ = Mean daily dew-point temperature in degrees F
$\mu$ = Specific growth rate of algae (1/sec)
$\mu'$ = Maximum specific growth rate of algae
$\rho$ = Density of the water (slugs)
$\rho_s$ = Density of the fluid (slugs)
$\rho_r$ = Respiration rate of algae
$\rho_{G}$ = Grazing rate for algae (1/sec)
$\sigma$ = Stefan-Boltzman constant, $1.18 \times 10^{-7}$ cal/sq-cm/day/deg K
$\sigma_o$ = Settling rate of algae
$\sigma_{oc}$ = Settling rate of $BOD_5$ (1/sec)
$\sigma_{N_1}$ = Settling rate for organic nitrogen (ft/sec)
$\sigma_{N_2}$ = Settling rate for ammonia (ft/sec)
$\sigma_{P_1}$ = Settling rate of organic phosphorus (ft/sec)
$\phi$ = Latitude of the site
$Z_0$ = Bottom elevation with respect to MSL (mg/l $O_2$ - ft/day)
This second volume is a documentation of various transport models used at USF. Included, is a comprehensive documentation of the water quality model used in the WLA of Tampa Bay. Documentation of the particle transport and thermal models has been included in addition to the water quality ecological model because the authors wished to provide a historic listing of the more recent or popular transport models relevant to studies in this area.

Chapter 4 is a discussion of the particle transport models including the oilspill and sedimentation model. Chapter 5 is a documentation of the thermal transport model. Chapter 6 is the detailed documentation of the water quality-ecological model used in the Tampa Bay WLA. Many of the rates, constants, and other parameters were confirmed, redefined or newly derived specifically in this study. Emphasis has been placed on using accepted and/or documented water quality parameter relationships where statistical analysis of local data showed non-realistic correlations. Many of these default values were taken out of EPA/TETRA TECH "Rates, Constants, and Kinetic Formulations in Surface Water Quality Modeling", EPA-600/3-78-105 (1978), as specified in the contract with FDER. A special report by Gaitho [1] summarizes the results of the statistical analysis performed on local water quality data as part of this contract.
A number of subroutines have been developed to study the motion of suspended substances. Cote [2] documented some of the original development of two of the subroutines. They have been designed to be used in conjunction with application of the hydraulic model. Because of the forces acting on each, discussion of particle transport is divided into two categories: non-bouyant (suspended) and bouyant particles.

Subroutine TRACK calculates the position of "particles" as they move with the tidal action. The philosophy behind suspended particle "tracking" is given in section 4.1. The additional considerations given for calculation of bouyant particle transport is outlined in section 4.2.

Subroutine TRACE is used to graphically display the results with the help of the canned, DIPLOT, subroutine package available on the Prime USF CEM computer.

4.1 - Non-Bouyant Particles

When particles are of the size and density (such as fine sands and clays) to be easily suspended and dispersed throughout the water column, calculation of their movement in a tidal driven embayment is straightforward. Suspended particle transport is primarily governed by the net water transport at that location. In cases concerning scour and deposition, velocity magnitudes, in addition to particle size, are also important factors.
4.1.1 - Suspension

The type of particle in this category can be any particle having neutrally buoyant mass and can even be a water particle. The tracking of water particles is common and is useful in determining the destinations of flows. It can also be useful in studying tidal exchange rates.

The philosophy, in developing the model, is that the neutrally suspended particle moves with the same velocity and in the same direction as the water which surrounds it. Each particle is spacially located not only by its grid location (i,j) but also by the distance in feet from the upper left hand corner of the grid in the X and Y directions (see Figure 4.1). Subroutine TRACK calculates the instantaneous velocity of each particle by a linear interpolation of the calculated velocities at the edges of each element. That is, the vector components for two-dimensional displacement of the velocity of a particle \( P \) at instantaneous time \( t \) is;

\[
\begin{align*}
    u^t_P &= U(i-1,j) + \frac{U(i,j) - U(i-1,j)}{dx} x^*_P \\
    v^t_P &= V(1,j-1) + \frac{V(i,j) - V(i,j-1)}{dy} y^*_P
\end{align*}
\]

The particle moves at that velocity for the length of time specified by the time-step, \( \Delta t \), and a new position \( (x^t_P, y^t_P) \) is calculated. The velocity at the new position is used to move the particle for the next time step and the process continues for as long as the particle is to be tracked.
FIGURE 4.1: Spatial Location of a Particle
A great deal of insight into the "flushing" of an embayment can be obtained by tracking a large number of particles at the same time. "Flush ing" can best be described as being the length of time necessary for a percentage of water to leave (and not return to) the area of interest. In the series of plots shown in Figures 4.2 - 4.6, particles which were initially placed in all grid locations in the Tampa Bay 1/2 mile model (Figure 4.2) were allowed to move with the currents. The locations at the end of 28 days (Figure 4.3) shows that many have washed out into the Gulf of Mexico, while others are "bunching-up" and still remain. Plots for locations at 42, 56 and 70 days follow in Figures 4.4, 4.5, and 4.6 respectively. It can be seen that "flushing" is not a rapid process but is a slow dilution that could take a great deal of time.

4.1.2 - Scour & Deposition

It is well accepted that scour and deposition is dependent on the size and density of the particles and on the water velocity and viscosity at the bottom. Numerous authors of common hydraulic texts, including Chow [3] and Streeter [4] give relationships for calculation in open channel flow. Application in two-dimensional estuary modeling is simply an extension of suspended solid transport with inclusion of regions of resuspension or deposition based on these relationships.
FIGURE 4.2: Particle Tracking in Tampa Bay - 0 Days
FIGURE 4.3: Particle Tracking in Tampa Bay - 28 Days
Figure 21. Particle positions after 42 days

FIGURE 4.4: Particle Tracking in Tampa Bay - 42 Days
FIGURE 4.5: Particle Tracking in Tampa Bay - 56 Days
FIGURE 4.6: Particle Tracking in Tampa Bay - 70 Days
4.2 - Bouyant Particle (Oilspill) Modeling

The transport of bouyant particles is somewhat more complicated than that of suspended particles. Bouyant particles, where not suspended or dispersed through the water column (as in the case of excessively turbulent regions), are primarily transported by surface currents and/or by dispersive "spreading" from sources (oilspills). While suspended particles are primarily transported by the local net velocity, bouyant particles are not strictly transported by these net velocities. Tidal velocities at the surface can be much larger than the net velocities. They can also be in different directions. In the absence of wind influences and because of the uniform flow typically associated with tidal transport in open areas, this condition is rare or insignificant.

Wind blowing across the water's surface can have a much greater influence on the transport of floating particles or debris than normal tidal transport. If the particle is of sufficiently light density to float "high" on the water's surface, wind forces acting on a exposed particle surface area can have a "sail" effect on its transport. Additionally, wind shear forces across the water's surface create a surface water transport that is approximately 3% of the wind velocity.
The USF two-dimensional thermal bay model was developed in 1977 to be used for modeling thermal plumes from water cooled power plants [5][6]. The mathematical formulation of the model is based on the conservation of energy (energy budget) and the calculation of heat flux. The energy budget and the calculation of heat flux terms follow.

5.1 - The Energy Budget

The most important natural mechanisms which influence the energy budget of a bay include:

1. Short-wave solar radiation \( H_s \),
2. Long-wave atmospheric radiation \( H_a \)
3. Reflected solar and atmospheric radiation \( H_{sr} \) and \( H_{ar} \)
4. Long-wave back radiation \( H_{br} \)
5. Evaporation \( H_e \)
6. Conduction \( H_c \)

Other factors included in the summation of heat flux terms \( \Sigma H_m \) are:

1. Advection by the evaporating water
2. Advection by streamflow
3. Bottom transfer
4. Precipitation
5. Decomposition of organics
6. Dissipation of friction heat

The heat exchange of a homogeneous volume of water can be expressed as the summation of all the heat flux terms which affect the heat content of the water. That is,

\[ \Delta = (H_s + H_a - H_{sr} - H_{ar}) - (H_{br} + H_e + H_c) + \Delta H_m \]  

(5.1)

The first set of terms on the right-hand side of the above equation is called the absorbed radiation term. It is independent of water temperature. The terms in the second set are functions of the water temperature. The last term is the summation of all the remaining heat flux terms. Both the first and second set of terms are considered to be surface phenomena, whereas the last term includes heat mechanisms which may exist at various locations within the water volume.

**Short-Wave Solar Radiation, \( H_s \)**

Short-wave solar radiation is the radiant energy which passes directly from the Sun to the Earth. The amount of short-wave radiation reaching the Earth's atmosphere can be determined from the solar constant, which is the rate at which energy is received at the top of the atmosphere upon a unit surface normal to the Sun's direction in free space at the Earth's mean distance from the sun. Measured values of the solar constant range from 1.89 to 2.05 calories per square centimeter per minute (cal/sq-cm/min.). The median of the measured values is approximately 1.93 cal/sq-cm/min, which is equivalent to 10,200 BTU/sq-ft/day. The amount of short-wave radiation reaching the Earth's surface depends upon the amount of absorption,
scattering and diffusive reflection which occurs in the atmosphere as a result of the presence of ozone, water vapor, particulate matter and dry air. The clear sky short-wave radiation received at the Earth's surface after losses due to scattering and absorption in the atmosphere may be expressed empirically as:

\[ H_{cs} = h_o \left[ a_1 + 0.5 (1 - a_2 - d) \right] \text{cal/sq-cm/day} \]

where:
- \( H_{cs} \) = clear sky short-wave radiation received at the Earth's surface
- \( h_o \) = extraterrestrial radiation flux
- \( a_1, a_2 \) = atmospheric transmission coefficients
- \( d \) = total dust attenuation

The short-wave radiation flux on the outer atmosphere during an interval of time is given by:

\[ h_o = \frac{I_0}{r} \left\{ \sin \phi \sin \delta (t_e - t_b) + \frac{12}{\pi} \cos \phi \cos \delta \left[ \sin \frac{\pi t_e}{12} - \sin \frac{\pi t_b}{12} \right] \right\} F_r \]

where:
- \( I_0 \) = solar constant, 1.93 cal/sq-cm/min
- \( r \) = ratio of the actual and mean distance between the Sun and the Earth
- \( \phi \) = latitude of the site
- \( \delta \) = declination of the Sun
- \( t_b, t_e \) = hour angles corresponding to the beginning and the end, respectively, of any time interval between sunrise and sunset.
The relative Earth-Sun distance, \( r \), can be calculated by the following equation:

\[
r = 1 + 0.017 \cos \left( \frac{\pi}{180} (181 - D) \right)
\]

where:

\( D \) = the number of the day of the year

The declination, \( \delta \), can be calculated by the following equation:

\[
\delta = \frac{23.45}{180} \cos \left( \frac{2\pi}{365} (173 - D) \right)
\]

The hour angles, \( t_b \) and \( t_e \), can be calculated by the following equation:

\[
\begin{align*}
    t_b &= ST_b - \Delta t_s - 12 + ET \\
    t_e &= ST_e - \Delta t_s - 12 + ET
\end{align*}
\]

where:

\( ST_b, ST_e \) = standard times of beginning and end, respectively, of the interval chosen for temperature analysis

\( ET \) = equation of time from a solar ephemeris

\( \Delta t_s \) = difference between standard and local civil time

The difference between standard and local time, \( \Delta t_s \), can be expressed as follows:

\[
\Delta t_s = \frac{E}{15} \left( L_{sm} - L_{lm} \right)
\]

where:

\( E = -1 \) for west longitude

\( E = +1 \) for west longitude

\( L_{sm} \) = longitude of standard meridian
Vol II

L_{lm} = longitude of local meridian

The diurnal exposure correction factor, $F_r$, is defined as follows:

$F_r = 1$ when $ST_r \leq ST_b$ or $ST_e \leq ST_s$

$F_r = 0$ when $ST_s \leq ST_b$ or $ST_e \leq ST_r$

where:

$ST_r, ST_s$ = standard times of sunrise and sunset, respectively

$ST_r = 12 - \frac{12}{\pi} \text{arccos} \left( \tan \frac{\pi \phi}{180} \right) + \Delta t_s + \Delta t_{or}$

$ST_s = 24 - ST_r + 2\Delta t_s + Wt_{or} - \Delta t_{os}$

where:

$\Delta t_{or}, \Delta t_{os}$ = hour corrections for obstruction of sunrise and sunset, respectively, at the site

For application to a lake or bay, the terms $\Delta t_{or}$ and $\Delta t_{os}$ are generally neglected.

The mean atmospheric transmission coefficient after scattering and absorption is expressed by:

$a_1 = e^{-\left(0.465 + 0.134 w \right) \left(0.179 + 0.421 e^{-0.721 m}\right) m}$

where:

$m = \text{the optical air mass}$

$m = \frac{Z}{10^{-19,100}}$

$\frac{180 \alpha_s}{\sin \alpha + 0.15 \left(\frac{\pi}{\alpha_s} + 3.885\right)^{1.153}}$
where:

\[ z = \text{altitude of the site in meters} \]

\[ a_s = \text{Sun's altitude in radians} \]

\[ w = \text{mean daily precipitable water content in the atmosphere} \]

\[ w = 0.86 \ e \]

where:

\[ \Theta_d = \text{mean daily dew-point temperature, in deg C} \]

The mean daily dew-point temperature, in degrees F, may be determined as a function of the air temperature and the relative humidity.

\[ \Theta_{dF} = T_aF - 21. + \frac{90. - T_aF (100. - 1.)}{20.} + .58 (\text{Hum} - 50.) - 8. \left( \frac{\text{Hum} - 50.}{50.} \right)^2 \]

The mean atmospheric transmission coefficient after scattering may be expressed as follows:

\[ a_2 = \ e^{- \left\{ - \left\{ 0.465 + 0.134 \ w \right\} \left\{ 0.129 + 0.171 \ e^{-0.880 \ m} \right\} m \} \}

The effect of cloud cover on short-wave radiation may be expressed by:

\[ H_s = H_{cs} \ C_s \]

where:

\[ H_{cs} = \text{clear sky short-wave radiation} \]

\[ C_s = \text{cloud cover factor} \]

\[ C_s = 1 - 0.65 \ C^2 \]

where:

\[ C = \text{a measure of the proportional (decimal) of the sky occupied by clouds} \]
$H_s$ is the magnitude of the short-wave radiation which reaches the Earth's surface following absorption, reflection, and scattering in the atmosphere, and the absorption and reflection in a cloud cover. This value is further reduced by reflection at the water surface, $H_{sr}$, before final absorption by the water mass.

**Long-Wave Atmospheric Radiation, $H_{sr}$**

Long-wave atmospheric radiation is primarily dependent on air temperature, vapor pressure and cloud cover. It may be expressed using the Stefan-Boltzmann fourth-power radiation law as follows:

$$H_a = E_a o (T_a + 273)^4 \text{ cal/sq-cm/day}$$

where:

- $E_a$ = emissivity of the atmosphere
- $o$ = Stefan-Boltzmann constant, $0.000000118$ cal/sq-cm/day/deg K
- $T_a$ = air temperature, in deg C

The emissivity of the atmosphere may be related empirically to the atmospheric vapor pressure and cloud cover, as follows:

$$E_a = a + b e_a$$

where:

- $e_a$ = atmospheric vapor pressure, in mb
  - $a = 0.74 + 0.025 \text{ C}$
  - $b = 0.0049 - 0.00005 \text{ C}$
C = cloud cover, in tenths

The vapor pressure is expressed as an empirical function of the air temperature, the wet-bulb temperature, and the atmospheric pressure, as follows:

\[
e_a = 217180000. \ e^{-\frac{4157.}{239.09 + T_{wb}}} \ - \ p_a (T_a - T_{wb}) (0.00066 + 0.000000759 T_{wb})
\]

where:

- \( T_{wb} \) = wet bulb temperature, in deg C
- \( p_a \) = atmospheric pressure, in mb

The wet-bulb temperature, in degrees F, may be determined as a function of the air temperature, in degrees F, and the relative humidity.

\[
T_{wbF} = T_{aF} - 15. + 90. - T_{aF} \left( \frac{100.}{\text{Hum}-1} \right) + .4(\text{Hum} - 50.) - 5.\left( \frac{\text{Hum} - 50.}{50.} \right)^2
\]

Reflected Solar Radiation, \( H_{sr} \)

A certain portion of the short-wave solar radiation is reflected by the water surface. The amount of reflection at a given time is a function of the incident radiation, \( H_s \), the altitude of the sun and the degree of cloudiness, and can be expressed as follows:

\[
H_{sr} = H_s R_s \text{ cal/sq-cm/day}
\]

where:

\[
R_s = A \left( \alpha_s \right)^B
\]
\( \alpha_s \) is the Sun's altitude in degrees (a function of time), and the empirical constants A and B depend on the degree of cloudiness.

Table 5.1: Reflected Solar Radiation Constants

<table>
<thead>
<tr>
<th>Cloudiness (Tenth)</th>
<th>Low Clouds</th>
<th>High Clouds</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clear (0)</td>
<td>1.18 -0.77</td>
<td>1.18 -0.77</td>
</tr>
<tr>
<td>Scattered (1-6)</td>
<td>2.17 -0.96</td>
<td>2.20 -0.98</td>
</tr>
<tr>
<td>Broken (6-9)</td>
<td>0.78 -0.68</td>
<td>1.14 -0.68</td>
</tr>
<tr>
<td>Overcast (10)</td>
<td>0.20 -0.30</td>
<td>0.50 -0.58</td>
</tr>
</tbody>
</table>

Reflected Atmospheric Radiation, \( H_{ar} \)

A portion of the long-wave atmospheric radiation received at the water surface is reflected back into the atmosphere. The amount reflected back is usually taken as a constant proportion of the incident atmospheric radiation as follows:

\[
H_{ar} = 0.03 H_{a} \text{ cal/sq-cm/day}
\]

Long-Wave Back Radiation, \( H_{br} \)

A body of water transfers energy back to the atmosphere, radiating as a black body. The back radiation, \( H_{br} \), may be expressed using the Stefan-Boltzmann fourth-power radiation law, as follows:
\[ H_{br} = E_w \sigma (T_w + 273.)^4 \text{cal/sq-cm/day} \]

where:

- \( E_w \) = emissivity of the water, 0.97
- \( \sigma \) = Stefan-Boltzmann constant,
  
  \[ \text{0.000000118 cal/sq-cm/day/deg K}^4 \]
- \( T_w \) = water surface temperature, in deg C

**Evaporation Heat Transfer, \( H_e \)**

The amount of heat lost from a body of water by the process of evaporation is directly proportional to the quantity of water evaporating, and can be expressed as follows:

\[ H_e = \rho q L \text{cal/sq-cm/day} \]

where:

- \( \rho \) = density of the water, 1. gm/cu-cm
- \( q \) = amount of water evaporated, in cm/day
- \( L \) = latent heat of vaporization, in cal/gm

The quantity of water evaporated, \( q \), may be estimated by the following empirical formula:

\[ q = 0.0045 u (e_s - e_a) \]

where:

- \( u \) = wind speed, in mph
- \( e_s \) = vapor pressure at the water surface, in mb
- \( e_a \) = atmospheric vapor pressure, in mb
The vapor pressure at the water surface, or the saturation vapor pressure, $e_s$, may be expressed empirically as a function of the water surface temperature, $T_w$, in degrees C, as follows:

$$e_s = \frac{-4157.}{239.09 + T_w}$$

The latent heat of vaporization, $L$, can be expressed as a function of the water surface temperature, in degrees C, as follows:

$$L = 595.9 - 0.545 T_w \text{ cal/gm}$$

Conductive Heat Transfer, $H_c$

The heat loss or gain across a water surface due to conduction is proportional to the evaporation heat transfer.

$$H_c = R H_e \text{ cal/sq-cm/day}$$

$R$ is the Bowen ratio which is expressed as follows:

$$R = 0.61 \frac{T_w - T_a}{e_s - e_a} \frac{P_a}{1000}$$

where:

- $T_w$ = water surface temperature, in deg C
- $T_a$ = air temperature, in deg C
- $e_s$ = saturation vapor pressure, in mb
- $e_a$ = atmospheric vapor pressure, in mb
- $P_a$ = atmospheric pressure, in mb
The equilibrium temperature, \( T_e \), of the water is that temperature for which the heat gained due to solar and atmospheric radiation exactly balances the heat due to back radiation, evaporation, and conduction. The net heat flux, \( \Delta H \), is zero. Therefore,

\[
H_s + H_a - H_{sr} - H_{ar} - H_{br} + H_e = H_c
\]

The net rate of heat transfer at the water surface is proportional to the product of the water surface area and the difference between the surface temperature and the equilibrium temperature.

Rate of heat dissipation = \(-\rho c_p k_h A (T_S - T_E)\) BTU/day

where:

- \( \rho \) - density of water, 1.94 slugs/cu ft
- \( c_p \) = specific heat of water, 32.2 BTUslug/deg F
- \( k_h \) = kinematic surface heat exchange coefficient, in ft/day
- \( A \) = water surface area, in sq ft
- \( T_S \) = water surface temperature, in deg F
- \( T_E \) = equilibrium temperature, in deg F

The surface heat exchange coefficient, \( K \), is defined as

\[
K = \rho c_p k_h \text{ BTU/sq ft/day/deg F}
\]
TEMPERATURE DISTRIBUTION

The time-dependent mass transfer equation augmented by the summation of heat flow mechanisms which affect the heat content in a volume of water may be applied for a shallow, vertically well-mixed system. The vertically-integrated equation of temperature distribution is:

$$\frac{\partial (DT)}{\partial t} + \frac{\partial (UT)}{\partial x} + \frac{\partial (VT)}{\partial y} - \frac{\partial}{\partial x} (DE_x \frac{\partial T}{\partial x}) - \frac{\partial}{\partial y} (DE_y \frac{\partial T}{\partial y}) - \frac{\Delta H}{\rho c_p} = q_s T_s$$

where:

- $x, y$ = horizontal Cartesian coordinates
- $t$ = time
- $T$ = vertically-averaged water temperature
- $U, V$ = transports in the $x$- and $y$-directions, respectively
- $D$ = water depth
- $E_x, E_y$ = $x$- and $y$-direction dispersion coefficients, respectively
- $\Delta H$ = summation of all heat flux terms
- $\rho$ = density of the fluid
- $c_p$ = specific heat of the fluid
- $q_s$ = specific discharge (discharge per unit surface area) of a given temperature source or sink $s$
- $T_s$ = temperature of the given source or sink $s$

When the temperature model is applied to a local discharge, the fine spatial and temporal resolutions dictate that the values of the dispersion coefficients be approximately the turbulent eddy conductivities. Large-scale models require an inflated dispersion coefficient to represent the effects of...
tidal mixing and small-scale convective processes. The dispersion coefficient at all land boundaries and structures must be set to zero.

The mass transfer heat equation can be approximated in an explicit finite-difference form, using upstream differences for the advective terms and centered spatial differences for the dispersive terms, advancing in time. The empirical accuracy criterion associated with the finite-difference form of the equation is as follows:

$$\Delta t \leq \frac{8E_{\min}}{|U_{\max}|}$$

where:

- $\Delta t$ = square cell size of the model ($\Delta t = \Delta x = \Delta y$)
- $E_{\min}$ = minimum value of the dispersion coefficient ($E_x$ or $E_y$)
- $|U_{\max}|$ = absolute value of the maximum velocity component ($U$ or $V$)

The cell size of the model must also be small enough to adequately represent the modeled area.

The stability criteria which determine the time step ($\Delta t$) which must be used are:

$$\Delta t \leq \frac{\Delta s}{|U_{\max}|}$$

or

$$\Delta t \leq \frac{\Delta s^2}{4E_{max}} \quad \text{or} \quad \Delta t \leq \frac{\Delta x^2 \Delta y^2}{2(E_x\Delta y^2 + E_y\Delta x^2)} \quad \text{for} \ \Delta x, \Delta y$$

where:

- $E_{max}$ = maximum value of the dispersion coefficient ($E_x$ or $E_y$) used

The temperature, $T$, in the temperature distribution equation is the local temporal water temperature. The most useful application of a
temperature distribution model, however, is for the investigation of the effects of heat sources such as power plant cooling water outfalls on the thermal environment of surrounding waters. For this type of application, the excess temperature distribution of the water due to the heat source is used.

$$\theta = T - t'$$

where:

- $\theta$ = excess water temperature due to a heat source (power plant)
- $T$ = perturbated water temperature
- $t'$ = naturally existing water temperature

$$\frac{\partial (\theta \delta)}{\partial t} + \frac{\partial (\theta U)}{\partial x} + \frac{\partial (\theta V)}{\partial y} - \frac{\partial}{\partial x} (\delta \frac{\partial \theta}{\partial x}) - \frac{\partial}{\partial y} (\delta \frac{\partial \theta}{\partial y}) = - K \theta + q \theta_s$$

where:

- $K$ = surface heat exchange coefficient
- $\theta_s$ = excess temperature of the given source or sink $s$

$$\frac{\Delta H}{\rho c_p} = K(T - T_e)$$

$T_e$ = equilibrium temperature (a function of temperature, relative humidity and atmospheric conditions)
5.2 - Input Variable Definition - Thermal Model

**HEAT FLUX**

Location information:

- **ELAT** - Latitude of the site (degrees)
- **ELAM** - Longitude of the site (degrees)
- **ELSM** - Longitude of the standard meridian (degrees)
- **EEE** - Time correction factor
  - If EEE = -1 West longitude
  - If EEE = 1 East longitude
- **ZZZ** - Altitude of the site (feet)

Time information:

- **NNN** - Number of days for the program run
- **NYEAR** - Year in which the program run starts
- **MONTH** - Month in which the program run starts
- **NDAY** - Day of the month in which the program run starts
- **NHOUR** - Hour of the day in which the program run starts

Initial temperature information:

- **TEMWFI** - Water temperature at the beginning of the program run

Program control codes:

- **IDYM** - Code to indicate whether the calculations will be for a dynamic or a steady-state system
  - If IDYM = 0 Steady-state
  - If IDYM = 1 Dynamic
KREAD - Number of hours between meteorological input data

KDEP - Number of timesteps between hydraulic input data

KMEAS - Code to indicate whether measured water temperature data will be compared against computed results

If KMEAS = 0 No measured data is available

If KMEAS = 1 Measured data is available

KPLOT - Code to indicate the desired plotting mode

If KPLOT = 0 No plotting is done

If KPLOT = 1 The calculated heat flux terms and resultant water temperatures are plotted as functions of time

If KPLOT = 2 The meteorological input data, calculated heat flux terms and resultant water temperatures are plotted as functions of time

Meteorological information (for every KREADth hour of the program run):

TEMPAF - Air temperature (degrees F)

HUM - Relative humidity (percent)

WNDVEL - Wind velocity (MPH)

CCC - Cloud cover (tenths)

LEVC - Level of the clouds

If LEVC = 1 The clouds are low (< 10,000 ft)

If LEVC = 2 The clouds are high (> 10,000 ft)

PERSA - Atmospheric pressure (mb)

DA - Total dust attenuation

Hydraulic information (for every KDEPth timestep during the program run:)

DEPTH - Water depth at the site (ft)

Measured water temperature information (needed only if KMEAS = 1):

NMEAS - Total number of water temperature measurements
(For L = 1 to NMEAS):

- TIMEAS(L) - Time (in hours since the beginning of the program run) at which measurement L was taken
- TIMEAS(L) - Water temperature when measurement L was taken (degrees F)

**TEMPERATURE DISTRIBUTION**

Grid layout information:

- M - Maximum number of grid increments in the x-direction
- N - Maximum number of grid increments in the y-direction
- DX - Grid increment in the x-direction (ft)
- DY - Grid increment in the y-direction (ft)

Program control information:

- KDAYSP - Total number of days to be simulated
- TIMECT - Temperature distribution model timestep (min)
- KOTHER - Number of timesteps between thermal data inputs
- KDYPRT - Number of days between printouts
- NSEW - Number of sources
- NPP - Number of power plants

Thermal information (for every KOTHERth timestep):

- TEQF - Equilibrium temperature (degrees F)
- SHEK - Surface heat exchange coefficient (BTU/sq-ft/day/F)

Source information (for each source, L = 1 to NSEW):

- IS(L), JS(L) - Grid location of the source (i,j)
- QS(L) - Discharge rate of the source (MGD)
- GST(L) - Temperature of the source water (degrees F)

- (QT(L,K),K=1,5 - Name of the source)
Power plant information (for each power plant, \( L = 1 \) to \( NPP \)):

- \( IPI(L), JPI(L) \) - Grid location of the power plant intake \((i,j)\)
- \( IPO(L), JPO(L) \) - Grid location of the power plant outfall \((i,j)\)
- \( QPP(L) \) - Discharge rate of the power plant (MGD)
- \( QOT(L) \) - Temperature of the power plant outfall water (degrees F)
- \( QTP(L,K), K=1,5 \) - Name of the power plant

Boundary conditions:

Exchange coefficients at the mouth of the bay.
5.3 - Output - Thermal Bay Model

**HEAT FLUX**

Lists of the calculated heat flux terms, water temperatures and surface heat exchange coefficients as functions of time;

Graphs comparing the computed water temperature as a function of time with measured water temperatures;

Graphs showing the meteorological input data as functions of time;

Graphs showing the calculated heat flux terms and resultant water temperatures as functions of time.

**TEMPERATURE DISTRIBUTION**

Printouts of the temperature distributions (in map form):

$T(I,J)$ - Temperature at grid location $I,J$ (degrees F)

Mass balance information for the temperature at the time of each printout:

- Total energy, in BTU, currently in the grid system;
- Change in the total energy in the system since the last printout;
- Total energy, in BTU's, which radiated from the grid system since the start of the program run;
- Energy radiated from the system since the last printout;
- Total energy, in BTU's, which was transported out of the grid system since the start of the program run;
- Energy which was transported out of the system since the last printout;
- Total energy, in BTU's, which entered the grid system from sources since the start of the program run;
- Energy which entered the system from sources since the last printout;
- Plotted contour maps of the distribution of temperature in the grid system at any given time;
- Plotted shade maps of the distribution of temperature in the grid system at any given time.
Chapter 6

MASS TRANSPORT MODELS

The calculation of mass transport in numerical simulation of fluid flow has been carried out by many investigators. A discussion of many successful techniques and the governing equations from which they are derived may be found in Tracor [7].

For well-mixed, shallow water-bodies (i.e., estuaries with minimum stratification) a two-dimensional solution of the equations of fluid motion is sufficient.

The 2-D water quality and mass transport programs developed at The Center for Mathematical Modeling in the College of Engineering, University of South Florida, are based on the vertically integrated equations of two-dimensional mass transport. The basic equation for 2-D mass transport used in the USF models is derived in Appendix C. In simple form it can be stated as:

\[
\frac{\partial (CD)}{\partial t} = - \frac{\partial (UC)}{\partial x} - \frac{\partial (VC)}{\partial y} + Q - KCD
\]

(6.1)

where,

C = the vertically averaged concentration of constituent (usually mg/l or ppt)

D = water depth in feet

U,V = transports in the x and y coordinate directions, respectively (CFS/ft. width)

Q = source or sink flows (CFS/ft²)
K = the first-order decay rate of constituent (1/sec)

This equation becomes more complex as the interactions and dependences on additional parameters or other modeled constituents are included. It is also different for each constituent.

Simple source and sink terms may be additions of material mass from inflows (such as sewage) or any constant withdrawal that is not dependent on the concentrations of substance present. Constituent decay is usually caused by interaction with biological processes. It is usually considered to be first-order, where the amount of decay is proportional to the amount present. However, it is also provided that constituent decay can be caused by more than one source. This is the case in the ecological model (see Section 6.2). In fertile water bodies biological processes are so complex that there is always a sacrifice of detail in the name of economy.

6.1 - Salinity Model

For conservative substances such as salinity concentrations a simplified form of the vertically integrated 2-D mass transport model is used. The decay term, -KCD, is not needed and is omitted. The source term, Q, includes both inflows of low saline fresh water dilution and/or possible redissolution of salts from reservoir sources. The equation of mass transport of dissolved salts is:

\[
\frac{\partial (SD)}{\partial t} = \frac{\partial (US)}{\partial x} + \frac{\partial (VS)}{\partial y} + Q_s 
\]  

(6.1a)
where,

\[ S = \text{salinity concentration (mg/l or ppt)} \]
\[ Q_s = \text{source/sink flows (CFS/ft}^2\) \]

It is important to note that for salinity modeling as with all conservative substances, the open water boundary interface must be delineated far enough out to be out of the noticeable transient region. That is, salinity concentrations at the open water boundary must be constant through time. If this is not possible, an alternate approach is to define an "exchange coefficient" at the boundary. This is because in the tidally driven case, much of the water that passes the boundary in the first stages of the ebb flow does not return, while most of the water passing the boundary in the latter stages of this flow does return in the first stages of flood flow. Early application of the WQ model included an effort to derive a constant exchange coefficient for Tampa Bay. With the bay marked with a conservative substance and the receiving water initially with zero concentrations, rates of exchange of 52% in one tidal cycle were observed. However, with no bay sources, and the local receiving water gaining concentrations as the two mix, this number rapidly decreased. After years of simulations with both the one mile and the 1/2 mile Tampa Bay Models, a more realistic exchange rate of 20% (Gulf water return) per tidal cycle has been found to be the standard. Constant exchange rates are acceptable except when simulations of salinity concentrations in the immediate vicinity of the boundary are required. In this case, it is usually more appropriate to define the concentrations in a time-varying sense. In the time-varying case, there exists a continuous gradient of water of salinity concentration between the concentration of the water at the boundary at ebb tide up to that of the concentration of the ultimate receiving water (Gulf water).
6.2 - Ecological Bay Model

Modeling of biological constituents such as phytoplankton and bacteria and dissolved gasses or nutrients effected by biological processes is far more difficult than that of just conservative substances. Figure 6.1 illustrates the complex processes associated with modeling productive estuary waters. The individual constituent transport and decay equations are listed in the following categories.

Water quality processes include spacial, temporal, thermal, and photic dependencies, to name just a few. Constituent concentrations and processes may be dependent on the specific decay relative concentrations of many other constituents or input parameters, or they can be limited by only one.

ALGAE (A)

The vertically-integrated mass transfer equation for algae biomass is:

\[
\frac{\partial (AD)}{\partial t} = - \frac{\partial (UA)}{\partial x} - \frac{\partial (VA)}{\partial y} + \frac{\partial}{\partial x} \left( E \frac{\partial (AD)}{\partial x} \right) + \frac{\partial}{\partial y} \left( E \frac{\partial (AD)}{\partial y} \right)
+ \mu AD - \rho AD - \sigma_A A - \rho_{p AD}
\]  

(6.2)

where:

- \( x, y \) = horizontal Cartesian coordinates (ft)
- \( U, V \) = transports in the \( x- \) and \( y- \)directions, respectively (cfs/ft width)
- \( A \) = vertically-averaged concentration of algae biomass (mg/l)
- \( D \) = total water depth (ft)
FIGURE 6.1: Processes Associated with Modeling Estuary Waters
Vol II 37

\[ t = \text{time (sec)} \]

\[ E_x, E_y = \text{dispersion coefficients in the x- and y-directions, respectively (ft}^2/\text{sec}) \]

\[ \mu = \text{specific growth rate of algae (1/sec)} \]

\[ \rho = \text{respiration rate of algae, temperature-dependent (1/sec)} \]

\( \text{usually} \ \rho = 0.1\mu' \)

\[ \sigma_A = \text{settling rate of algae (ft/sec)} \]

\[ \rho_G = \text{grazing rate for algae, temperature-dependent (1/sec)} \]

and,

\[ \mu = \mu' \frac{N_2 + N_3}{N_2 + N_3 + K_N} \frac{P_2}{P_2 + K_P} \lambda D e \left\{ -\left( \frac{I_A}{I_s} \right) e^{-\lambda D} - \frac{I_A}{I_s} \right\} \]  

(6.2a)

where:

\[ \mu' = \text{maximum specific growth rate of algae, temperature-dependent (1/sec)} \]

\[ N_2 = \text{concentration of ammonia as N (mg/l)} \]

\[ N_3 = \text{concentration of nitrate as N (mg/l)} \]

\[ K_N = \text{nitrogen half-saturation constant for algae growth (mg/l)} \]

\[ P_2 = \text{concentration of orthophosphate as P (mg/l)} \]

\[ K_P = \text{phosphorus half-saturation constant for algae growth (mg/l)} \]

\[ \lambda = \text{light extinction coefficient (1/ft)} \]

\[ \lambda = \frac{1.9}{H_s'} + 0.052 A \]

where:

\[ H_s' = \text{algae-free secchi depth (ft)} \]
The vertically-integrated mass transport equation for ultimate carbonaceous biochemical oxygen demand is:

$$\frac{\partial (CD)}{\partial t} = - \frac{\partial (UC)}{\partial x} - \frac{\partial (VC)}{\partial y} + \frac{3}{\partial x} (E \frac{\partial (CD)}{\partial x}) + \frac{3}{\partial y} (E \frac{\partial (CD)}{\partial y})$$

$$+ Q_C - \beta C_{CD} - \sigma C$$  \hspace{1cm} (6.3)

where:
- \(C\) = vertically-averaged concentration of ultimate carbonaceous biochemical oxygen demand (BOD\(_C\)) (mg/l)
- \(Q_C\) = source rate of BOD\(_C\) from point and nonpoint sources and sinks (mg/l-ft/sec)
- \(S\)
- \(Q_C = \sum_{s=1}^{S} q_s C_s\)
- \(q_s\) = specific discharge of source or sink \(s\) (ft/sec)
- \(C_s\) = concentration of BOD\(_C\) in source or sink \(s\) (mg/l)
- \(S\) = total number of sources and/or sinks
- \(f_1\) = low-dissolved oxygen correction factor for the decay of BOD\(_C\)
- \(f_1 = \frac{0.01}{0.01 + 0.01}\)
- \(\beta C\) = first-order decay rate of BOD\(_C\) due to oxidation, temperature-dependent (1/sec)
- \(\sigma C\) = settling rate of BOD\(_C\) (ft/sec)
ORGANIC PHOSPHORUS AS P

The vertically-integrated mass transport equation for organic phosphorus is:

\[
\frac{\partial(P_1D)}{\partial t} = -\frac{\partial(U P_1)}{\partial x} - \frac{\partial(V P_1)}{\partial y} + \frac{\partial}{\partial x}(E_x \frac{\partial(P_1D)}{\partial x}) + \frac{\partial}{\partial y}(E_y \frac{\partial(P_1D)}{\partial y})
+ Q_{P_1} + \alpha_p p AD - \beta P_1 P_1 D - \sigma P_1 P_1
\]

(6.4)

where:

- \( P_1 \) = vertically-averaged concentration of organic phosphorus as phosphorus (mg/l)
- \( Q_{P_1} \) = source rate of \( P_2 \) from point and nonpoint sources and sinks (mg/l-ft/sec)
- \( \sigma P_1 \) = settling rate of organic phosphorus (ft/sec)
- \( P_1 \) = concentration of \( P_1 \) in source of sink \( s \) (mg/l)
ORTHOPHOSPHATE AS P (PO_4-P)

The vertically-integrated mass transport equation for orthophosphate is:

\[
\frac{\partial (P_2 D)}{\partial t} = -\frac{\partial (UP_2)}{\partial x} - \frac{\partial (VP_2)}{\partial y} + \frac{\partial}{\partial x} \left( E_x \frac{\partial P_2}{\partial x} \right) + \frac{\partial}{\partial y} \left( E_y \frac{\partial P_2}{\partial y} \right) + Q_{P_2} - \alpha_P \mu A D + \beta_{P_1} P_1 D + \eta_{P_2}
\]

(6.5)

where:

\( P_2 \) = vertically-averaged concentration of orthophosphate as phosphorus

\((\text{PO}_4 - \text{P}) \) (mg/l)

\( Q_{P_2} \) = source rate of \( P_2 \) from point and nonpoint sources and sinks

\((\text{mg/l-ft/sec})\)

\[
Q_{P_2} = \sum_{s=1}^{S} q_s P_{2s}
\]

\( P_{2s} \) = concentration of \( P_2 \) in source or sink \( s \) (mg/l)

\( \alpha_P \) = fraction of algae biomass which is P (mg P/mg A)

\( \eta_{P_2} \) = benthic source rate for \( P_2 \), variable with location (ft/sec)
ORGANIC NITROGEN AS N

The vertically-integrated mass transport equation for organic nitrogen is:

\[
\frac{\partial (N_1 D)}{\partial t} = -\frac{\partial (UN_1)}{\partial x} - \frac{\partial (VN_1)}{\partial y} + \frac{\partial}{\partial x} (E x \frac{\partial}{\partial x}) + \frac{\partial}{\partial y} (E y \frac{\partial}{\partial y}) + Q_{N_1} + \alpha_N \rho AD - \beta_{N_1} N_1 D - \sigma_N N_1 
\]

where:

- \( N_1 \) = vertically-averaged concentration of organic nitrogen as nitrogen (mg/l)
- \( Q_{N_1} \) = source rate of organic nitrogen from point and nonpoint sources and sinks (mg/l-ft/sec)
- \( N_s \) = concentration of organic nitrogen in source or sink \( s \)
- \( \alpha_N \) = fraction of algae biomass which is N (mg N/mg A)
- \( \beta_{N_1} \) = first-order decay rate of organic nitrogen to ammonia, temperature-dependent (1/sec)
- \( \sigma_{N_1} \) = settling rate for organic nitrogen (ft/sec)
AMMONIA AS $N$ ($NH_3$)

The vertically-integrated mass transfer equation for ammonia is:

$$\frac{\partial (N_2 D)}{\partial t} = - \frac{\partial (UN_2)}{\partial x} - \frac{\partial (VN_2)}{\partial y} + \frac{\partial}{\partial x} (E_N \frac{\partial (N_2 D)}{\partial x}) + \frac{\partial}{\partial y} (E_N \frac{\partial (N_2 D)}{\partial y}) + Q_{N_2}$$

$$+ \beta_{N_2} N_2 D - f_2 \beta_{N_2} N_2 D + \eta_{N_2} - \Delta N_2 \mu A D - \sigma_{N_2} N_2$$  \hspace{1cm} (6.7)

where:

$N_2$ = vertically-averaged concentration of ammonia as $N$ ($NH_3-N$) (mg/l)

$Q_{N_2}$ = source rate of ammonia from point and nonpoint sources and sinks (mg/l-ft/sec)

$$Q_{N_2} = \sum_{s=1}^{S} q_s N_{2s}$$

$N_{2s}$ = concentration of ammonia in source or sink $s$ (mg/l)

$f_2$ = low-dissolved oxygen correction factor for decay of ammonia $[15]$

$$f_2 = \frac{0}{0 + 0.50}$$

$\beta_{N_2}$ = rate of biological oxidation of ammonia to nitrate, temperature-dependent (1/sec)

$\eta_{N_2}$ = benthic source rate for $N_2$, variable with location (mg/l N-ft/sec)

$\sigma_{N_2}$ = settling rate for ammonia (ft/sec)

$\Delta$ = ammonia preferential consumption factor

$$\Delta = \frac{W_1 N_2}{W_1 N_2 + W_3 N_3}$$
where:

$$W_1 + W_2 = 1$$
$$W_1 = 0.95$$
$$W_2 = 0.05$$

**NITRATE AS N (NO₃⁻)**

The vertically-integrated mass transfer equation for nitrate is:

$$\frac{\partial (VN_3)}{\partial t} - \frac{\partial (UN_3)}{\partial x} - \frac{\partial (VN_3)}{\partial y} + \frac{\partial}{\partial x} \left( E_x \frac{\partial N_3}{\partial x} \right) + \frac{\partial}{\partial y} \left( E_y \frac{\partial N_3}{\partial y} \right)$$

$$+ Q_{N_3} + f_2 \beta N_2 D + \eta_{N_3} = (1-\Delta)\sigma_{N} \mu_{AD}$$

(6.8)

where:

$$N_3 = \text{vertically-averaged concentration of nitrate as N (NO₃⁻-N) (mg/l)}$$

$$Q_{N_3} = \text{source rate of nitrate from point to nonpoint sources and sinks (mg/l-ft/sec)}$$

$$Q_{N_3} = \sum_{s=1}^{S} q_s N_{3s}$$

$$N_{3s} = \text{concentration of nitrate in source or sink s (mg/l)}$$

$$\eta_{N_3} = \text{benthic source rate of nitrate, variable with location (mg/l N - ft/sec)}$$
Dissolved Oxygen (O)

The vertically-integrated mass transfer equation for oxygen is:

\[
\frac{\partial (OD)}{\partial t} = - \frac{\partial (UO)}{\partial x} - \frac{\partial (VO)}{\partial y} + \frac{\partial}{\partial x} \left( E \frac{\partial (OD)}{\partial x} \right) + \frac{\partial}{\partial y} \left( F \frac{\partial (OD)}{\partial y} \right) \\
+ \beta_0 (O_s - O) D + \alpha_{OG} \mu A D - \alpha_{OR} \rho A D \\
- f_1 \beta_c C D - f_2 \alpha_{ON} \beta N_2 D - \eta_0
\]

(6.9)

where:

- \( O \) = vertically-averaged concentration of oxygen (mg/l)
- \( \beta_0 \) = reaeration rate of oxygen, temperature-dependent, velocity and wind-dependent (1/sec)

\[
\beta_0 = \frac{11.6101}{86400} \frac{U^{0.969} + \beta_{OW} W}{D^{1.673}}
\]

(6.9a)

where:

- \( U \) = water velocity (ft/sec)
- \( W \) = wind velocity (ft/sec)
- \( \beta_{OW} \) = wind coefficient for reaeration
- \( O_s \) = saturation oxygen concentration, temperature- and chloride-dependent (mg/l)
- \( \alpha_{OG} \) = oxygen production per unit of algae growth (photosynthesis)
  
  \( (mg O/mg A) \)
- \( \alpha_{OR} \) = oxygen uptake per unit of algae respiration (mg O/mg A)
\( n_0 \) = benthic demand for oxygen, variable with location
\( \text{ (mg/l}O - \text{ ft/day)} \)

\( \alpha_{ON_2} \) = oxygen uptake per unit of ammonia oxidation \( \text{(mg O/mg N)} \)

**OXYGEN DEFICIT \((O_D)\)**

The oxygen deficit in the water can be calculated as the difference between the saturation concentration and the dissolved oxygen:

\[ O_D = O_s - O \] \hspace{1cm} (6.10)

where:

\( O \) = concentration of dissolved oxygen \( \text{(mg/l)} \)

\( O_s \) = saturation concentration of oxygen, temperature- and salinity-dependent \( \text{(mg/l)} \) \([15]\)

and,

\[ O_s = e^{(P_1 + P_2)s} \] \hspace{1cm} (6.10a)

where:

\[ P_1 = -173.0722 - 21.8492 T_Z + \frac{249.6339}{T_Z} + 143.3483 \cdot \ln T_Z \] \hspace{1cm} (6.10b)

\[ P_2 = -0.033096 + 0.014259 T_Z \] \hspace{1cm} (6.10c)

\[ T_Z = \frac{T + 273.15}{100} \]

\( T \) = water temperature \( \text{(deg C)} \)

\( S \) = salinity \( \text{(ppt)} \)

\( O_D \) = oxygen deficit concentration \( \text{(mg/l)} \)
**CHLOROPHYLL a**

The concentration of chlorophyll $a$ in the water can be calculated as a function of the total algae biomass:

$$C_L = \alpha_{C_L} A$$  \hspace{1cm} (6.11)

where:

- $C_L$ = concentration of chlorophyll $a$ (mg/l)
- $\alpha_{C_L}$ = ratio of chlorophyll $a$ to the total algae biomass ($\mu g$ Chl $a$/mg A)

**TEMPERATURE DEPENDENCE**

Many of the rates used in the water quality distribution equations are dependent on the temperature of the water:

$$X_T = X_{T_s} \Theta$$  \hspace{1cm} (6.12)

where:

- $X_T$ = the value of the variable at the local temperature, $T$
- $X_{T_s}$ = the value of the variable at the standard temperature, $T_s$
- $\Theta$ = an empirical constant for each variable, usually taken as 1.06 for the maximum specific algae growth rate, 1.0159 for the oxygen
reaeration coefficient and 1.045 for all other temperature-dependent variables (respiration and grazing of algae; and the decay rates of \( \text{BOD}_c \), organic phosphorus, organic nitrogen, ammonia)

\[
\begin{align*}
T &= \text{local water temperature (degrees C)} \\
T_s &= \text{standard temperature, 20 degrees C}
\end{align*}
\]

6.2.1 - Input Data Variable Definition - Ecological Model

Grid layout information:

\[
\begin{align*}
M &= \text{Maximum number of grid increments in the x-direction} \\
N &= \text{Maximum number of grid increments in the y-direction} \\
DX &= \text{Grid increment in the x-direction (ft)} \\
DY &= \text{Grid increment in the y-direction (ft)}
\end{align*}
\]

Program control information:

\[
\begin{align*}
\text{KDAYSP} &= \text{Total number of days to be simulated} \\
\text{TIMECT} &= \text{Ecological model timestep (min)} \\
\text{NSEW} &= \text{Number of sources} \\
\text{NPP} &= \text{Number of power plants}
\end{align*}
\]

Rates and constants:

\[
\begin{align*}
\text{GROMAX} &= \text{Maximum specific growth rate of algae at 20\degree C (1/day)} \\
\text{RESP} &= \text{Respiration rate of algae at 20\degree C (1/day)} \\
\text{GRAZE} &= \text{Grazing rate for algae at 20\degree C (1/day)} \\
\text{HSATP} &= \text{Phosphorus half-saturation constant for algae growth (mg/1 of P)}
\end{align*}
\]
HSATN - Nitrogen half-saturation constant for algae growth (mg/l of N)
PHOTO - Photo period -- percent of the day which is sunlit (hundredths)
EXTL - Light extinction coefficient (1/ft)
XLIGHT - Average light intensity at the water surface (Langley/day)
XSATLI - Saturation light intensity (Langley/day)
CHL - Ratio of chlorophyll a to the algae biomass (µg Chl a/mg A)
DKC - Decay rate of carbonaceous BOD due to oxidation at 20° C (1/day)
DKP - Decay rate of organic phosphorus to orthophosphate due to hydrolysis at 20° C (1/day)
DKNH3 - Decay rate of ammonia to nitrate due to biological oxidation at 20° C (1/day)
SETA - Settling rate of algae (ft/day)
SETC - Settling rate of carbonaceous BOD (ft/day)
SETP - Settling rate of organic phosphorus (ft/day)
SETN - Settling rate of organic nitrogen (ft/day)
SETNH3 - Settling rate of ammonia (ft/day)
BENPO4 - Benthic source rate for orthophosphate (mg P/m² -day)
BENNHH3 - Benthic source rate for ammonia (mg N/m² -day)
BENNO3 - Benthic source rate for nitrate (mg N/m² -day)
FP - Fraction of the algae biomass which is phosphorus (mg P/mg A)
PRF - Preference rate of ammonia over nitrate for use in the growth of algae (hundredths of a percent)
OXGROW - Oxygen production per unit of algae growth (mg O/mg A)
OXRESP - Oxygen uptake per unit of algae respiration (mg O/mg A)
OXNH3 - Oxygen uptake per unit of ammonia oxidation (mg O/mg N)
BENOX - Benthic demand for oxygen (mg O₂/m²·day)

TCOGRO - Temperature correction coefficient for the maximum specific growth rate of algae

TCORSP - Temperature correction coefficient for the respiration rate of algae

TCOGRZ - Temperature correction coefficient for the grazing rate of algae

TCOC - Temperature correction coefficient for the decay rate of carbonaceous BOD

TCOP - Temperature correction coefficient for the decay rate of organic phosphorus

TCON - Temperature correction coefficient for the decay rate of organic nitrogen

TCONH₃ - Temperature correction coefficient for the decay rate of ammonia

TCOREA - Temperature correction coefficient for the reaeration rate of dissolved oxygen

WIND - Wind velocity (mph)

WINDCO - Wind coefficient for reaeration

TEMP - Water temperature (deg C)

SALIN - Water salinity (ppt)

Source information (for each source, L = 1 to NSEW):

IS(L), JS(L) - Grid location of the source (i,j)

GS(L) - Discharge rate of the source (MGD)

NTREAT(L) - Sewage treatment level

If NTREAT(L) = 1, Primary treatment
If NTREAT(L) = 2, Marginal Secondary treatment
If NTREAT(L) = 3, Secondary treatment with Nitrification
If NTREAT(L) = 4, High Rate Biological treatment
If NTREAT(L) = 5, Advanced treatment
If NTREAT(L) = 6, Ultimate treatment
If NTREAT(L) = 0, Other treatment

The following 6 parameters need to be input only if NTREAT(L) = 0:

QSC(L) - Source concentration of ultimate carbonaceous BOD (mg/l of BODC)
QSP(L) - Source concentration of phosphorus (mg/l of P)
QSN(L) - Source concentration of organic nitrogen as N (mg/l of Org N-N)
QSNH3(L) - Source concentration of ammonia as N (mg/l of NH3-N)
QSN02(L) - Source concentration of nitrate as N (mg/l of NO2-N)
QSN03(L) - Source concentration of nitrate as N (mg/l of NO3-N)

(QT(L,K),K=1,5) - Name of the source

Power plant information (for each power plant, L = 1 to NPP):

IPI(L), JPI(L) - Grid location of the power plant intake (i,j)
IPO(L), JPO(L) - Grid location of the power plant outfall (i,j)
QPP(L) - Discharge rate of the power plant (MGD)

(QTP(L,K),K=1,5) - Name of the power plant
Open-water boundary conditions:

Exchange coefficients at the open water interface must be defined.

Optionally, steady-state or time varying concentrations may be specified for each constituent.

6.2.2 - Output Forms - Ecological Model

Because water quality interests are both spatial and temporal in nature, output from the ecological water quality bay model is divided into two major categories: mapped and simplified block diagram forms.

Mapped Output:

Printouts of the concentration distributions (in map form) are provided for each of the calculated constituents. They are useful in determining the spatial distributions of specific constituent concentrations, but are instantaneous in time. While some temporal accounting is included, it is cumbersome and often requires reruns of printout. Temporal accounting is specific to the various "box models" described in the next section.

Mapped output are included for the following variables:

$TA(I,J)$ - Concentration of algae biomass at grid location $I,J$ (mg/l of A) (The concentration of chlorophyll $A$ may be calculated as: $CHLA * TA(I,J)$.)

$TC(I,J)$ - Concentration of ultimate carbonaceous BOD at grid location $I,J$ (mg/l of BOD$_C$)

$TP(I,J)$ - Concentration of phosphorus at grid location $I,J$ (mg/l of P)
Vol II 52

TN(I,J) - Concentration of organic nitrogen as N at grid location I,J (mg/l of Org N-N)

TNH3(I,J) - Concentration of ammonia as N at grid location I,J (mg/l of NH3-N)

TN02(I,J) - Concentration of nitrite as N at grid location I,J (mg/l of NO2-N)

TN03(I,J) - Concentration of nitrate as N at grid location I,J (mg/l of NO3-N)

TOXD(I,J) - Concentration of oxygen deficit at grid location I,J (mg/l of Oxygen Deficit) (The concentration of dissolved oxygen may be calculated as: SATDO - TOXD(I,J).)

An example of mapped output from a Tampa Bay simulation is not shown due to the physical size of such a document (see Vol. V).

Mass balance information for each constituent (algae, BODc, P, organic N-N, NH3-N, NO2-N and NO3-N) at the time of each printout includes:

Total weight, in pounds, of the constituent currently in the grid system;

Change in the total weight in the system since the last printout;

Total weight, in pounds, of the constituent in the grid system which decayed since the start of the program run;

Amount decayed in the system since the last printout;

Total weight, in pounds, of the constituent which was transported out of the grid system since the start of the program run;

Amount which was transported out of the system since the last printout;
Total weight, in pounds, of the constituent which entered the grid system from sources since the start of the program run;
Amount which entered the system from sources since the last printout;
Total weight, in pounds, of the constituent which was produced by growth or by the decay of other constituents in the grid system since the start of the program run;
Amount which was produced from growth or from other constituents since the last printout.

Additional mapped output include:

Plotted contour maps of the distribution of any constituent in the grid system at any given time (algae, chlorophyll a, BOD\(_c\), P, organic N-N, NH\(_3\)-N, NO\(_2\)-N, NO\(_3\)-N, oxygen deficit or dissolved oxygen);

Plotted shade maps of the distribution of any constituent in the grid system at any given time.

Block Convention:

Block diagram output includes several forms of output which are useful and specific in determining temporal distributions of constituent concentrations. This form of output gives a total accounting with time, of rates and quantities of constituent transformation, total individual storage of various forms and the amounts exchanged at the boundaries. It is very useful in determining broad effects on the equilibrium of the system such as algal blooms or other major perturbations.

Block diagram output given by the two-dimensional water quality model is not independent of spacial considerations. All information shown is tallied from the sum of the data from each \(\Delta X, \Delta Y\) model water grid. However, because
of the costs associated with spacial and temporal simulation, a simplified "Box Model" has also been developed as part of this study. The costs and facilities needed to operate this model are greatly reduced. This model is explained in great detail, along with its limitations in the fourth volume of this series.

Figure 6.2 shows an example of the simplified mass block diagram output for a typical water quality simulation. This form of output gives a total accounting of rates and quantities of constituent transformation, total individual storage of various forms and the amounts exchanged at the boundaries. This form is very useful in determining broad effects such as algal blooms or other major perturbations in the system.
FIGURE 6.2: Simplified Mass Block Diagram Output

Relationships of Water Quality Parameters in the U.S. Ecosystemal Bug Model

(heavy concentration in the body bag)
List of References


Appendix C

DERIVATION AND DISCUSSION OF THE MASS TRANSPORT EQUATION

To derive the basic transport equation consider a substance called "P" which is moved by a fluid "F," such that the weight of "P" is negligible when compared to the weight of "F". The volume of fluid "F" will be considered to be an elemental volume whose dimensions are $dx$, $dy$, and $dz$ within a Cartesian rectangular co-ordinate system.

$M_{in}$ is the total mass of substance "P" which enters the elemental volume through that side which is perpendicular to the X-axis. $M_{in}$ includes all of "P" which is transported either by convection or by molecular diffusion into the volume through that face.

$$M_{in} = \rho u c dy dz + \int_x$$
\( \rho \) = average density of the solution

\( u \) = velocity of the solution in the X-direction

\( c \) = concentration of the substance "P" in solution

\( j_x \) = rate of transport in the X-direction due to molecular diffusion

To define the term \( j_x \) more precisely, consider Fick's first law of molecular diffusion:

\[
    j_x = -\rho D_m \frac{\partial c}{\partial x} \, dydz
\]

\( D_m \) = the molecular diffusion coefficient.

The total mass of "P" leaving through the opposite face of the elemental volume may be expressed as \( M_{\text{out}} \):

\[
    M_{\text{out}} = \rho u dydz + \frac{\partial c}{\partial x} (\rho uc) \, dx dydz
    + j_x + \frac{\partial c}{\partial x} (j_x) \, dx
\]

The net change in the total mass of substance "P" due to convection and molecular diffusion in the X-direction is simply:

\[
    \Delta M_x = M_{\text{out}} - M_{\text{in}}
\]

\[
    \Delta M_x = \frac{\partial c}{\partial x} (\rho uc) \, dx dydz - \frac{\partial c}{\partial x} \{\rho D_m \frac{\partial c}{\partial x}\} \, dx dydz
\]
Equations for the change in mass in the \( Y \)- and \( Z \)-directions may be derived in a similar fashion.

\[
\Delta M_y = \frac{\partial}{\partial y} (\rho v c) \, dx \, dy \, dz - \frac{\partial}{\partial y} (\rho D m \frac{\partial c}{\partial y}) \, dx \, dy \, dz
\]

\[
\Delta M_z = \frac{\partial}{\partial z} (\rho w c) \, dx \, dy \, dz - \frac{\partial}{\partial z} (\rho D m \frac{\partial c}{\partial z}) \, dx \, dy \, dz
\]

\( v \) = velocity in the \( Y \)-direction

\( w \) = velocity in the \( Z \)-direction

The rate of flux is defined as the net change in total mass per unit time. Thus, \( \Delta M_x \), \( \Delta M_y \), and \( \Delta M_z \) may be referred to as the rates of flux in the \( X \)-, \( Y \)-, and \( Z \)-directions, respectively.

According to the law of conservation of mass, the rate of increase of mass of substance "\( P \)" within an elemental volume will be equal to the net rate of flux into the volume, plus the net rate of production of "\( P \)" within the elemental volume. Dividing through by \( dx \, dy \, dz \) yields Equation 2 where \( n_p \) is the rate of production of "\( P \)" within the elemental volume.

\[
\frac{\partial}{\partial t} (\rho c) = - \frac{\partial (\rho u c)}{\partial x} - \frac{\partial (\rho v c)}{\partial y} - \frac{\partial (\rho w c)}{\partial z} + \frac{\partial}{\partial x} (\rho D m \frac{\partial c}{\partial x}) + \frac{\partial}{\partial y} (\rho D m \frac{\partial c}{\partial y}) + \frac{\partial}{\partial z} (\rho D m \frac{\partial c}{\partial z}) + n_p
\]

The instantaneous velocity components \((u, v, \text{ and } w)\) and concentration \((c)\) are defined to be the sum of their time-averaged values and the amounts of fluctuation at that instant in time due to turbulent flow.
\[ u = \bar{u} + u' \]
\[ v = \bar{v} + v' \]
\[ w = \bar{w} + w' \]
\[ n_p = \bar{n}_p + n'_p \]
\[ c = \bar{c} + c' \]

The bar refers to the time-averaged value and the prime represents the turbulent fluctuation.

\[ \bar{u} = \frac{1}{T} \int_{t}^{t+T} u \, dt \quad (4) \]

\[ T \] = the time interval for \( u \)

Taking the time-averaged value for \( u' \):

\[ \bar{u}' = \frac{1}{T} \int_{t}^{t+T} u' \, dt = \frac{1}{T} \int_{t}^{t+T} (u - \bar{u}) \, dt \quad (5) \]

\[ \bar{u}' = \bar{u} - \bar{u} \]

\[ \bar{u}' = 0 \]

Equations similar to Equations 4 and 5 may be derived in the same manner for \( v, w, \) and \( c \). However, the time-averaged value for the product of any two
fluctuations may not necessarily be equal to zero [8] and should be kept in the equation.

Substituting the expressions of 3 into 2 and expanding yields Equation 6.

\[
\frac{\partial (\bar{c} \rho)}{\partial t} + \frac{\partial (c' \rho)}{\partial t} = - \frac{\partial}{\partial x} (\rho (\bar{u} + u') (\bar{c} + c')) - \frac{\partial}{\partial y} (\rho (\bar{v} + v') (\bar{c} + c')) - \frac{\partial}{\partial z} (\rho (\bar{w} + w') (\bar{c} + c')) \\
+ \frac{\partial}{\partial x} \left[ \rho_D \frac{\partial (\bar{c} + c')}{\partial x} \right] \\
+ \frac{\partial}{\partial y} \left[ \rho_D \frac{\partial (\bar{c} + c')}{\partial y} \right] \\
+ \frac{\partial}{\partial z} \left[ \rho_D \frac{\partial (\bar{c} + c')}{\partial z} \right] \\
+ (\bar{n}_p + n'_p)
\]

Taking the time-average of Equation 6, using Equations 4 and 5, yields Equation 7.

\[
\frac{\partial (\bar{c} \rho)}{\partial t} = - \frac{\partial}{\partial x} (\rho \bar{u} \bar{c}) - \frac{\partial}{\partial y} (\rho \bar{v} \bar{c}) - \frac{\partial}{\partial z} (\rho \bar{w} \bar{c}) \\
+ \frac{\partial}{\partial x} \left[ D_{\rho} \frac{\partial \bar{c}}{\partial x} \right] + \frac{\partial}{\partial y} \left[ D_{\rho} \frac{\partial \bar{c}}{\partial y} \right] + \frac{\partial}{\partial z} \left[ D_{\rho} \frac{\partial \bar{c}}{\partial z} \right] \\
- \frac{\partial}{\partial x} (\rho \bar{u}' \bar{c}') - \frac{\partial}{\partial y} (\rho \bar{v}' \bar{c}') - \frac{\partial}{\partial z} (\rho \bar{w}' \bar{c}')
\]
Since the products of turbulent fluctuations are not necessarily zero, those terms involving such products are assumed to be diffusive terms of convective transport. By analogy to Fick's first law, these terms are defined as follows:

\[ \rho u'c' = - \rho e_x \frac{\partial \bar{c}}{\partial x} \]  
\[ \rho v'c' = - \rho e_y \frac{\partial \bar{c}}{\partial y} \]  
\[ \rho w'c' = - \rho e_z \frac{\partial \bar{c}}{\partial z} \] 

where \( e_x, e_y, \) and \( e_z \) are turbulent dispersion coefficients [8].

Substituting Equations 8a, 8b, and 8c into Equation 7 and collecting terms yields Equation 9.

\[ \frac{\partial (\rho \bar{c})}{\partial t} = - \frac{\partial}{\partial x} (\rho \bar{u} \bar{c}) - \frac{\partial}{\partial y} (\rho \bar{v} \bar{c}) - \frac{\partial}{\partial z} (\rho \bar{w} \bar{c}) \]

\[ + \frac{\partial}{\partial x} [\rho (D_m + e_x) \frac{\partial \bar{c}}{\partial x}] + \frac{\partial}{\partial y} [\rho (D_m + e_y) \frac{\partial \bar{c}}{\partial y}] \]

\[ + \frac{\partial}{\partial z} [\rho (D_m + e_z) \frac{\partial \bar{c}}{\partial z}] + \bar{n}_p \] 

(9)
Equation 9 is the transport equation at a point. With the application of the following definitions Equation 9 becomes the transport equation for a two-dimensional flow.

\[
\begin{align*}
\bar{u} &= U + u'' \\
\bar{v} &= V + v'' \\
\bar{w} &= w'' \\
\bar{w} &= 0 \\
\bar{c} &= C + c'' \\
\bar{n} &= N_p + n''
\end{align*}
\] (10a)

All the above values without a bar or a double prime are time-averaged and vertically averaged values at a depth. The double prime refers to the vertical fluctuations from the time-averaged and vertically averaged values.

\[
U = \frac{1}{D} \int_{z_0} \bar{u} \, dz
\] (10b)

\[
D = H - z_0
\] (10c)
FIGURE C.2: Schematic of Depth Parameters

Substituting Equations 10a, b, and c into Equation 9, expanding, and integrating vertically, yields Equation 11.

\[
\begin{align*}
(1) & \int_{z_0}^{H} \frac{\partial (\rho C)}{\partial t} \, dz + \int_{z_0}^{H} \frac{\partial (\rho c'')}{\partial t} \, dz = \int_{z_0}^{H} \frac{\partial}{\partial x} \left[ \rho (D_m + e_x) \frac{\partial c}{\partial x} \right] \, dz \\
(2) & \int_{z_0}^{H} \frac{\partial}{\partial y} \left[ \rho (D_m + e_y) \frac{\partial c}{\partial y} \right] \, dz + \int_{z_0}^{H} \frac{\partial}{\partial z} \left[ \rho (D_m + e_z) \frac{\partial c}{\partial z} \right] \, dz
\end{align*}
\]
Equation 11 has been simplified in Progress Report, Volumes I and II [9]. The concentration \( C \) has been defined as a constant over depth since it is a time-averaged and vertically averaged value. Thus, \( C \) can be factored out and Equation 11 will become 12.

\[
\frac{\partial}{\partial t} (\rho C D) = \frac{\partial}{\partial x} (\rho D \frac{\partial C}{\partial x}) + \frac{\partial}{\partial y} (\rho D \frac{\partial C}{\partial y}) \\
- \frac{\partial}{\partial x} (\rho u'c') D - \frac{\partial}{\partial y} (\rho v'c'') D \\
+ C(\rho U) \frac{\partial H}{\partial x} - C(\rho U) \frac{\partial z_o}{\partial x}
\]
A double bar represents the average value over the depth. Equations 13a, b, and c are three terms from Equation 12 which have been expanded by the use of the product rule for partial differentiation.

\[
\frac{\partial}{\partial t} (\rho CD) = C \frac{\partial (\rho D)}{\partial t} + \rho D \frac{\partial C}{\partial t} \quad (13a)
\]

\[
\frac{\partial}{\partial x} (\rho UCD) = C \frac{\partial (\rho DU)}{\partial x} + \rho DU \frac{\partial C}{\partial x} \quad (13b)
\]

\[
\frac{\partial}{\partial y} (\rho VCD) = C \frac{\partial (\rho DV)}{\partial y} + \rho DV \frac{\partial C}{\partial y} \quad (13c)
\]
It is now necessary to look at the continuity equation, Equation 14.

\[
\frac{\partial p}{\partial t} + \frac{\partial}{\partial x} (\rho u) + \frac{\partial}{\partial y} (\rho v) + \frac{\partial}{\partial z} (\rho w) = 0
\]  

(14)

Taking the time-average of Equation 14 and substituting in Equation 10a, then integrating over the depth, and multiplying by \( C \) results in Equation 15. These steps are shown in Volume II, Appendix II [9].

\[
- C \frac{\partial}{\partial t} (\rho D) = C \frac{\partial}{\partial x} (\rho UD) + C \frac{\partial}{\partial y} (\rho VD) - C (\rho U) \frac{dH}{dx}
\]

\[
+ C (\rho U) \frac{\partial z_0}{\partial x} - C (\rho u'') \frac{\partial H}{\partial x}
\]

\[
+ C (\rho u'') \frac{\partial z_0}{\partial x} - C (\rho V) \frac{\partial H}{\partial y}
\]

\[
+ C (\rho V) \frac{\partial z_0}{\partial y} - C (\rho v'') \frac{\partial H}{\partial y}
\]

\[
+ C (\rho v'') \frac{\partial z_0}{\partial y}
\]

(15)

Equation 16 is obtained by performing the following steps:

1. Add equation 15 to 12.
2. Rearrange Equations 13a, b, and c and substitute them into the sum of Equations 12 and 15.
3. Divide by \( \rho \) which has already been defined as a constant.
4. Factor out \( U \) and \( V \) which have been defined as constants over depth.

The result of these steps is Equation 16.
Terms 6 through 13 of Equation 16 are shown to be zero when the following boundary conditions are applied and the terms are summed as shown in Volume II, Appendix III [9].

1. No mass transport occurs across the free surface.
2. No mass transport occurs across the bottom surface.
Since $\bar{e}^m$ is an average value over a depth, the term containing $\bar{e}^m D$ may be redefined using Fick's first law of molecular diffusion, such that:

$$\bar{e}^m_0 D = D e'_x \frac{\partial C}{\partial x}$$

(18a)

where $e'_x$ is the coefficient of longitudinal dispersion. The same analogy may be used for $\bar{e}^n$ which yields:

$$\bar{e}^n_0 D = D e'_x \frac{\partial C}{\partial y}$$

(18b)

By substituting terms 18a and b into Equation 17, dividing by depth, summing the dispersion coefficients as shown:

$$e'_x + \bar{e}_x = E_x$$

(19a)

$$e'_y + \bar{e}_y = E_y$$

(19b)

and substituting $E_x$ and $E_y$ into Equation 17, the result is the general two-dimensional mass transport equation shown as Equation 20.

$$\frac{\partial C}{\partial t} = \frac{1}{D} \left( \frac{\partial}{\partial x} \left( D \frac{\partial C}{\partial x} \right) \right) + \frac{1}{D} \left( \frac{\partial}{\partial y} \left( D \frac{\partial C}{\partial y} \right) \right) - U \frac{\partial C}{\partial x}$$

$$- \nu \frac{\partial^2 C}{\partial y^2} + \frac{M}{\rho} + \frac{N}{\rho}$$

(20)
By use of the rule for the differentiation of a product, Equation 20 may be rewritten as Equation 21.

\[
\frac{\partial C}{\partial t} = E_x \frac{\partial^2 C}{\partial x^2} + \frac{E_x}{D} \frac{\partial C}{\partial x} \frac{\partial D}{\partial x} + \frac{\partial C}{\partial x} \frac{\partial E_x}{\partial x} + E_y \frac{\partial^2 C}{\partial y^2} \\
+ \frac{E_y}{D} \frac{\partial C}{\partial y} \frac{\partial D}{\partial y} + \frac{\partial C}{\partial y} \frac{\partial E_y}{\partial y} - U \frac{\partial C}{\partial x} - V \frac{\partial C}{\partial y} \\
+ \frac{M}{\rho} + \frac{N}{\rho}
\]  

Figure C.3 shows four grids out of a matrix and defines the points where known values of \( U, V, D, C, E_x, \) and \( E_y \) exist.

The terms of Equation 21 are determined by use of the finite difference method.

1. Term 11 has a first forward difference in time applied to it.
2. Terms 1 and 4 have second-order central differences in time.
3. Terms 9 and 10 are just redefined as \( M_{pp} (I,J) \) and \( N_{pp} (I,J) \) respectively.
4. The remaining terms are solved by the use of the first central difference in space.
FIGURE C.3: Position of Parameters in the Grid Matrix.
Summing and rearranging the solved terms of Equation 21 yields Equation 22.

\[ C'(I,J) = C(I,J) + N_{pp} \Delta t + M_{pp} \Delta t \]  
\[ + \frac{(EX(I,J) - 2C(I,J) - C(IM,J))}{\Delta x^2} \]  
\[ + \frac{(EY(I,J) - 2C(I,J) - C(IM,J))}{\Delta y^2} \]  
\[ + \frac{(C(IP,J) - C(IM,J))}{2\Delta x} \cdot \frac{(EX(IP,J) - EX(IM,J))}{2\Delta x} \]  
\[ + \frac{(C(I,JP) - C(I,JM))}{2\Delta y} \cdot \frac{(EY(I,JP) - EY(I,JM))}{2\Delta y} \]  
\[ + \frac{(EX(I,J) - C(IP,J) - C(IM,J))}{2\Delta x} \cdot \frac{(D(IP,J) - D(IM,J))}{2\Delta x} \]  
\[ + \frac{(EY(I,J) - C(I,JP) - C(I,JM))}{2\Delta y} \cdot \frac{(D(I,JP) - D(I,JM))}{2\Delta y} \]  
\[ - \frac{(U(IP,J) + U(I,J))}{2} \cdot \frac{(C(IP,J) - C(IM,J))}{2\Delta x} \]  
\[ - \frac{(V(IP,J) + V(I,J))}{2} \cdot \frac{(C(I,JP) - C(I,JM))}{2\Delta x} \Delta t \]  
\[ (22) \]
Terms 11 and 12 of Equation 22 have velocity summed and divided by two to get a value at the end of the center of the grid. Thus, the mass transport equation is defined at the center of each grid.

The mass transport equation can be applied as a short term model or it can be applied as a long term model. The short term model studies the transport of substance "P" during a tidal cycle. Thus, it will show the instantaneous concentrations of substance "P". Also, in the short term model, the velocities and depths in Equation 22 become the instantaneous velocities and depths. These values can be calculated by use of a digital computer in a program referred to as a hydraulic model. To minimize the output time of data from the hydraulic model and to minimize the input time of data into the short term model, the short term model is run as a subroutine of the hydraulic program.

The long term model application of the mass transport equation studies the transport of substance "P" for long periods of time, calculating net average rather than instantaneous concentrations. The data used are the tidal cycle average depths and the net tidal cycle velocities.

The magnitude of the net tidal cycle velocity is generally much smaller than the instantaneous velocities for the same point. In fact, the net tidal cycle velocity is usually just the velocity created by sources and sinks in the hydraulic model. For this reason, the net tidal cycle velocities of a region will be on the order of only a few hundredths of a ft/sec. The long term mass transport model requires only one set of velocity and depth data and it is usually run after the hydraulic model of the region has finished. Such a model has the advantage of giving the net concentrations of a region in relatively short running time, compared to the running time of the short
term model. If the short term model is run long enough, it too will yield the net concentrations of a region.

Since Equation 22 is a finite solution of Equation 21, there exists a certain stability to the model. Research in this field performed by Shankar and Masch [10] has yielded the following stability criteria:

\[
\begin{align*}
\Delta x &< \frac{2E_x}{U_{\text{max}}} \\
\Delta y &< \frac{2E_y}{V_{\text{max}}} \\
\Delta t &< \frac{\Delta x^2 \Delta y^2}{2(E_x \Delta y^2 + E_y \Delta x^2)}
\end{align*}
\]

\(U_{\text{max}}\) and \(V_{\text{max}}\) for the short term model become the maximum tidal cycle velocities, while for the long term model they become the net tidal cycle velocities.

In applying Equation 22 to a grid system, the finite solution requires that initial values of concentration be specified. The value specified will be either too high or too low, causing the concentration in each grid to decrease or increase respectively. In either case, the concentration of each grid will approach or oscillate about a particular value of concentration (Figure C.4). When this occurs, the program has reached its steady-state condition. It is this distribution of concentration that should compare favorably with the field data for the area. Once the steady-state distribution has been obtained, an hour simulated by the computer will relate an hour of real time.
FIGURE C.4: Concentration Approaching Steady-State Conditions
Line A shows $C_1$ too small and approaching $C$
Line B shows $C_2$ too large and oscillating about $C$
If $C_1$ and $C_2$ were interchanged, lines A and B would still occur.
DISPERSION COEFFICIENTS

A solution to the mass transport equation cannot be found until some value of dispersion coefficients has been determined. The purposes of this chapter are to mention the numerical method of determining dispersion coefficients, and then to explain the method used to obtain these coefficients for the Tampa Bay estuary.

Taylor [11] found a dimensionless dispersion coefficient for a one-directional uniform turbulent flow in a long straight pipe. He concluded that the dispersion coefficient ($E$) was proportional to the product of shear velocity ($S_*$) and the radius of the pipe ($r_o$), (Equation 26).

$$E = 10.1 \, S_* \, r_o$$  \hspace{1cm} (26)

Equation 26 can be rewritten to express the dispersion coefficients for one-directional open channel flow (Equation 29), by use of the following relationships.

1. Chezy coefficient ($C_h$):

$$C_h = \left( \frac{8g}{f} \right)^{0.5} = \frac{1.49}{n} \, R_h^{1/6}$$  \hspace{1cm} (27)

$f$ = friction, $n$ = Manning's "n", $R_h$ = hydraulic radius

2. Conversion of shear velocity to velocity:

$$S_* = \left( \frac{f}{8} \right)^{1/2}$$  \hspace{1cm} (28)
3. Expressing the pipe radius in terms of its hydraulic radius:

$$E = \frac{77nSR_{\text{h}}^{5/6}}{}$$  \hspace{1cm} (29)

Holly et al. [12] used the modified form of the Taylor equation as a starting point in order to determine the dispersion coefficients in an oscillating flow. Their result is similar to that of Equation 29, except that the velocity is now the maximum value of the absolute value of velocity ($s_{\text{max}}$) and there is a new constant, thus producing Equation 30:

$$E = 100nS_{\text{max}}R_{\text{h}}^{5/6}$$  \hspace{1cm} (30)

Harleman and Lee [13] applied Equation 30 to the Delaware River near Philadelphia and found that $E = 0.2 \text{ mi}^2/\text{day}$ or 234,000 ft$^2$/hour, while the maximum tidal cycle velocity was 2 ft/sec and Manning's "n" = 0.028.

Equation 30 was also applied to research in the tidal portion of the Potomac River. There, with Manning's "n" = 0.035 and a maximum velocity of 1.6 ft/sec, a dispersion coefficient of $0.14 \text{ mi}^2/\text{day}$ or 162,600 ft$^2$/hr. was found.

The dispersion coefficients that are determined by Equation 30 are useful for short term mass transport models as well as the long term model. These coefficients actually describe what is happening and thus the concentrations that are calculated can be related to the real time.
Other work in determining dispersion coefficients has been done by Diachishin [14] and applied successfully by Shankar and Masch [10]. Diachishin compared the tidal mixing of a substance 'P' to a random walk process. He said that the dispersion coefficient was proportional to the product of the square root of the average mean path of a particle ($L_{avg}^2$) and the tidal period ($1/T$).

$$E = \frac{1}{2T} (L_{avg}^2)$$

Shankar and Masch defined $L_{ave}$ as follows:

$$L_{ave} = T \int_0^T |S_t| \, dt$$

$$T_1 = \text{time between slack tides}$$

$$S_t = \text{instantaneous velocity}$$

The dispersion coefficients calculated by Equation 31 and used by Shankar and Masch have no relationship to real time. They can only be used to obtain a long term steady-state distribution of substance 'P'.

The last method to be discussed in determining dispersion coefficients is the experimental method. On October 30, 1971, a field test was conducted under the direction of Dr. Bernard E. Ross, on Upper Old Tampa Bay. The primary objective of the field test was to obtain the velocity data to verify a hydraulic model of the area. To do so, sodium fluorescein dye was dropped into the water and time-recorded photographs were taken of the dye (Volume
II, Appendix IV) [9]. By use of a compensating polar planimeter, the area of the dye was found for several photographs and the dispersion coefficient found by the use of Equation 33.

\[
E = \frac{\text{change in area}}{\text{time increment}} = \frac{\Delta A}{\Delta t}
\]

The results yielded a dispersion coefficient of about 0.12 mi\(^2\)/day or 140,000 ft\(^2\)/hr, which is a real time dispersion coefficient and agrees favorably with those coefficients determined by Harleman and Lee [13] for the Delaware and Potomac Rivers.
APPENDIX D

PROGRAM LISTING: TWO-DIMENSIONAL ECOLOGICAL MODEL

C*****
C*****
C*****
DOUBLE PRECISION RDT,GRO,DGROMX,DRESP,DSETA,DGRAZE,DDKC,DSETC.
1 DDKP1,DSETP1,DEBENP2,DDKN1,DSETP1,DDKN2,DDSETN2,DEBENN2,DEBENN3,
2 DBENOX
COMMON/A/M,N,TIMECT,DX,DT,IPRTDAY,IPRTT,ISPEC,IN,NI,TDAY,DAYLEN
1 /B/U(69.40),V(68.41),DEPTH(68.40)
2 /D/TA(70.42),OLDA(70.42),GRO(70.42),FC,FP,FN,DGROMX,HSATP,HSATN,
3 PHOTO,EXTLI,EXTL2,XSLATI,RDT,DRESP,DSETA,DGRAZE,ZOOEFF
4 /E/TC(70.42),OLDC(70.42),DDKC,DSETC.OXC
5 /F/TP(70.42),OLDP(70.42),DDKP1,DSETP1
6 /G/TP04(70.42),DBENP2(68.40)
7 /H/TN(70.42),OLDN(70.42),DDKN1,DSETN1
8 /I/TNH3(70.42),OLDNH3(70.42),PREF(70.42),DDKN2,PRF,DSETN2,
9 DBENN2(68.40)
10 /J/TNO3(70.42),DBENN3(68.40)
2 /K/TOX(70.42),WINDF,WINDCO,TCOREA,TMP,OXGROW,OXRESP,OXNH3,
3 DBENOX(68.40)
DIMENSION UVEL(69.40),VVEL(68.41)
INTEGER*4 IPRPRT,IPRPLT,IPRCON.ISPAN.LPRT,LPLT,LCON
C*****
10 FORMAT (//,10X,'END OF RUN')
100 FORMAT(//,5X,'***THIS PROGRAM CANNOT RUN BECAUSE THE TOTAL LENGTH
1OF THE RUN ('',I4,' DAYS'),/,'***IS EITHER SHORTER THAN THE NUMBER
2OF DAYS PER PRINTOUT ASKED FOR ('',I4,''),/,'***OR THE NUMBER OF DAYS
3BETWEEN PLOTTING OUTPUTS ('',I4,''),/,'***OR THE NUMBER OF DAYS
4BETWEEN CONCENTRATION FILE OUTPUTS ('',I4,)')
101 FORMAT(//,5X,'***THIS PROGRAM CANNOT RUN BECAUSE THE NUMBER OF MINUTES
1IN AN HOUR ('',F5.0,'') IS NOT EXACTLY DIVISIBLE BY THE TIMESTEP
2YOU WANT ('',F10.4,' MINUTES).')
102 FORMAT(//,5X,'***THIS PROGRAM CANNOT RUN BECAUSE THE NUMBER OF MINUTES
1IN A DAY ('',F5.0,'') IS NOT EXACTLY DIVISIBLE BY THE TIMESTEP YOU
2 WANT ('',F10.4,' MINUTES)')
154 FORMAT(1H1,,12X,'MATHMATICAL ECOLOGICAL MODEL
1(ALGAE, BOD, ORG P, PO4, ORG N, NH3, NO3, DISSLOVED OXYGEN'),//)
200 FORMAT(5X,110,'DAYS')
203 FORMAT(//,30X,'NUMBER OF DAYS SIMULATED ',I4,,30X,'TIME STEP IN M
1MINUTES ',F5.0,,30X,'NUMBER OF TIME STEPS PER DAY ',I4,,30X,
2 'NUMBER OF DAYS SKIPPED BETWEEN PRINTOUTS',I4,,30X,
3 'NUMBER OF HOURS SKIPPED BETWEEN PRINTOUTS',I4,,30X,
4 'NUMBER OF DAYS SKIPPED BETWEEN PLOTS',I4,,30X,
5 'NUMBER OF HOURS SKIPPED BETWEEN PLOTS',I4,,30X,
6 'NUMBER OF DAYS SKIPPED BETWEEN CONCENTRATION OUTPUTS',I4,,30X,
7 'NUMBER OF HOURS SKIPPED BETWEEN CONCENTRATION OUTPUTS',I4,,//)
400 FORMAT(//,,60X,'PROBLEM PARAMETERS',/,,30X,'GRID INCREMENT (1DX) IS',F10.2,' FEET',/,,30X,'GRID INCREMENT (1DY) IS',F10.2,' FEET',/,,30X,'NUMBER OF GRIDS IN X-DIR. (M) IS',I4,,30X,'NUMBER OF GRIDS IN Y-DIR. (N) IS',I4)
500 FORMAT(2F10.0)
501 FORMAT(7I4)
1792 FORMAT(2I3,2F10.0)
2005 FORMAT(4(2I3,I2~3F4.0))

C*****
READ(5,1792) M,N,DX,DY
M1=M+1
N1=N+1
WRITE(6,154)
WRITE(6,400) DX,DY,M,N
READ(5,500) TIMECT, Ti
READ(5,501) KDAYSP,KDYPRT,KHRPRT,KDYPLT,KHRPLT,KDYCON,KHRCON
DAYM=1440.
HOURM=60.
IPRT=0
IF(KDYPRT.GE.0) GO TO 331
IF(KHRPRT.GE.0) GO TO 332
IPRT=1
KDPRT=1ABS(KDYPRT)
331 IF(KHRPRT.GE.0) GO TO 332
IPRT=1
KHPRT=1ABS(KHRPRT)
332 CONTINUE
IF((KDAYSP.GE.KDYPRT).AND.(KDAYSP.GE.KDYPLT).AND.
1 (KDAYSP.GE.KDYCON)) GO TO 301
PRINT 100, KDAYSP,KDYPRT,KDYPLT,KDYCON
STOP
301 IF((KDYPRT.GT.0).AND.(KDYPLT.GT.0).AND.(KDYCON.GT.0)) GO TO 302
IF(AMOD(HOURM;TIMECT).EQ.O.) GO TO 303
PRINT 101, HOURM,TIMECT
STOP
302 CONTINUE
IF(AMOD(DAYM,TIMECT).EQ.O.) GO TO 303
PRINT 102, DAYM,TIMECT
STOP
303 CONTINUE
IPRDAY=INT(DAYM/TIMECT)
IPRHR=INT(HOURM/TIMECT)
IPRPRT=KDYPRT*IPRDAY+KHPRT*IPRHR
IPRPLT=KDYPLT*IPRDAY+KHPRT*IPRHR
IPRCON=KDYCON*IPRDAY+KHPCON*IPRHR
ISPAN=KDAYSP*IPRDAY
LPRT=O
LPFL=O
LCON=O
TDT=TIMECT*60.
C KDAYSP IS THE NUMBER OF DAYS TO BE SIMULATED****
C IPDAYS IS THE NUMBER OF TIME STEPS (TIMECT) IN ONE DAY****
C TIMECT IS THE TIME STEP IN MINUTES ****
C KDYPRT IS THE NUMBER OF DAYS BETWEEN PRINTOUTS
WRITE(6,203) KDAYSP,TIMECT,IPRDAY,KDYPRT,KHRPRT,KDYPLT,KHRPLT,1
KDYCON,KHRCON
C SET THE TIME COUNTER TO ZERO***
DO 81 I=1,M
V(I,N1)=O.
DO 81 J=1,N
Vol II

CONTINUE
DO 82 J=1,N
U(M1,J)=0.
82 CONTINUE
C READ(9) UVEL,VVEL, U,V, DEPTH
C DO 80 LL=1,73
READ(9) ((U(IK,JK), JK=1,N), IK=1,M1), ((V(IK,JK), JK=1,N1), IK=1,M),
1 ((DEPTH(IK,JK), JK=1,N), IK=1,M)
80 CONTINUE
CALL SEWER(1)
CALL SEWER(7)
C******MAJOR TIME ITERATION LOOP **********
IF(IPRPLT.GT.0) CALL OPENLH(XLENGTH, XLENGTH)/* OPEN & INIT LRG HORIZ
PLT
DO 68 KT=1,ISPAN
C PRINT 999, KT,TDAY
999 FORMAT(5X, 'KT=', I10, 5X, 'TDAY=', F10.3)
LPRT=LPRT+1
LPLT=LPLT+1
LCON=LCON+1
TI=TI+TIMECT
C 61 READ(9, END=62) UVEL, VVEL, U, V, DEPTH
61 READ(9, END=62) ((U(IK,JK), JK=1,N), IK=1,M1), ((V(IK,JK), JK=1,N1),
1 IK=1,M), ((DEPTH(IK,JK), JK=1,N), IK=1,M)
GO TO 63
62 REWIND 9
C READ(9) UVEL, VVEL, U, V, DEPTH
C DO 91 LL=1,73
READ(9) ((U(IK,JK), JK=1,N), IK=1,M1), ((V(IK,JK), JK=1,N1), IK=1,M),
1 ((DEPTH(IK,JK), JK=1,N), IK=1,M)
91 CONTINUE
GO TO 61
63 CONTINUE
CALL SEWER(2)
IF(LPRT.NE.IPRPRT) GO TO 304
IF(IPRT.EQ.0) CALL SEWER(3)
IF(IPRT.EQ.1) CALL SEWER(5)
LPRT=0
304 IF(LPLT.NE.IPRPLT) GO TO 305
CALL SEWER(6)
LPLT=0
305 IF(LCON.NE.IPRCON) GO TO 306
CALL SEWER(7)
LCON=0
306 CONTINUE
68 CONTINUE
IF(IPRPLT.GT.0) CALL ENDPPT
WRITE(6,10)
STOP
END
C*****
C*****
SUBROUTINE SEWER(KEY)

DOUBLE PRECISION SORC, DKAY, PROD, EDGE, QSORC, QDKAY, QPROD, QEDGE, QGRID

QINIT, QRESP, QBEN, QGROW, QSET, Q02, QONH3, QRESP2, QBEN2, QGROW2, QSET2,
Q022, QONH32

DOUBLE PRECISION RDT, GRO, DGROMX, DRESP, DSETA, DGRAZE, DDKC, DSETC,
1 DDKP1, DSETP1, DBENP2, DDKN1, DSETN1, DDKN2, DSETN2, DBENN2, DBENN3,
2 DBENOX

DOUBLE PRECISION CONS1, CONS2, CON, TBENOX, TBENP2, TBENN2, TBENN3

DOUBLE PRECISION AVGDEP, CONVOL

COMMON /A/M, N, TIMECT, DX, DY, TDT, IPRDAY, KYPR, T1, M, N1, TDAY, DAYLEN
1 /B/U(69,40), V(68,41), DEPTH(68,40)
2 /C/DOLD(68,40), NSEW, IS(100), JS(100), QS(100), QSC(100), QSP(100),
3 QSP04(100), QSN(100), QSNH3(100), QSN03(100), QSOX(100)
4 /D/TA(70,42), OLDA(70,42), GRO(70,42), FC, FF, FN, DGROMX, HSATP, HSATN,
5 PHOTO, EXTL1, EXTL2, XLIGHT, XSATL1, RDT, DRESP, DSETA, DGRAZE, ZOEEFF
6 /E/TC(70,42), OLC(70,42), DDKC, DSETC, OXC
7 /F/TP(70,42), OLP(70,42), DDKP1, DSETP1
8 /G/TPO4(70,42), DBENP2(68,40)
9 /H/TN(70,42), OLDP(70,42), DDKN1, DSETN1
1 /T/TH3(70,42), OLDP(70,42), PEP(70,42), DDKN2, PEP, DSETN2,
2 DBENN2(68,40)
3 /J/TN3(70,42), DBENN3(68,40)
4 /K/TOX(70,42), WINDP, WINDC, TCOREA, TEMP, OXGROW, OXRESP, OXNH3,
5 DBENOX(68,40)
6 /L/NPP, IPI(10), JPI(10), IPO(10), JPO(10), QPP(10)
7 /M/CONS1, CONS2, BODU5

DIMENSION NTREAT(100), QSBD5(100), QT(4,100), QTP(5,10)

DIMENSION SORC(8), DKAY(8), PROD(8), EDGE(8), QSORC(8), QDKAY(8),
1 QPROD(8), QEDGE(8), QGRID(8), QINIT(8), QRESP(8), QBEN(8), QGROW(8),
2 QSET(8), QRESP2(8), QBEN2(8), QGROW2(8), QSET2(8)

DIMENSION MONDY(12)

REAL*4 TITLE1(4), TITLE2(4), AVGCON(8)

DATA NTREAT/100*0, QSBD5/100*0., QT/400**, 'QTP/50** ' /

DATA SORC, DKAY, PROD, EDGE, QSORC, QDKAY, QPROD, QEDGE, QGRID/72*0. /

DATA QINIT, QRESP, QBEN, QGROW, QSET, QRESP2, QBEN2, QGROW2, QSET2/72*0. /

DATA QONH3, QONH32/2*0./

DATA MONDY/31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31, 30, 31/)

DATA PI/3.14159/

100 FORMAT(2I3)
101 FORMAT(10F8.0)
102 FORMAT(2I3,F7.3,I2,8F6.0,1X,4A4)
103 FORMAT(4I3,F7.0,2X,5A4)
104 FORMAT(15F5.0)
105 FORMAT(4A4)
106 FORMAT(5I4)
200 FORMAT(/,, 47X,'TIME STEP FOR DECAY COMPUTATIONS (TD
1T) = ',F5.0,' SEC',/,, 53X,'NUMBER OF SEWAGE OUTFALLS (NSEW) = ',I4,
2 /,, 57X,'NUMBER OF POWER PLANTS (NPP) = ',I4,/,
3 31X,'MAXIMUM SPECIFIC GROWTH RATE OF ALGAE AT 20 C (GROMAX) = ',
4 F10.4,' (1/DAY)',/,, 45X,
Vol II

Appendix D

Ecological Model

5 'RESPIRATION RATE OF ALGAE AT 20 C (RESP) = ',F10.4,' (1/DAY)',/,6
47X,'GRAZING RATE FOR ALGAE AT 20 C (GRAZE) = ',F10.4,' (1/DAY)',/
7 '/,54X,'ZOOPHANLEKTON EFFICIENCY (ZOOBFF) = ',F10.4,' (MG ASSIM./MG
8GRAZ.)',/
9 25X,'PHOSPHORUS HALF-SATURATION CONSTANT FOR ALGAE GROWTH (HSATP)
1 = ',F10.4,' (MG P/L)',/,27X,'NITROGEN HALF-SATURATION CONSTANT FO
2R ALGAE GROWTH (HSATN) = ',F10.4,' (MG N/L)',/
3 28X,'PHOTOPERIOD -- PERCENT OF THE DAY WHICH IS SUNLIT (PHOTO) =
4',F10.4,' (HUNDREDTHS)',/
5 38X,'LIGHT EXTINCTION COEFFICIENT CONSTANT 1 (EXTL1) = ',F10.4,
6 ' (1/FT)',/,38X,'LIGHT EXTINCTION COEFFICIENT CONSTANT 2 (EXTL2)
7= ',F10.4,' (1/FT/MG A)',/
8/',32X,'AVERAGE LIGHT INTENSITY AT THE WATER SURFACE (XLIGHT) = ',
9 F10.4,' (LANGEYS/DAY)',/
1 50X,'SATURATION LIGHT INTENSITY (XSATLI) = ',F10.4,
2 ' (LANGEYS/DAY)',/,36X,'RATIO OF CHLOROPHYLL A TO THE ALGAE
3 BIOMASS (CHL) = ',F10.4,' (MICROG CHL A/MG A)',/
4 30X,'RATIO OF ORGANIC PHOSPHORUS TO 5-DAY BOD (BODU5) = ',
5 F10.4,' (MG CBODU/MG BOD5)',/
6 12X,'FIRST-ORDER DECAY RATE OF CARBONACEOUS BOD DUE TO OXIDATION
7AT 20 C (DKC) = ',
8 F10.4,' (1/DAY)',/,27X,'FIRST-ORDER DECAY RATE OF ORGANIC
9C PHOSPHORUS AT 20 C (DPK) = ',F10.4,' (1/DAY)',/
1 18X,'FIRST-ORDER DECAY RATE OF ORGANIC NITROGEN TO AMMONIA AT 20
2C (DKN) = ',
3  F10.4,' (1/DAY)',/,25X,'FIRST-ORDER DECAY RATE OF AMMONIA TO NI
4TRATE AT 20 C (DKNH3) = ',F10.4,' (1/DAY)',/
5 52X,'SETTLING VELOCITY OF ALGAE (SETA) = ',F10.4,' (FT/DAY)',/
6 45X,'SETTLING RATE OF CARBONACEOUS BOD (SETC) = ',F10.4,' (1/DAY)
7',/',43X,'SETTLING RATE OF ORGANIC PHOSPHORUS (SETP) = ',F10.4,
8 ' (1/DAY)',/,45X,'SETTLING RATE OF ORGANIC NITROGEN (SETN) = ',
9 F10.4,' (1/DAY)',/,52X,'SETTLING RATE OF AMMONIA (SETNH3) = ',
1 F10.4,' (1/DAY)',/
2 35X,'FRACTION OF THE ALGAE BIOMASS WHICH IS CARBON (FC) = ',
3 F10.4,
4  ' (MG C/MG A)',/,31X,'FRACTION OF THE ALGAE BIOMASS WHICH IS
5PHOSPHORUS (FP) = ',F10.4,' (MG P/MG A)',/,33X,'FRACTION OF THE A
6LGEA BIOMASS WHICH IS NITROGEN (FN) = ',F10.4,' (MG N/MG A)',/
7 23X,'PREFERENCE RATE OF AMMONIA OVER NITRATE FOR ALGAE GROWTH (PR
8F) = ',F10.4,' (HUNDREDTHS)',/
9 34X,'OXYGEN PRODUCTION PER UNIT OF ALGAE GROWTH (OXGROW) = ',
1 F10.4,' (MG O/MG A)',/,36X,'OXYGEN DEMAND PER UNIT OF ALGAE RESPI
2RED (OXRESP) = ',
3 F10.4,' (MG O/MG A)',/,41X,'OXYGEN DEMAND PER UNIT OF CARBON ALGA
4E (OXC) = ',F10.4,' (MG O/MG C)',/,34X,'OXGEN UPTAKE PER UNIT 5
T OF AMMONIA OXIDATION (OXNH3) = ',F10.4,' (MG O/MG N)',/
6 10X,'TEMPERATURE COEFFICIENT FOR THE MAXIMUM SPECIFIC ALGAL GROWT
7H RATE (TOCGRO) = ',F10.4,' (1/DAY)',/,22X,'TEMPERATURE COEFFICIENT FOR THE A
8LGER RESPIRATION RATE (TCORSP) = ',F10.4,' (1/DAY)',/,26X,'TEMPERATURE COEFFI
9ENT FOR THE ALGAL GRAZING RATE (TCOCRZ) = ',F10.4,' (1/FT/MG A),
1 10X,'TEMPERATURE COEFFICIENT FOR THE DECAY OF CARBONACEOUS BOD (TOC)
2 = ',
2 F10.4,' (1/FT/MG A)',/,19X,'TEMPERATURE COEFFICIENT FOR THE DECAY OF ORGANIC PHO
3SPHORUS (TCP) = ',F10.4,' (1/DAY)',/21X,'TEMPERATURE COEFFICIENT FOR THE DE
4CAY OF ORGANIC NITROGEN (TCON) = ',F10.4,' (1/DAY)',/28X,'TEMPERATURE COEFFI
SCIENT FOR THE DECAY OF AMMONIA (TCONH3) = ',F10.4,/,19X,' TEMPERATU
6RE COEFFICIENT FOR THE REAERATION RATE OF OXYGEN (TCOREA) = ',
7 F10.4,/, 
8 65X,'WIND VELOCITY (WIND) = ',F10.4,)' (MPH)',/45X,'WIND COEFFICI
GENT FOR REAERATION (WINDCO) = ',F10.4,/,61X,'WATER TEMPERATURE (TE
1MP) = ',F10.4,)' (DEG C)',/63X,'WATER SALINITY (SALIN) = ',F10.4,
2 ' (PPT)',///
201 FORMAT///,45X,'OUTFALL CONCENTRATIONS (MG/L)',/7X,'I J',2X,
1 'FLOW (MGD)',3X,'BOD-C',1X,
2 'ORG P-P',3X,'PO4-P',1X,'ORG N-N',3X,
3 'NH3-N',3X,'NO3-N',2X,'OXYGEN',3X,'BOD-5',1X,'NTREAT',4X,
4 'OUTFALL',///
202 FORMAT(2X,3I3,F12.3,8F8.2,I5,6X,4A4)
203 FORMAT///,52X,'POWER PLANT DATA',/9X,'INTAKE',5X,'OUTFALL',5X,
1 'FLOW',/11X,'I J',5X,'I J',4X,)'(MGD)',5X,'POWER PLANT',///
204 FORMAT(5X,I3,2X,213,6X,2I3,F12.3,2X,5A4)
205 FORMAT(32X,'SATURATION CONCENTRATION OF DISSOLVED OXYGEN (SATDOX)
1 = ',F10.4,)' (MG/L)',///
206 FORMAT(1H1,///,42X,'BENTHIC OXYGEN DEMAND (MG O/SQ M/DAY)',///
207 FORMAT(2X,'J/I',18(2X,I3,1X),/)
208 FORMAT///,48X,'AVERAGE BENTHIC OXYGEN DEMAND (BENOX) = ',F10.4,
1 ' (MG O/SQ M/DAY)',///
210 FORMAT(1H1,///,33X,'BENTHIC SOURCE RATE FOR ORTHOPHOSPHATE (MG P/SQ
1 M/DAY)',///
211 FORMAT///,30X,'AVERAGE BENTHIC SOURCE RATE FOR ORTHOPHOSPHATE (BEN
1 P04) = ',F10.4,)' (MG P/SQ M/DAY)',///
212 FORMAT(1H1,///,36X,'BENTHIC SOURCE RATE FOR AMMONIA (MG N/SQ M/DAY)
1 = ',///
213 FORMAT///,37X,'AVERAGE BENTHIC SOURCE RATE FOR AMMONIA (BENNH3) =
1 ',F10.4,)' (MG N/SQ M/DAY)',///
214 FORMAT(1H1,///,36X,'BENTHIC SOURCE RATE FOR NITRATE (MG N/SQ M/DAY)
1 = ',///
215 FORMAT///,37X,'AVERAGE BENTHIC SOURCE RATE FOR NITRATE (BENNO3) =
1 ',F10.4,)' (MG N/SQ M/DAY)',///
216 FORMAT///,2(52X,4A4,///,42X,'STARTING TIME = ','I2',':',',I2',' ON ',
1 I2',/',',I2','/',I4','45X,'LATITUDE = ',F10.4,' DEGREES',///
901 FORMAT///,10X,'WELL DONE')
C*****
 IF(KEY.EQ.2) GO TO 19
 IF(KEY.EQ.3) GO TO 138
 IF(KEY.EQ.5) GO TO 138
 IF(KEY.EQ.6) GO TO 139
 IF(KEY.EQ.7) GO TO 140
 READ(5,105) TITLE1
 READ(5,105) TITLE2
 READ(5,100) NSEW,NPP
 READ(5,106) MON,NDAY,NYR,NHR,MIN
 READ(5,101) ELAT
 READ(5,101) AA,AC,AP,APO4,AN,ANH3,ANO3,AOX
 READ(5,101) GROMAX,RESP,SETA,GRAZE,ZOOEFF
 READ(5,101) HSATP,HSATN,EXTL1,EXTL2,XLIGHT,XSATLI,CHL
 READ(5,101) DKC,BODU5,FC,SETC,DKP,FP,SETP
 READ(5,101) DKN,FN,SETN,DKNH3,SETNH3,PRF
READ(5,101) OXGROW,OXRESP,OXC,OXNH3
READ(5,101) TCOGRO,TCORSP,TCOGRZ,TCOC,TCOP,TCON,TCONH3,TCOREA
READ(5,101) WIND,WINDCO,TEMP,SALIN
M2=M+2
N2=N+2
RDT=1./IPRDAY
ELAT=ELAT*PI/180.
NDAYR=0
DO 51 MO=1,12
IF(MO.GT.(MON-1)) GO TO 52
NDAYR=NDAYR+MONDY(MO)
IF((MO.EQ.2).AND.(MOD(NYR,4).EQ.0)) NDAYR=NDAYR+1
51 CONTINUE
52 CONTINUE
DECLIN=0.409*SIN(0.0172*NDAYR-1.3762)
DAYLEN=24./(PI/(ARCOS(-TAN(ELAT)*TAN(DECLIN))))
TDAY=NHR+MIN/60.
PHOTO=DAYLEN/24.
WRITE(6,216) TITLE1,TITLE2,NHR,MIN,MON,NDAY,NYR,ELAT
WRITE(6,200) TDT,NSEW,NPP,GROMAX,RESP,GRAZE,ZOOGEP
1 HSATP,HSATN,PHOTO,EXTL1,EXTL2,XLIGHT,XSATL1,CHL,BODU5,DKC,DKP,DKN
2,DKNH3,SETA,SETC,SETN,SETH3,FC,FP,FN,PRF
3 OXGROW,OXRESP,OXC,OXNH3,TCGRO,TCORSP,TCOGRZ,TCOC,TCOP,TCON,
4TCOH3,TCOREA,WIND,WINDCO,TEMP,SALIN
DGROMX=GROMAX*TCGRO**(TEMP-20.)
DRESP=RESP*TCORSP**(TEMP-20.)
DGRAZE=GRAZE*TCOGRZ**(TEMP-20.)
DDKC=DKC*TCOC**(TEMP-20.)
DDKP1=DKP*TCOP**(TEMP-20.)
DDKN1=DKN*TCON**(TEMP-20.)
DDKN2=DKNH3*TCONH3**(TEMP-20.)
SATDOX=OXSAT(TEMP,SALIN)
WRITE(6,205) SATDOX
DRESP=DRESP*RDT/2.
DSETA=SETA*RDT/2.
DGRAZE=DGRAZE*RDT/2.
DDKC=DDKC*RDT/2.
DSETC=SETC*RDT/2.
DDKP1=DDKP1*RDT/2.
DSETP=SETP*RDT/2.
DDKN1=DDKN1*RDT/2.
DSETN=SETN*RDT/2.
DDKN2=DDKN2*RDT/2.
DSETP2=SETP2*RDT/2.
WIND=WIND*5280./3600.
IF(NSEW.EQ.0) GO TO 9
READ(5,102) (IS(L),JS(L),QS(L),NTREAT(L),QSC(L),QSP(L),QSP04(L),
1 QSN(L),QSNH3(L),QSN03(L),QSOX(L),QSBOD5(L),(QT(K,L),K=1,4),
2 L=1,NSEW)
WRITE(6,201)
DO 10 L=1,NSEW
NTRT=NTREAT(L)
IF(NTRT.EQ.0) GO TO 695
GO TO (604,609,621,639,688),NTRT
Appendix D
Ecological Model

C NTREAT = 1 PRIMARY TREATMENT
604 QSC(L)=183.2
QSP(L)=7.9
QSN(L)=192.8
GO TO 695
C NTREAT = 2 MARGINAL SECONDARY TREATMENT
609 QSC(L)=72.3
QSP(L)=6.3
QSN(L)=173.5
GO TO 695
C NTREAT = 3 HIGH RATE BIOLOGICAL TREATMENT
687 QSC(L)=35.7
QSP(L)=6.3
QSN(L)=154.2
GO TO 695
C NTREAT = 4 SECONDARY TREATMENT WITH NITRIFICATION
621 QSBOD5(L)=25.
QSP(L)=8.
QSP04(L)=0.
QSN(L)=20.
QSNH3(L)=0.
QSN03(L)=0.
QSOX(L)=SATDOX
GO TO 695
C NTREAT = 5 ADVANCED TREATMENT
639 QSBOD5(L)=8.
QSP(L)=2.
QSP04(L)=1.8
QSN(L)=1.7
QSN03(L)=2.5
QSOX(L)=SATDOX
GO TO 695
C NTREAT = 6 SEMI-AWT
688 QSBOD5(L)=8.
QSP(L)=8.
QSP04(L)=7.2
QSN(L)=1.6
QSNH3(L)=3.4
QSN03(L)=5.
QSOX(L)=SATDOX
C NTREAT = 0 OTHER OR NO TREATMENT
695 IF((QSC(L).EQ.0.).AND.(QSBOD5(L).GT.0.)) QSC(L)=QSBOD5(L)*BODU5
IF((QSC(L).GT.0.).AND:(QSBOD5(L).EQ.0.)) QSBOD5(L)=QSC(L)/BODU5
IF(QSOX(L).EQ.0.)QSOX(L)=SATDOX
WRITE(6,202) L,IS(L),JS(L),QS(L),QSC(L),QSP(L),QSP04(L),QSN(L),
QSNH3(L),QSN03(L),QSOX(L),QSBOD5(L),NTREAT(L),(QT(K,L),K=1,4)
TEMPF=1.55*QS(L)*QSC(L)
TEMPF1=1.55*QS(L)*QSP(L)
TEMPF2=1.55*QS(L)*QSP04(L)
TEMP1=1.55*QS(L)*QSN(L)
TEMP2=1.55*QS(L)*QSNH3(L)
TEMP3=1.55*QS(L)*QSN03(L)
TEMP4=1.55*QS(L)*QSOX(L)
QSC(L) = TEMPC / (DX * DY)
QSP(L) = TEMPP1 / (DX * DY)
QSP04(L) = TEMPP2 / (DX * DY)
QSN(L) = TEMPN1 / (DX * DY)
QSNH3(L) = TEMPN2 / (DX * DY)
QSN03(L) = TEMPN3 / (DX * DY)
QSOX(L) = TEMPOX / (DX * DY)

10 CONTINUE
9 CONTINUE
IF (NPP .EQ. 0) GO TO 82
WRITE (6, 203)
DO 81 LP = 1, NPP
READ (5, 103) IPI(LP), JPI(LP), IPO(LP), JPO(LP), QPP(LP), (QTP(K, LP),
1 K = 1, 5)
WRITE (6, 204) LP, IPI(LP), JPI(LP), IPO(LP), JPO(LP), QPP(LP), (QTP(K, LP),
1 K = 1, 5)
QPP(LP) = QPP(LP) * 1.55 / (DX * DY)

81 CONTINUE
82 CONTINUE
DO 81 I = 1, M
DO 81 J = 1, N
IF (DEPTH(I, J) .GT. 100.) GO TO 11
IT = I + 1
JT = J + 1
TA(IT, JT) = AA
TC(IT, JT) = AC
TP(IT, JT) = AP
TP04(IT, JT) = APO4
TN(IT, JT) = AN
TNH3(IT, JT) = ANH3
TN03(IT, JT) = AN03
TOX(IT, JT) = AOX

11 CONTINUE
70 READ (8, END = 71) ((TA(I, J), I = 1, M2), J = 1, N2), ((TC(I, J), I = 1, M2), J = 1, N2),
1 ((TP(I, J), I = 1, M2), J = 1, N2), ((TP04(I, J), I = 1, M2), J = 1, N2),
2 ((TN(I, J), I = 1, M2), J = 1, N2), ((TNH3(I, J), I = 1, M2), J = 1, N2),
3 ((TN03(I, J), I = 1, M2), J = 1, N2), ((TOX(I, J), I = 1, M2), J = 1, N2)
GO TO 70

71 CONTINUE
DO 72 J = 1, N
DO 72 I = 1, M
IF (DEPTH(I, J) .GT. 100.) GO TO 72
I1 = I + 1
J1 = J + 1
ALGBOD = TA(I1, J1) * FC * OXC
IF (TC(I1, J1) .LT. ALGBOD) TC(I1, J1) = ALGBOD

72 CONTINUE
TA(48, 1) = .4
TA(53, 1) = .4
TC(48, 1) = TA(48, 1) * FC * OXC
TC(53, 1) = TA(53, 1) * FC * OXC
TP(48, 1) = TA(48, 1) * FP
TP(53, 1) = TA(53, 1) * FP
TN(48, 1) = TA(48, 1) * FN
TN(53, 1) = TA(53, 1) * FN
TOX(48, 1) = (SATDOX + TOX(48, 2)) * .5
TOX(53, 1) = (SATDOX + TOX(53, 2)) * .5
DO 83 I = 57, 68
   TA(I + 1, 1) = .4
   TC(I + 1, 1) = TA(I + 1, 1) * FC * OXC
   TP(I + 1, 1) = TA(I + 1, 1) * FP
   TN(I + 1, 1) = TA(I + 1, 1) * FN
83 TOX(I + 1, 1) = (SATDOX + TOX(I + 1, 2)) * .5
DO 84 J = 1, 9
   TA(M + 2, J + 1) = .4
   TC(M + 2, J + 1) = TA(M + 2, J + 1) * FC * OXC
   TP(M + 2, J + 1) = TA(M + 2, J + 1) * FP
   TN(M + 2, J + 1) = TN(M + 2, J + 1) * FN
84 TOX(M + 2, J + 1) = (SATDOX + TOX(M + 1, J + 1)) * .5
AVGDEP = 0.
NDEP = 0
DO 12 J = 1, N
   DO 12 I = 1, M
      IF(DEPTH(I, J).GT.100.) GO TO 12
      AVGDEP = AVGDEP + DEPTH(I, J)
      NDEP = NDEP + 1
      DOLD(I, J) = DEPTH(I, J)
      IT = I + 1
      JT = J + 1
      OLDA(IT, JT) = TA(IT, JT)
      OLDTC(IT, JT) = TC(IT, JT)
      OLDP(IT, JT) = TP(IT, JT)
      OLDN(IT, JT) = TN(IT, JT)
      OLDN(H3)(IT, JT) = TNH3(IT, JT)
      QGRID(1) = QGRID(1) + DEPTH(I, J) * TA(IT, JT)
      QGRID(2) = QGRID(2) + DEPTH(I, J) * TC(IT, JT)
      QGRID(3) = QGRID(3) + DEPTH(I, J) * TP(IT, JT)
      QGRID(4) = QGRID(4) + DEPTH(I, J) * TP04(IT, JT)
      QGRID(5) = QGRID(5) + DEPTH(I, J) * TN(IT, JT)
      QGRID(6) = QGRID(6) + DEPTH(I, J) * TNH3(IT, JT)
      QGRID(7) = QGRID(7) + DEPTH(I, J) * TNO3(IT, JT)
      QGRID(8) = QGRID(8) + DEPTH(I, J) * TOX(IT, JT)
12 CONTINUE
   AVGDEP = AVGDEP / NDEP
   DSET = DSETC * AVGDEP
   DSETP1 = DSETP1 * AVGDEP
   DSETN1 = DSETN1 * AVGDEP
   DSETN2 = DSETN2 * AVGDEP
   CONS1 = 64.0 * 1.E-06
   CONS2 = CONS1 * DX * DY
   DO 13 L = 1, 8
      QGRID(L) = QGRID(L) * CONS2
      QINIT(L) = QGRID(L)
13 CONTINUE
   DO 14 I = 1, M
14 READ(5, 104) (DBENOX(I, J), J = 1, N)
   DO 401 NSTRT = 1, N, 18
      NSTOP = NSTRT + 17
IF(NSTOP.GT.N) NSTOP=N
WRITE(6,206)
WRITE(6,207) (J,J=NSTRT,NSTOP)
DO 401 I=1,M
  401 WRITE(6,208) (I,(DBENOX(I,J),J=NSTRT,NSTOP))
  DO 501 I=1,M
    501 READ(5,104) (DBENP2(I,J),J=1,N)
    DO 502 NSTRT=1,N,18
    NSTOP=NSTRT+17
    IF(NSTOP.GT.N) NSTOP=N
    WRITE(6,210)
    WRITE(6,207) (J,J=NSTRT,NSTOP)
    DO 502 I=1,M
  502 WRITE(6,208) (I,(DBENP2(I,J),J=NSTRT,NSTOP))
  DO 503 I=1,M
    503 READ(5,104) (DBENN2(I,J),J=1,N)
    DO 504 NSTRT=1,N,18
    NSTOP=NSTRT+17
    IF(NSTOP.GT.N) NSTOP=N
    WRITE(6,212)
    WRITE(6,207) (J,J=NSTRT,NSTOP)
    DO 504 I=1,M
  504 WRITE(6,208) (I,(DBENN2(I,J),J=NSTRT,NSTOP))
  DO 505 I=1,M
    505 READ(5,104) (DBENN3(I,J),J=1,N)
    DO 506 NSTRT=1,N,18
    NSTOP=NSTRT+17
    IF(NSTOP.GT.N) NSTOP=N
    WRITE(6,214)
    WRITE(6,207) (J,J=NSTRT,NSTOP)
    DO 506 I=1,M
  506 WRITE(6,208) (I,(DBENN3(I,J),J=NSTRT,NSTOP))
  TBENOX=0.
  TBENP2=0.
  TBENN2=0.
  TBENN3=0.
  NBEN=0
  CON=RDT*3.28084/1000.
  DO 402 J=1,N
  DO 402 I=1,M
    IF(DEPTH(I,J).GT.100.) GO TO 402
    TBENOX=TBENOX+DBENOX(I,J)
    TBENP2=TBENP2+DBENP2(I,J)
    TBENN2=TBENN2+DBENN2(I,J)
    TBENN3=TBENN3+DBENN3(I,J)
    NBEN=NBEN+1
    DBENOX(I,J)=DBENOX(I,J)*CON
    DBENP2(I,J)=DBENP2(I,J)*CON
    DBENN2(I,J)=DBENN2(I,J)*CON
    DBENN3(I,J)=DBENN3(I,J)*CON
  402 CONTINUE
  BENOX=TBENOX/NBEN
  BENP04=TBENP2/NBEN
  BENNH3=TBENN2/NBEN

Appendix D
Ecological Model
BENNO3 = TBENN3/NBEN
WRITE(6,209) BENOX
WRITE(6,211) BENPO4
WRITE(6,213) BENNH3
WRITE(6,215) BENNO3
RETURN
19 CONTINUE
TDAY = TDAY + TIMECT / 60.
IF (TDAY .GT. 24.) TDAY = TDAY - 24.
CALL ALGAE (DKAY(1), EDGE(1), QRESP(1), QGROW(1), QSET(1))
TDAY = TDAY
CALL BODC (SORC(2), DKAY(2), EDGE(2), QRESP(2), QSET(2))
TDAY = TDAY
CALL ORGP (SORC(3), DKAY(3), EDGE(3), QRESP(3), QSET(3))
TDAY = TDAY
CALL PHOS (SORC(4), PROD(4), EDGE(4), QGROW(4), QBEN(4))
TDAY = TDAY
CALL ORGN (SORC(5), DKAY(5), EDGE(5), QRESP(5), QSET(5))
TDAY = TDAY
CALL AMMON (SORC(6), DKAY(6), PROD(6), EDGE(6), QGROW(6), QBEN(6))
1 QBEN(6)
TDAY = TDAY
CALL TRATE (SORC(7), PROD(7), EDGE(7), QGROW(7), QBEN(7))
TDAY = TDAY
CALL OXYGEN (SATDOX, SORC(8), DKAY(8), PROD(8), EDGE(8), QO2,
2 QGROW(8), QRESP(8), QBEN(8), QONH3)
TDAY = TDAY
DO 20 J = 1, N
DO 20 I = 1, M
IF (DEPTH(I, J) .GT. 100.) GO TO 20
DOLD(I, J) = DEPTH(I, J)
20 CONTINUE
DO 21 J = 2, N
DO 21 I = 2, M
IF (DEPTH(I-1, J-1) .GT. 100.) GO TO 21
OLDA(I, J) = TA(I, J)
OLDC(I, J) = TC(I, J)
OLDP(I, J) = TP(I, J)
OLDN(I, J) = TN(I, J)
OLDNH3(I, J) = TNH3(I, J)
21 CONTINUE
RETURN
138 CONTINUE
CALL MAPALG (1, TA, CHL, QINIT(1), DKAY(1), EDGE(1), QRESP(1), QGROW(1),
1 QSET(1), QDKAY(1), QEDGE(1), QRESP2(1), QGROW2(1), QSET2(1),
2 QGRID(1), KEY)
CALL MAPALG (2, TA, CHL, QINIT(1), DKAY(1), EDGE(1), QRESP(1), QGROW(1),
1 QSET(1), QDKAY(1), QEDGE(1), QRESP2(1), QGROW2(1), QSET2(1),
2 QGRID(1), KEY)
CALL MAPCON (2, TC, QINIT(2), SORC(2), DKAY(2), PROD(2), EDGE(2), QRESP(2),
1 QBEN(2), QGROW(2), QSET(2), QSORC(2), QDKAY(2), QPROD(2), QEDGE(2),
2 QRESP2(2), QBEN2(2), QGROW2(2), QSET2(2), QGRID(2), KEY)
CALL MAPCON (9, TC, QINIT(2), SORC(2), DKAY(2), PROD(2), EDGE(2), QRESP(2),
1 QBEN(2), QGROW(2), QSET(2), QSORC(2), QDKAY(2), QPROD(2), QEDGE(2),
2 QRESP2(2), QBEN2(2), QGROW2(2), QSET2(2), QGRID(2), KEY)
Vol II 93 Appendix D

Ecological Model

2 QRESP2(2), QBEN2(2), QGROW2(2), QSET2(2), QGRID(2), KEY)
CALL MAPCON(3, TP, QINIT(3), SORC(3), DKAY(3), PROD(3), EDGE(3), QRESP(3),
1, QBEN(3), QGROW(3), QSET(3), QSORC(3), QDKAY(3), QPROD(3), QEDGE(3),
2 QRESP2(3), QBEN2(3), QGROW2(3), QSET2(3), QGRID(3), KEY)
CALL MAPCON(4, TPO4, QINIT(4), SORC(4), DKAY(4), PROD(4), EDGE(4),
1 QRESP(4), QBEN(4), QGROW(4), QSET(4), QSORC(4), QDKAY(4), QPROD(4),
2 QEDGE(4), QRESP2(4), QBEN2(4), QGROW2(4), QSET2(4), QGRID(4), KEY)
CALL MAPCON(5, TN, QINIT(5), SORC(5), DKAY(5), PROD(5), EDGE(5), QRESP(5),
1, QBEN(5), QGROW(5), QSET(5), QSORC(5), QDKAY(5), QPROD(5), QEDGE(5),
2 QRESP2(5), QBEN2(5), QGROW2(5), QSET2(5), QGRID(5), KEY)
CALL MAPCON(6, TNH3, QINIT(6), SORC(6), DKAY(6), PROD(6), EDGE(6),
1 QRESP(6), QBEN(6), QGROW(6), QSET(6), QSORC(6), QDKAY(6), QPROD(6),
2 QEDGE(6), QRESP2(6), QBEN2(6), QGROW2(6), QSET2(6), QGRID(6), KEY)
CALL MAPCON(7, TN03, QINIT(7), SORC(7), DKAY(7), PROD(7), EDGE(7),
1 QRESP(7), QBEN(7), QGROW(7), QSET(7), QSORC(7), QDKAY(7), QPROD(7),
2 QEDGE(7), QRESP2(7), QBEN2(7), QGROW2(7), QSET2(7), QGRID(7), KEY)
CALL MAPDOX(8, TOX, SATDOX, QINIT(8), SORC(8), PROD(8), EDGE(8),
1 QBPEN(8), QONH3, Q02, QGROW(8), QRESP(8), QSORC(8), QDKAY(8), QPROD(8),
2 QEDGE(8), QBEN2(8), QONH32, Q022, QGROW2(8), QRESP2(8), QGRID(8), KEY)
RETURN

139 CONTINUE
AVGDEP=0.
DO 31 J = 1, N
DO 31 I = 1, M
IF(DEPTH(I,J).GT.100.) GO TO 31
AVGDEP=AVGDEP+DEPTH(I,J)
31 CONTINUE
CONVOL=1./(AVGDEP*CONS2)
DO 32 L = 1, 8
AVGCON(L)=QGRID(L)*CONVOL
32 CONTINUE
CALL PLOTBX(QINIT, QGRID, QSORC, QDKAY, QPROD, QEDGE, QGROW2, QRESP2,
1 QSET2, QBEN2, QO22, QONH32, QROMAX, RESP, GRAZE, ZOBEFF, HSATP, HSATN,
2 PHOTO, EXTL1, EXTL2, XLIGHT, XSATL1, CHL, BODU5, DKC, DKP, DKN, DKNH3, SETA,
3 SETA, SETP, SETN, SETNH3, BENP04, BENNH3, BENNO3, FC, FP, FN, PRF, OXGROW,
4 OXRESP, OXC, OXNH3, BENOX, TCOGRO, TCORSP, TCOCRZ, TCOC, TCOP, TCN, TCN03,
5 , TCOREA, WIND, Windco, TEMP, SALIN, TI, TITLE1, TITLE2, AVGCON)
RETURN

140 CONTINUE
WRITE(7) ((TA(I,J), I=1, M2), J=1, N2), ((TC(I,J), I=1, M2), J=1, N2),
1 , ((TP(I,J), I=1, M2), J=1, N2), ((TP04(I,J), I=1, M2), J=1, N2),
2 (TN(I,J), I=1, M2), J=1, N2), ((TNH3(I,J), I=1, M2), J=1, N2),
3 (TN03(I,J), I=1, M2), J=1, N2), ((TOX(I,J), I=1, M2), J=1, N2)
RETURN

FUNCTION TAN(X)

TAN=SIN(X)/COS(X)
RETURN

END

C*****
FUNCTION ARCOS(X)

SINE = SQRT(1 - X*X)
ARCOS = ATAN2(SINE, X)
RETURN
END

FUNCTION OXSAT(TEMP, SALIN)

DIMENSION A(7)

DATA A/-173.0722, -21.8492, -0.33096, 249.6339, 143.3483, 0.14259, 7000E-21

TZ = (TEMP + 273.15) / 100.
P1 = A(1) + A(2) * TZ + A(4) / TZ + A(5) * ALOG(TZ)
P2 = A(3) + A(6) * TZ + A(7) * TZ * TZ
OXSAT = EXP(P1 + P2 * SALIN)
RETURN
END

SUBROUTINE ALGAE(QGRAZ, EDGE, QRESP, QGROW, QSET)

DOUBLE PRECISION QGRAZ, EDGE, QRESP, QGROW, QSET
DOUBLE PRECISION RDT, GRO, DGROMX, DRESP, DSETA, DGRAZE
COMMON /A/M, N, TIMECT, DX, DT, HDFDAY, KDIPRT, TI, M1, N1, TDAY, DAYLEN
1 /B/U(69, 40), V(68, 41), DEPTH(68, 40)
2 /C/DOLD(68, 40), NSEW, IS(100), JS(100), QS(100), QSC(100), QSP(100),
3 QSP04(100), QSN(100), QSNH3(100), QSNO3(100), QSOX(100)
4 /D/TA(70, 42), OLDA(70, 42), GRO(70, 42), FC, FP, FN, DGROMX, HSATP, HSATN,
5 PHOT0, EXTL1, EXTL2, XLIGHT, XSATLI, RDT, DRESP, DSETA, DGRAZE, ZOOEFF
6 /L/NPP, IPI(10), JPI(10), IPO(10), JPO(10), QPP(10)
DIMENSION A(70), QIN(10)

DOUBLE PRECISION GAMMA, GPL, GMIN, GPLUS, GMINUS, GROW, GR

DATA A/70*0., QIN/10*0. /

GAMMA = DRESP + DGRAZE
GPL = 1. + GAMMA
GMIN = 1. - GAMMA

DO 1 I = 1, M
IF((I .NE. 47).AND.(I .NE. 52).AND.(I .LT. 57)) GO TO 1

TE = TA(I + 1, 2)
TL = 2.*TE - TA(I + 1, 3)
IF(TL .LE. 0.) TL = 1.*TE
IF(TL .GE. TE) TL = 9.*TE
TA(I + 1, 1) = TL
1 CONTINUE
DO 2 I=1,M
TE=TA(I+1,N+1)
TL=2.*TE-TA(I+1,N)
IF(TL.LT.0.) TL=.1*TE
IF(TL.GT.;TE) TL=.9*TE
TA(I+1,N+1)=TL
2 CONTINUE
DO 3 J=1,N
TE=TA(M1,J+1)
TL=2.*TE-TA(M,J+1)
IF(TL.LT.0.) TL=.1*TE
IF(TL.GT.0.0.) TL=.9*TE
TA(M+1,J+1)=TL
3 CONTINUE
DO 4 J=1,N
TE=TA(2,J+1)
TL=2.*TE-TA(3,J+1)
IF(TL.LT.0.) TL=.1*TE
IF(TL.GT.0.0.) TL=.9*TE
TA(1,J+1)=TL
4 CONTINUE
IF(NPPL.EQ.0) GO TO 6
DO 5 LP=1,NPP
5 QIN(LP)=0.
6 CONTINUE
LP1=1
LP0=1
DO 30 J=1,N
DO 30 I=1,M
IT=I+1
JT=J+1
HOLD=A(IT)
IF(DEPTH(I,J).LT.100.) GO TO 21
A(IT)=1.E06
GO TO 29
21 DNEW=DEPTH(I,J)
DOLDS=DOLD(I,J)
TMP=TA(IT,JT)
IF(DNEW.GT.2) GO TO 2746
A(IT)=TMP
DNEW=.2
GO TO 4219
2746 CONTINUE:
T1=TMP
T2=TMP
T3=TMP
T4=TMP
U1=U(I,J)
U3=U(I+1,J)
V2=V(I,J)
V4=V(I,J+1)
IF(U1.GT.0.) T1=TA(IT-1,JT)
IF(U3.LT.0.) T3=TA(IT+1,JT)
Vol II 96

Ecological Model

NJ@+1%'A%4%1 A1BA?@NA&LAU[])
 IF(V4.LT.0.) T4=TA(IT,JT+1)
 IF(JT.EQ.2) EDGE=EDGE-V2*T2*DX
 IF(IT.EQ.1) EDGE=EDGE+U3*T3*DY
 IF(IT.EQ.2) EDGE=EDGE-U1*T1*DY
 IF(JT.EQ.1) EDGE=EDGE+V4*T4*DX
 TX=(U1*T1-U3*T3)/DX
 TY=(V2*T2-V4*T4)/DY
 IF((I.NE.IPI(LPI)).OR.(J.NE.JPI(LPI)).OR.(LPI.GT.NPP)) GO TO 24
 IF(QIN(LPI).EQ.0.) QIN(LPI)=TMP*QPP(LPI)
 TX=TX-QIN(LPI)
 LPI=LPI+1
 24 IF((I.NE.IPO(LPO)).OR.(J.NE.JPO(LPO)).OR.(LPO.GT.NPP)) GO TO 23
 IF(QIN(LPO).EQ.0.) QIN(LPO)=TA(IPI(LPO)+1,JPI(LPO)+1)*QPP(LPO)
 LPO=LPO+1
 23 CONTINUE
 22 CONTINUE
 EXTL=EXTL1+EXTL2*TA(IT,JT)
 GRO(IT,JT)=GROW(IT,JT,DOLDS,DNEW,DGROMX,HSATP,HSATN,PHOTO,EXTL,
 1 XLIGHT,XTATL,TDAY,DAYLEN)*RDT/2.
 GR=GRO(IT,JT)
 GPLUS=GPL-GR+DSETA/DNEW
 GMINUS=GMIN+GR-DSETA/DOLDS
 A(IT)=(DOLDS*GMINUS+TMP+DT*(TX+TY))/(DNEW*GPLUS)
 IF(A(IT).LT.0.) A(IT)=0.
 AD=TMP*DOLDS+A(IT)*DNEW
 QGROW=QGROW+AD*GR
 QGRAZ=QGRAZ+AD*DGRAZE
 QRESP=QRESP+AD*DRESP
 QSET=QSET+DSETA/DNEW
4219 CONTINUE
 29 IF(JT.GT.2) TA(IT,J)=HOLD
 30 CONTINUE
 DO 31 IT=2,M1
 31 TA(IT,M1)=A(IT)
 RETURN

C*****
C*****
C*****
FUNCTION GROW(I,J,DOLDS,DNEW,DGROMX,HSATP,HSATN,PHOTO,EXTL,XLIGHT,
 1 XSATL,TDAY,DAYLEN)
C*****
DOUBLE PRECISION GROW,DGROMX,DBENN3,DDKN2,DSETN2,DBENN2,DBENP2
 COMMON/J/TNO3(70,42),DBENN3(68,40)
 1 /1/TNH3(70,42),OLDNH3(70,42),PREF(70,42),DDKN2,PRF,DSETN2,
 2 DBENN2(68,40)
 3 /G/TP04(70,42),DBENP2(68,40)
C*****
DATA E/2.71828/,PI/3.14159/
C*****
CP2=TP04(I,J)
CN2=TNH3(I,J)
ECOLOGICAL MODEL

Vol II 97

Appendix D

CN3=TNO3(I,J)
RATIO=XLIGHT*DAYLIG(TDAY,DAYLEN)/XSATL1
EX=EXTL*(DOLD+DNEW)/2.
GROW=DGROMX*CP2/(CP2+HSATP)*(CN2+CN3)/(CN2+CN3+HSATN)*
1 (E/EX)*(EXP(-RATIO*EXP(-EX))-EXP(-RATIO))
C IF(GROW.GT.DGROMX) GROW=DGROMX
RETURN
END

FUNCTION DAYLIG(TDAY,DAYLEN)
DATA PI=3.141591
TOD=TDAY-12.
DAYLIG=0.
IF(TOD.LT.(-DAYLEN/2.)) RETURN
IF(TOD.GT.(DAYLEN/2.)) RETURN
DAYLIG=1.+(COS((2.*PI*TOD)/DAYLEN)
RETURN
END

SUBROUTINE BODC(SORC,DKAY,EDGE,QRESP,QSET)
DOUBLE PRECISION SORC,DKAY,EDGE,QRESP,QSET
DOUBLE PRECISION RDT,GRO,DGROMX,DRESP,DSETA,DGRAZE,DDKC,DSETC
COMMON/A/M,N,TMECT,DX,DY,TDT,IPRDAY,KDYPRT,IM1,N1,TDAY,DAYLEN
/B/U(69,40),V(68,41),DEPTH(68,40)
/C/DOLD(68,40),NSEW,JS(100),JS(100),QS(100),QSP(100),QSPO4(100),QSN(100),QSN3(100),QSOX(100)
/D/TA(70,42),OLDA(70,42),GRO(70,42),FC,FP,FN,DGROMX,HSATP,HSATN,
PHOTO,EXTL1,EXTL2,XL1GHT,XSATL1,RDT,DRESP,DSETA,DGRAZE,ZOOEFF
/E/TC(70,42),OLDC(70,42),DDKC,DSETC,OXC
/L/NPP,IP(10),JPI(10),IP0(10),JPO(10),QPP(10)
DOUBLE PRECISION GAMMA,GPLUS,GMINUS,RES
DIMENSION A(70),QIN(10)
DATA A/70*0.,QIN/10*0./

DO 1,I=1,M
DO 2 I=1,M
CONTINUE
1 CONTINUE
2 CONTINUE
C IF(TL.GT.TE) TL=.9*TE
C TC(I+1,N+2)=TL
C 2 CONTINUE
C DO 3 J=1,N
C IF(J.GT.9) GO TO 3
C TE=TC(M1,J+1)
C TL=2.*TE-TC(M,J+1)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.TE) TL=.9*TE
C TC(M+2,J+1)=TL
C 3 CONTINUE
C DO 4 J=1,N
C TE=TC(2,J+1)
C TL=2.*TE-TC(3,J+1)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.TE) TL=.9*TE
C TC(1,J+1)=TL
C 4 CONTINUE
C L=1
C IF(NPP.LE.0) GO TO 6
C DO 5 LP=1:NPP
C 5 CONTINUE
C 6 CONTINUE
C LPI=1
C LPO=1
C DO 30 J=1,N
C DO 30 I=1,M
C IT=I+1
C JT=J+1
C HOLD=A(IT)
C IF(DEPTH(I,J).LT.100.) GO TO 21
C A(IT)=1.E06
C GO TO 29
C 21 DNEW=DEPTH(I,J)
C DOLDS=DOLD(I,J)
C TMP=TC(IT,JT)
C IF(DNEW.GT.0.) GO TO 2746
C A(IT)=TMP
C DNEW=.2
C GO TO 4219
C 2746 CONTINUE
C T1=TMP
C T2=TMP
C T3=TMP
C T4=TMP
C U1=U(I,J)
C U3=U(I+1,J)
C V2=V(I,J)
C V4=V(I,J+1)
C IF(U1.GT.0.) T1=TC(IT-1,JT)
C IF(U3.LT.0.) T3=TC(IT+1,JT)
C IF(V2.GT.0.) T2=TC(IT,JT-1)
C IF(V4.LT.0.) T4=TC(IT,JT+1)
C IF(JT.EQ.2) EDGE=EDGE-V2*T2*DX
IF(IT.EQ.M1) EDGE=EDGE+U3*T3*DY
IF(IT.EQ.2) EDGE=EDGE-U1*T1*DY
IF(JT.EQ.N1) EDGE=EDGE+V4*T4*DX
TX=(U1*T1-U3*T3)/DX
TY=(V2*T2-V4*T4)/DY
IF((I.NE.IPI(LPI)).OR.(J.NE.JPI(LPI)).OR.(LPI.GT.NPP)) GO TO 24
IF(QIN(LPI).EQ.0) QIN(LPI)=TMP*QPP(LPI)
TX=TX-QIN(LPI)
SORC=SORC-QIN(LPI)
LPI=LPI+1
24 IF((I.NE.IPO(LPO)).OR.(J.NE.JPO(LPO)).OR.(LPO.GT.NPP)) GO TO 23
IF(QIN(LPO).EQ.0) QIN(LPO)=TC(IPI(LPO)+1,JPI(LPO)+1)*QPP(LPO)
TX=TX+QIN(LPO)
SORC=SORC+QIN(LPO)
LPO=LPO+1
23 IF((I.N.E.IS(L)).OR.(J.NE.JS(L)).OR.(L.GT.NSEW)) GO TO 22
TX=TX+QSC(L)
SORC=SORC+QSC(L)
L=L+1
GO TO 23
22 AD=OLDA(IT,JT)*DOLDS+TA(IT,JT)*DNEW
ADM=TA(IT,JT)*DNEW-OLDA(IT,JT)*DOLDS
RES=((1.-Z00EFF)*DGRAZE*AD+ADM)*FC*OXC
GAMMA=DDKC*F1(IT,JT)
GPLUS=1.+GAMMA+DSETC/DNEW
GMINUS=1.-GAMMA-DSETC/DOLDS
A(IT)=(DOLDS*GMINUS*TMP+RES+TDT*(TX+TY))/(DNEW*GPLUS)
IF(A(IT).LT.0.) A(IT)=0.
DKAY=DKAY+(TMP*DOLDS+A(IT)*DNEW)*GAMMA
QRESP=QRESP+RES
QSET=QSET+DSETC*(TMP+A(IT))
4219 CONTINUE
29 IF(JT.GT.2) TC(IT,J)=HOLD
30 CONTINUE
DO 31 IT=2,M1
31 TC(IT,N1)=A(IT)
RETURN
C*****
C*****
FUNCTION F1(I,J)
C*****
DOUBLE PRECISION DBENOX
COMMON/K/TOX(70,42),WINDF,WINDCO,TCOREA,TEMP,OXGROW,OXRESP,OXNH3,
1 DBENOX(68,40)
C*****
F1=TOX(I,J)/(TOX(I,J)+0.01)
RETURN
END
C*****
C*****
SUBROUTINE ORGP(SORC,DKAY,EDGE,QRESP,QSET)
**Ecological Model**

DOUBLE PRECISION SORC, DKAY, EDGE, QRESP, QSET
DOUBLE PRECISION RDT, GRO, DGROMX, DRESP, DSETA, DGRAZE, DDKP1, DSETP1
COMMON/A/M, N, TIMECT, DX, DY, TDT, IPRDAY, KDYPRT, TI, M1, N1, TDAY, DAYLEN
1 /B/U(69,40), V(68,41), DEPTH(68,40)
2 /C/DOLD(68,40), NSEW, IS(100), JS(100), QS(100), QSC(100), QSP(100),
3 QSP04(100), QSN(100), QSNH3(100), QSN3(100), QSOX(100),
4 /D/TA(70,42), OLDA(70,42), GRO(70,42), FC, FP, FN, DGROMX, HSATP, HSATN,
5 PHOTO, EXTL1, EXTL2, XLIGHT, XSATL1, RDT, DRESP, DSETA, DGRAZE, ZOOEFF
6 /F/TP(70,42), OLDP(70,42), DDKP1, DSETP1
7 /L/NPP, IPI(10), JPI(10), IPO(10), JPO(10), QPP(10)

DOUBLE PRECISION GAMMA, GPL, GMIN, GPLUS, GMINUS, RES
DIMENSION A(70), QIN(10)

DATA A/70*0., QIN/10*0./

GAMMA=DDKP1
GPL=1.+GAMMA
GMIN=1.-GAMMA

C DO 1 I=1,M
C IF((I.LE.47).AND.(I.LE.52).AND.(I.LT.57)) GO TO 1
C TE=TP(I+1,2)
C TL=2.*TE-TP(I+1,3)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.5E) TL=.9*TE
C TP(I+1,1)=TL
C 1 CONTINUE
C DO 2 I=1,M
C TE=TP(I+1,N+1)
C TL=2.*TE-TP(I+1,N)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.5E) TL=.9*TE
C TP(I+1,N+2)=TL
C 2 CONTINUE
C DO 3 J=1,N
C IF(J.GT.9) GO TO 3
C TE=TP(M1,J+1)
C TL=2.*TE-TP(M,J+1)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.5E) TL=.9*TE
C TP(M+2,J+1)=TL
C 3 CONTINUE
C DO 4 J=1,N
C TE=TP(2,J+1)
C TL=2.*TE-TP(3,J+1)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.5E) TL=.9*TE
C TP(1,J+1)=TL
C 4 CONTINUE
L=1
IF(NPP.LE.0) GO TO 6
DO 5 LP=1, NPP
5 QIN(L)=0.
6 CONTINUE
LPI=1
LPO=1
DO 30 J=1,N
DO 30 I=1,M
IT=I+1
JT=J+1
HOLD=A(IT)
IF(DEPTH(I,J).LT.100.) GO TO 21
A(IT)=1.E06
GO TO 29
21 DNEW=DEPTH(I,J)
DOLDS=DOLD(I,J)
TMP=TP(IT,JT)
IF(DNEW.GT..2) GO TO 2746
A(IT)=TMP
DNEW=.2
GO TO 4219
2746 CONTINUE
T1=TMP
T2=TMP
T3=TMP
T4=TMP
U1=U(I,J)
U3=U(I+1,J)
V2=V(I,J)
V4=V(I,J+1)
IF(U1.GT.0.) T1=TP(IT-1,JT)
IF(U3.LT.0.) T3=TP(IT+1,JT)
IF(V2.GT.0.) T2=TP(IT,JT-1)
IF(V4.LT.0.) T4=TP(IT,JT+1)
IF(JT.EQ.2) EDGE=EDGE-V2*T2*DX
IF(IT.EQ.1) EDGE=EDGE+U3*T3*DY
IF(IT.EQ.2) EDGE=EDGE-U1*T1*DY
IF(JT.EQ.1) EDGE=EDGE+V4*T4*DX
TX=(U1*T1-U3*T3)/DX
TY=(V2*T2-V4*T4)/DY
IF(I.LT.IPI(LPI)).OR.(J.LT.JPI(LPI)).OR.(LPI.GT.NPP)) GO TO 24
IF(QIN(LPI).EQ.0.) QIN(LPI)=TMP*QPP(LPI)
TX=TX-QIN(LPI)
SORC=SORC-QIN(LPI)
LPI=LPI+1
24 IF(I.LT.IPO(LPO)).OR.(J.LT.JPO(LPO)).OR.(LPO.GT.NPP)) GO TO 23
IF(QIN(LPO).EQ.0.) QIN(LPO)=TP(IP1(LPO)+1, JPI(LPO)+1)*QPP(LPO)
TX=TX+QIN(LPO)
SORC=SORC+QIN(LPO)
LPO=LPO+1
23 IF(I.LT.IS(L)).OR.(J.LT.JS(L)).OR.(L.GT.NSEW)) GO TO 22
TX=TX+QSP(L)
SORC=SORC+QSP(L)
L=L+1
GO TO 23
22 AD=OLDA(IT,JT)*DOLDS+TA(IT,JT)*DNEW
RES=(DRESP+(1.-ZOOEFF)*DGRAZE)*FP*AD
GPLUS=GPL-DSTEP1/DNEW
GMINUS = GMIN - DSETP1 / DOLDS

A(IT) = (DOLDS * GMINUS * TMP + RES + TDT * (TX + TY)) / (DNEW * GPLUS)

IF(A(IT).LT.0.) A(IT) = 0.

DKAY = DKAY + GAMMA * (TMP * DOLDS + A(IT) * DNEW)

QRESP = QRESP + RES

QSET = QSET + DSETP1 * (TMP + A(IT))

CONTINUE

IF(JT.GT.2) TP(IT,J) = HOLD

CONTINUE

DO 31 IT = 2, M1

31 TP(IT, N1) = A(IT)

RETURN

C*****

C*****

C*****

END

SUBROUTINE PHOS(SORC, PROD, EDGE, QGROW, QBEN)

C*****

DOUBLE PRECISION SORC, PROD, EDGE, QGROW, QBEN

DOUBLE PRECISION RDT, GR, DGROMX, DRESP, DSETA, DGRAZE, DDKP1, DSETP1,

1 DBENP2

COMMON/A/M,N, TIMECT, DX, DY, TDT, IPDAY, KDYRT, TI, M1, N1, TDAY, DAYLEN

1 /B/U(69,40), V(68,41), DEPTH(68,40)

2 /C/DOLD(68,40), NSEW, IS(100), JS(100), QS(100), QSC(100), QSP(100),

3 QSP0(100), QSN(100), QSNH3(100), QSN03(100), QSOX(100),

4 /D/TA(70,42), OLDA(70,42), GRO(70,42), FC, FP, FN, DGROMX, HSATP, HSATN,

5 PHOT0, EXTL1, EXTL2, XLIGHT, XSATLI, RDT, DRESP, DSETA, DGRAZE, ZOOEFF

6 /F/TP(70,42), OLDP(70,42), DDKP1, DSETP1

7 /G/TP04(70,42), DBENP2(68,40)

8 /L/NPP, IP1(10), JPI(10), IPO(10), JPO(10), QPP(10)

DOUBLE PRECISION PRO, GR, EXCH

DIMENSION A(70), QIN(10)

C*****

DATA A/70*0., QIN/10*0./

C*****

DO 1 I = 1, M

IF((I.NE.47).AND.(I.NE.52).AND.(I.LT.57)) GO TO 1

TE = TP04(I+1,2)

TL = 2.*TE - TP04(I+1,3)

IF(TL.LT.0.) TL = .1*TE

IF(TL.GT.TE) TL = .9*TE

TP04(I+1,1) = TL

1 CONTINUE

C DO 2 I = 1, M

C TE = TP04(I+1, N+1)

C TL = 2.*TE - TP04(I+1, N)

C IF(TL.LT.0.) TL = .1*TE

C IF(TL.GT. TE) TL = .9*TE

C TP04(I+1, N+2) = TL

2 CONTINUE

DO 3 J = 1, N

IF(J.GT.9) GO TO 3

TE = TP04(M1, J+1)

TL = 2.*TE - TP04(M, J+1)
IF(TL.LT.0.) TL=.1*TE
IF(TL.GT.TE) TL=.9*TE
TP04(M+2;J+1)=TL
3 CONTINUE
C DO 4 J=1,N
C TE=TP04(2,J+1)
C TL=2.*TE-TP04(3,J+1)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.TE) TL=.9*TE
C TP04(1,J+1)=TL
C 4 CONTINUE
L=1
IF(NPP.LE.0) GO TO 6
DO 5 LP=1,NPP
5 QIN(LP)=0.
6 CONTINUE
LPI=1
LPO=1
DO 30 J=1,N
DO 30 I=1,M
IT=I+1
JT=J+1
HOLD=A(IT)
IF(DEPTH(I,J).LT.100.) GO TO 21
A(IT)=1.E06
GO TO 29
21 DNEW=DEPTH(I,J)
DOLDS=DOLD(I,J)
TMP=TP04(IT,JT)
IF(DNEW.GT.2) GO TO 2746
A(IT)=TMP
DNEW=.2
GO TO 4219
2746 CONTINUE
T1=TMP
T2=TMP
T3=TMP
T4=TMP
U1=U(I,J)
U3=U(I+1,J)
V2=V(I,J)
V4=V(I,J+1)
IF(U1.GT.0.) T1=TP04(IT-1,JT)
IF(U3.LT.0.) T3=TP04(IT+1,JT)
IF(V2.GT.0.) T2=TP04(IT,JT-1)
IF(V4.LT.0.) T4=TP04(IT,JT+1)
IF(JT.EQ.2) EDGE=EDGE-V2*T2*DX
IF(IT.EQ.1) EDGE=EDGE+U3*T3*DY
IF(IT.EQ.2) EDGE=EDGE-U1*T1*DY
IF(JT.EQ.3) EDGE=EDGE+V4*T4*DX
TX=(U1*T1-U3*T3)/DX
TY=(V2*T2-V4*T4)/DY
IF(I.NE.IPI(LPI)).OR.(J.NE.JPI(LPI)).OR.(LPI.GT.NPP)) GO TO 24
IF(QIN(LPI).EQ.0.) QIN(LPI)=TMP*QPP(LPI)
Appendix D
Ecological Model

TX=TX-QIN(LPI)
SORC=SORC-QIN(LPI)
LPI=LPI+1

24 IF((I.NE.IPO(LPO)).OR.(J.NE.JPO(LPO)).OR.(LPO.GT.NPP)) GO TO 23
    IF(QIN(LPO).EQ.0.) QIN(LPO)=TPO4(IPI(LPO)+1,JPI(LPO)+1)*QPP(LPO)
    TX=TX+QIN(LPO)
    SORC=SORC+QIN(LPO)
    LPO=LPO+1

23 IF((I.NE.IS(L)).OR.(J.NE.JS(L)).OR.(L.GT.NSEW)) GO TO 22
    TX=TX+QSP04(L)
    SORC=SORC+QSP04(L)
    L=L+1
    GO TO 23

22 AD=OLDA(IT,JT)*DOLDS+TA(IT,JT)*DNEW
    PRO=DDKP1*(OLDP(IT,JT)*DOLDS+TP(IT,JT)*DNEW)
    GR=Gro(IT,JT)*FP*AD
    EXCH=PRO-GR+DBNP2(I,J)
    A(IT)=(DOLDS*TMP+EXCH+TDT*(TX+TY))/DNEW
    IF(A(IT).LT.0.) A(IT)=O.
    QGROW=QGROW+GR
    PROD=PROD+PRO
    QBEN=QBEN+DBNP2(I,J)

4219 CONTINUE
29 IF(JT.GT.2) TPO4(IT,J)=HOLD
30 CONTINUE
DO 31 IT=2,M1
31 TPO4(IT,N1)=A(IT)
RETURN

C*****
C*****
C*****
C*****
SUBROUTINE ORGN(SORC,DKAY,EDGE,QRESP,QSET)

C*****
DOUBLE PRECISION SORC,DKAY,EDGE,QRESP,QSET
DOUBLE PRECISION RDT,GRO,DGROMX,DRESP,DATA,DGRAZE,DDKN1,DSETN1
COMMON/A/M,N,TIMECT,DX,DY,TDT,IPRDAY,KDYPRT,MI,M1,N1,TDAY,DAYLEN
  /B/U(69,40),V(68,41),DEPTH(68,40)
  /C/DOLD(68,40),NSEE,IS(100),JS(100),QS(100),QSC(100),QSP(100),
  /D/TPO4(100),QSN(100),QSNH3(100),QSN03(100),QSOX(100)
  /H/TN(70,42),OLDN(70,42),DDKN1,DSETN1
  /D/TA(70,42),OLDA(70,42),GRO(70,42),FC,FP,FN,DGROMX,HSATP,HSATN,
  /PHOTO,EXTL1,EXTL2,XLIGTH,XSATLI,RDT,DRESP,DSETA,DGRAZE,ZO EFF
  /L/NPP,IP(10),JPI(10),IPO(10),JPO(10),QPP(10)
DOUBLE PRECISION GAMMA,GPL,GMIN,GPLUS,GMINUS,RES
DIMENSION A(70),QIN(10)

C*****
DATA A/70*0./,QIN/10*0./
C*****
GAMMA=DDKN1
GPL=1.+GAMMA
GMIN=1.-GAMMA
C
DO 1 I=1,M
C
IF((I.NE.47).AND.(I.NE.52).AND.(I.LT.57)) GO TO 1
Vol II

C TE=TN(I+1,2)
C TL=2.*TE-TN(I+1,3)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.0.) TL=.9*TE
C TN(I+1,1)=TL
C 1 CONTINUE
C DO 2 I=1,M
C TE=TN(I+1,N+1)
C TL=2.*TE-TN(I+1,N)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.0.) TL=.9*TE
C TN(I+1,N+1)=TL
C 2 CONTINUE
C DO 3 J=1,N
C IF(J.GT.9) GO TO 3
C TE=TN(M1,J+1)
C TL=2.*TE-TN(M,J+1)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.0.) TL=.9*TE
C TN(M+2,J+1)=TL
C 3 CONTINUE
C DO 4 J=1,N
C TE=TN(2,J+1)
C TL=2.*TE-TN(3,J+1)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.0.) TL=.9*TE
C TN(1,J+1)=TL
C 4 CONTINUE
C L=1
C IF(NPP.LE.0) GO TO 6
C DO 5 LP=1,NPP
C 5 QIN(LP)=O.
C 6 CONTINUE
C LPI=1
C LPO=1
C DO 30 J=1,N
C DO 30 I=1,M
C IT=I+1
C JT=J+1
C HOLD=A(IT)
C IF(DEPTH(I,J).LT.100.) GO TO 21
C A(IT)=1.E06
C GO TO 29
C 21 DNEW=DEPTH(I,J)
C DOLDS=DOLD(I,J)
C TMP=TN(IT,JT)
C IF(DNEW.GT.2) GO TO 2746
C A(IT)=TMP
C DNEW=.2
C GO TO *219
C 2746 CONTINUE
C T1=TMP
C T2=TMP
C T3=TMP

Appendix D
Ecological Model
T4 = TMP
U1 = U(I, J)
U3 = U(I+1, J)
V2 = V(I, J)
V4 = V(I, J+1)

IF(U1.GT.0.)  T1 = TN(IT-1, JT)
IF(U3.LT.0.)  T3 = TN(IT+1, JT)
IF(V2.GT.0.)  T2 = TN(IT, JT-1)
IF(V4.LT.0.)  T4 = TN(IT, JT+1)

IF(JT.EQ.2)  EDGE = EDGE - V2*T2*DX
IF(IT.EQ.M1) EDGE = EDGE + U3*T3*DY
IF(IT.EQ.1)  EDGE = EDGE - U1*T1*DY
IF(JT.EQ.N1) EDGE = EDGE + V4*T4*DX

TX = (U1*T1 - U3*T3)/DX
TY = (V2*T2 - V4*T4)/DY

IF(I.EQ.IPI(LPI)) OR (J.EQ.JPI(LPI)) OR (LP1.GT.NPP)) GO TO 24
IF(QIN(LPI).EQ.0.)  QIN(LPI) = TMP*QPP(LPI)
TX = TX - QIN(LPI)
SORC = SORC - QIN(LPI)
LPI = LPI + 1

IF(I.EQ.IPO(LPO)) OR (J.EQ.JPO(LPO)) OR (LPO.GT.NPP) GO TO 23
IF(QIN(LPO).EQ.0.)  QIN(LPO) = TN(IP1(LPO)+1, JPI(LPO)+1)*QPP(LPO)
TX = TX + QIN(LPO)
SORC = SORC + QIN(LPO)
LPO = LPO + 1

IF(I.EQ.IS(L)) OR (J.EQ.JS(L)) OR (L.GT.NSEW) GO TO 22
TX = TX + QSN(L)
SORC = SORC + QSN(L)
L = L + 1
GO TO 23

CONTINUE

GPLUS = GPL + DSETN1/DNEW
GMINUS = GMIN - DSETN1/DOLDS
AD = OLDA(IT, JT)*DOLDS + TA(IT, JT)*DNEW
RES = (DRESP + (1 - Z0EFF)*DGRAZE)*FN*AD
A(IT) = (DOLDS*GMINUS*TMP + RES + TDT*(TX + TY))/(DNEW*GPLUS)
IF(A(IT).LT.0.)  A(IT) = 0.

DKAY = DKAY + (TMP*DOLDS + A(IT)*DNEW)*GAMMA
QRESP = QRESP + RES
QSET = QSET + DSETN1*(TMP + A(IT))

CONTINUE

IF(JT.GT.2)  TN(IT, J) = HOLD

CONTINUE
DO 31 IT = 2, M1
31  TN(IT, N1) = A(IT)
RETURN

SUBROUTINE AMM0N(SORC, DKAY, PROD, EDGE, QGROW, QSET, QBEN)
DOUBLE PRECISION SORC, DKAY, PROD, EDGE, QGROW, QSET, QBEN
DOUBLE PRECISION RDT, GRO, DGROMX, DRESP, DSETA, DGRAZE, DDKN1, DSETN1,
DOUBLE PRECISION GAMMA,GPLUS,GMINUS,PRO,GR,EXCH
DIMENSION A(70),QIN(10)
DATA A/70*0.I,QIN/10*0.1
DO 1 I=1,M
  IF((I.NE.47).AND.(I.NE.52).AND.(I.LT.57)) GO TO 1
  TE=TNH3(I+1,2)
  TL=2.*TE-TNH3(I+1,3)
  IF(TL.LT.0.) TL=1.*TE
  IF(TL.GT.TE) TL=9.*TE
  TNH3(I+1,1)=TL
1 CONTINUE
C DO 2 I=1,M
  TE=TNH3(I+1,N+1)
  TL=2.*TE-TNH3(I+1,N)
  IF(TL.LT.0.) TL=1.*TE
  IF(TL.GT.TE) TL=9.*TE
  TNH3(I+1,N+2)=TL
2 CONTINUE
C DO 3 J=1,N
  IF(J.GT.9) GO TO 3
  TE=TNH3(M1,J+1)
  TL=2.*TE-TNH3(M,J+1)
  IF(TL.LT.0.) TL=1.*TE
  IF(TL.GT.TE) TL=9.*TE
  TNH3(M+2,J+1)=TL
3 CONTINUE
C DO 4 J=1,N
  TE=TNH3(2,J+1)
  TL=2.*TE-TNH3(3,J+1)
  IF(TL.LT.0.) TL=1.*TE
  IF(TL.GT.TE) TL=9.*TE
  TNH3(1,J+1)=TL
4 CONTINUE
L=1
IF(NPP.LE.0) GO TO 6
DO 5 LP=1,NPP
  QIN(LP)=0.
5 CONTINUE
LPI=1
LPO=1
DO 30 J=1,N
DO 30 I=1,M
IT=I+1
JT=J+1
HOLD=A(IT)
IF(DEPTH(I,J).LT.100.) GO TO 21
A(IT)=1.E06
GO TO 29
21 DNEW=DEPTH(I,J)
DOLDS=DOLD(I,J)
TMP=TNH3(IT,JT)
IF(DNEW.GT.2) GO TO 2746
A(IT)=TMP
DNEW=.2
GO TO '4219
2746 CONTINUE
T1=TMP
T2=TMP
T3=TMP
T4=TMP
U1=U(I,J)
U3=U(I+1,J)
V2=V(I,J)
V4=V(I,J+1)
IF(U1.GT.0.) T1=TNH3(IT-1,JT)
IF(U3.LT.0.) T3=TNH3(IT+1,JT)
IF(V2.GT.0.) T2=TNH3(IT,JT-1)
IF(V4.LT.0.) T4=TNH3(IT,JT+1)
IF(IT.EQ.M1) EDGE=EDGE-U1*T1*DY
IF(IT.EQ.2) EDGE=EDGE+U3*T3*DX
TX=(U1*T1-U3*T3)/DX
TY=(V2*T2-V4*T4)/DY
IF((I.NE.IPO(LPI)).OR.(J.NE.JPI(LPI)).OR.(LPI.GT.NPP)) GO TO 24
IF(QIN(LPI).EQ.0.) QIN(LPI)=TMP*QPP(LPI)
TX=TX-QIN(LPI)
SORC=SORC-QIN(LPI)
LPI=LPI+1
24 IF((I.NE.IPO(LPO)).OR.(J.NE.JPO(LPO)).OR.(LPO.GT.NPP)) GO TO 23
IF(QIN(LPO).EQ.0.) QIN(LPO)=TNH3(IPO(LPO)+1,JPO(LPO)+1)*QPP(LPO)
TX=TX+QIN(LPO)
SORC=SORC+QIN(LPO)
LPO=LPO+1
23 IF((I.NE.IS(L)).OR.(J.NE.JS(L)).OR.(L.GT.NSEW)) GO TO 22
TX=TX+QSNH3(L)
SORC=SORC+QSNH3(L)
L=L+1
GO TO 23
22 CONTINUE
GAMMA=DDKN2*F2(IT,JT)
GPLUS=1.+GAMMA+DSETN2/DNEW
QMINUS=1.-GAMMA-DSETN2/DOLDS
PRO=DDKN1*(OLDN(IT,JT)*DOLDS+TN(IT,JT)*DNEW)
PREF(IT,JT)=PREFER(IT,JT)
\[GR = \text{PREF}(IT, JT) \times \text{GRO}(IT, JT) \times FN \times (\text{OLDA}(IT, JT) \times DOLDS + \text{TA}(IT, JT) \times DNEW)\]

\[\text{EXCH} = \text{PRO} - \text{GR} + \text{DBENN2}(I, J)\]

\[A(IT) = (DOLDS \times \text{GMINUS} \times \text{TMP} + \text{EXCH} + \text{TDT} \times (\text{TX + TY}) \big/ \big( \text{DNEW} \times \text{GPLUS} \big)\]

\[\text{IF}(A(IT) \lt 0.) \text{ A(IT)} = 0.\]

\[\text{DKAY} = \text{DKAY} \times (\text{TMP} \times \text{DOLDS} + A(IT) \times \text{DNEW}) \times \text{GAMMA}\]

\[\text{QSET} = \text{QSET} + \text{DSETN2} \times (\text{TMP} + A(IT))\]

\[\text{QBEN} = \text{QBEN} + \text{DBENN2}(I, J)\]

\[\text{QGROW} = \text{QGROW} + \text{GR}\]

\[\text{PROD} = \text{PROD} + \text{PRO}\]

\[\text{CONTINUE}\]

\[\text{IF}(\text{JT} \gt 2) \text{ TNH3}(IT, J) = \text{HOLD}\]

\[\text{CONTINUE}\]

\[\text{DO} 31 \text{ IT} = 2, M1\]

\[31 \text{ TNH3}(IT, N1) = A(IT)\]

\[\text{RETURN}\]

C*****

C*****

FUNCTION F2(I, J)

C*****

DOUBLE PRECISION DBENOX

COMMON/K/TOX(70,42), WINDF, WINDCO, TCOREA, TEMP, OXGROW, OXRESP, OXNH3,
\[1 \text{ DBENOX}(68,40)\]

C*****

\[F2 = \text{TOX}(I, J) / (\text{TOX}(I, J) + 0.50)\]

RETURN

END

C*****

FUNCTION PREFER(I, J)

C*****

DOUBLE PRECISION DDKN2, DSETN2, DBENN2, DBENN3

COMMON/I/TNH3(70,42), OLDNH3(70,42), PREF(70,42), DDKN2, PRF, DSETN2,
\[1 \text{ DBENN2}(68,40)\]

\[2 / J/TNO3(70,42), DBENN3(68,40)\]

C*****

\[\text{PRF1} = \text{PRF}\]

\[\text{PRF2} = 1. - \text{PRF}\]

\[\text{CN2} = \text{TNH3}(I, J)\]

\[\text{CN3} = \text{TNO3}(I, J)\]

\[PREFER = 0.\]

\[\text{IF}((\text{CN2} \leq 0.) \text{ .AND. (CN3} \leq 0.)) \text{ RETURN}\]

\[PREFER = (\text{PRF1} \times \text{CN2}) / (\text{PRF1} \times \text{CN2} + \text{PRF2} \times \text{CN3})\]

RETURN

END

C*****

C*****

SUBROUTINE TRATE(SORC, PROD, EDGE, QGROW, QBEN)

C*****

DOUBLE PRECISION SORC, PROD, EDGE, QGROW, QBEN
DOUBLE PRECISION RDT, GRO, DGROMX, DRESP, DSETA, DGRAZE, DDKN2, DSETN2,
DBENN2, DBENN3
COMMON /A/M,N,TIMECT, DX, DY, TDT, IPRDAY, KDYPRT, TI, M1, N1, TDAY, DAYLEN
1 /B/U(69,40), V(68,41), DEPTH(68,40)
2 /C/DOLD(68,40), NSEW, IS(100), JS(100), QS(100), QSC(100), QSP(100),
3 QSP04(100), QSN(100), QSNH3(100), QSNF3(100), QSOX(100)
4 /T/TN3(70,42), OLDNH3(70,42), PREF(70,42), DDKN2, PRF, DSETN2,
5 DBENN2(68,40)
6 /J/TN03(70,42), DBENN3(68,40)
7 /D/TA(70,42), OLDA(70,42), GRO(70,42), FC, FP, FN, DGROMX, HSATP, HSATN,
8 PHOTO, EXTL1, EXTL2, XLIGHT, XSATL, RDT, DRESP, DSETA, DGRAZE, ZOOEFF
9 /L/NPP, IP1(10), JPI(10), IPO(10), JPO(10), QPP(10)
DOUBLE PRECISION PRO, GR, EXCH
DIMENSION A(70), QIN(10)

C*****
DATA A/70*0., QIN/10*0. /
C*****

DO 1 I=1,M
IF(.NOT.(I.NE.47).AND.(I.NE.52).AND.(I.LT.57)) GO TO 1
TE=TN03(I+1,2)
TL=2.0*TE-TN03(I+1,3)
IF(TL.LT.0.0) TL=-1.0*TE
IF(TL.GT.TE) TL=9.0*TE
TN03(I+1,1)=TL
1 CONTINUE

C DO 2 I=1,M
C TE=TN03(I+1,N+1)
C TL=2.0*TE-TN03(I+1,N)
C IF(TL.LT.0.0) TL=-1.0*TE
C IF(TL.GT.TE) TL=9.0*TE
C TN03(I+1,N+2)=TL
2 CONTINUE

C DO 3 J=1,N
C IF(J.GT.9) GO TO 3
C TE=TN03(M1, J+1)
C TL=2.0*TE-TN03(M, J+1)
C IF(TL.LT.0.0) TL=-1.0*TE
C IF(TL.GT.TE) TL=9.0*TE
C TN03(M+2, J+1)=TL
3 CONTINUE

C DO 4 J=1,N
C TE=TN03(2, J+1)
C TL=2.0*TE-TN03(3, J+1)
C IF(TL.LT.0.0) TL=-1.0*TE
C IF(TL.GT.TE) TL=9.0*TE
C TN03(1, J+1)=TL
4 CONTINUE

L=1
IF(NPP.LE.0) GO TO 6
DO 5 LP=1,NPP
5 QIN(LP)=0.
6 CONTINUE
LP=1
DO 30 J=1,N
DO 30 I=1,M
IT=I+1
JT=J+1
HOLD=A(IT)
IF(DEPTH(I,J).LT.100.) GO TO 21
A(IT)=1.E06
GO TO 29
21 DNEW=DEPTH(I,J)
DOLDS=DOLD(I,J)
TMP=TN03(IT,JT)
IF(DNEW.GT.2) GO TO 2746
A(IT)=TMP
DNEW=.2
GO TO 4219
2746 CONTINUE
T1=TMP
T2=TMP
T3=TMP
T4=TMP
U1=U(I,J)
U3=U(I+1,J)
V2=V(I,J)
V4=V(I,J+1)
IF(U1.GT.0.) T1=TN03(IT-1,JT)
IF(U3.LT.0.) T3=TN03(IT+1,JT)
IF(V2.GT.0.) T2=TN03(IT,JT-1)
IF(V4.LT.0.) T4=TN03(IT,JT+1)
IF(JT.EQ.2) EDGE=EDGE-V2*T2*DX
IF(IT.EQ.1) EDGE=EDGE-U1*T1*DY
IF(IT.EQ.2) EDGE=EDGE-U1*T1*DY
IF(IT.EQ.1) EDGE=EDGE+V4*T4*DX
T=U1-T1-U3*T3/DX
TY=V2*V4*V4*V4/(DY)
IF(I.EQ.IPI(LPI)) OR (J.EQ.JPI(LPI)) OR (LPI.GT.NPP) GO TO 24
IF(QIN(LPI).EQ.0.) QIN(LPI)=TMP*QPP(LPI)
TX=TX+QIN(LPI)
SORC=SORC+QIN(LPI)
LPI=LPI+1
24 IF(I.EQ.IPO(LPO)) OR (J.EQ.JPO(LPO)) OR (LPO.GT.NPP) GO TO 23
IF(QIN(LPO).EQ.0.) QIN(LPO)=TN03(IPI(LPO)+1,JPI(LPO)+1)*QPP(LPO)
TX=TX+QIN(LPO)
SORC=SORC+QIN(LPO)
LPO=LPO+1
23 IF(I.EQ.IS(L)) OR (J.EQ.JS(L)) OR (L.GT.NSEW) GO TO 22
TX=TX+QSN03(L)
SORC=SORC+QSN03(L)
L=L+1
GO TO 23
22 PRO=F2(IT,JT)*DDKN2*(OLDNH3(IT,JT)*DOLDS+TNH3(IT,JT)*DNEW)
GR=(1.-PREP(IT,JT))*GRO(IT,JT)*FN*(OLDA(IT,JT)*DOLDS+TA(IT,JT))
1*DNEW
EXCH=PRO*GR+DBENN3(I,J)
A(IT)=(DOLDS*TMP+EXCH+TDT*(TX+TY))/DNEW
IF(A(IT).LT.0.) A(IT)=0.
QGROW=QGROW+GR
QBEN=QBEN+DBENN3(I,J)
PROD=PROD+PRO
CONTINUE
29 IF(JT.GT.2) TNO3(IT,J)=HOLD
30 CONTINUE
31 TNO3(IT,N1)=A(IT)
RETURN
C*****
C*****
C*****
END

SUBROUTINE OXYGEN(SATDOX,SORC,QOBOD,REAR,EDGE,QO2,QGROW,QRESP,
1 QBEN,QONH3)

DOUBLE PRECISION SORC,QOBOD,REAR,EDGE,QO2,QGROW,QRESP,QBEN,QONH3
DOUBLE PRECISION RDT,GRO,DGRMX,DRESP,DSETA,DGRAZE,DKKC,DSETC,
1 DDKN2,DSETN2,DDENOX
COMMON/A/M,N,TIMECT,DX,DY,TDT,IPRDAY,KDYPRT,IT,M1,N1,TDAY,DAYLEN
1 /B/U(69,40),V(68,41),DEPTH(68,40)
2 /C/DOLD(68,40),NSEW,IS(100),JS(100),QS(100),QSC(100),QSP(100),
3 QSP04(100),QSN(100),QSNH3(100),QSN03(100),QSOX(100)
4 /D/T(70,42),OLDA(70,42),GRO(70,42),FC,FP,FN,DGRMX,HSATP,HSATN,
5 PHOTO,EXTL1,EXTL2,XLIGHT,XSATL1,RT,DRESP,DSETA,DGRAZE,ZOOEFF
6 /E/TC(70,42),OLDC(70,42),DKKC,DSETC,OX
7 /I/TNH3(70,42),OLDNH3(70,42),PREF(70,42),DDKN2,PRF,DSETN2,
8 DBENOX(68,40)
9 /K/TOX(70,42),WINDF,WINDCO,TCOREA,TEMP,OXGROW,OXRESP,OXNH3,
1 DBENOX(68,40)
2 /L/NPP,IP1(10),IP2(10),JPO(10),JPP(10),QPP(10)
3 DOUBLE PRECISION REAER,REAERA,GAMMA,GPLUS,GMINUS,RES,REAR,
1 OBOD,ONH3,GR,REA,EXCH
DIMENSION A(70),QIN(10)

DATA A/70*0.,QIN/10*0./

DO 1 I=1,M
1 TE=TOX(I+1,2)
C TL=2.*TE-TOX(I+1,3)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.TE) TL=.9*TE
C TOX(I+1,1)=TE
C IF(V(I,1).LT.0.) TOX(I+1,1)=(SATDOX+TOX(I+1,2))*5
2 CONTINUE
C DO 2 I=1,M
C TE=TOX(I+1,N+1)
C TL=2.*TE-TOX(I+1,N)
C IF(TL.LT.0.) TL=.1*TE
C IF(TL.GT.TE) TL=.9*TE
C TOX(I+1,N+2)=TE
C 2 CONTINUE
DO 3 J=1,N
IF(J.GT.9) GO TO 3
C
TE=TOX(M1,J+1)
C
TL=2.*TE-TOX(M,J+1)
C
IF(TL.LT.0.) TL=.1*TE
C
IF(TL.GT.TE) TL=.9*TE
C
TOX(M+2,J+1)=TE
 IF(U(M+1,J).LT.0.) TOX(M+2,J+1)=(SATDOX+TOX(M+1,J+1))*5
3 CONTINUE
C
DO 4 J=1,N
C
TE=TOX(2,J+1)
C
TL=2.*TE-TOX(3,J+1)
C
IF(TL.LT.0.) TL=.1*TE
C
IF(TL.GT.TE) TL=.9*TE
C
TOX(1;J+1)=TE
4 CONTINUE
L=1
IF(NPP.LE.0) GO TO 6
DO 5 LP=1,NPP
QIN(LP)=0.
6 CONTINUE
LPI=1
LP0=1
DO 30 J=1,N
DO 30 I=1,M
IT=I+1
JT=J+1
HOLD=A(IT)
IF(DEPTH(I,J).LT.100.) GO TO 21
A(IT)=1.E06
GO TO 29
21 DNEW=DEPTH(I,J)
DOLDS=DOLD(I,J)
TMP=TOX(IT,JT)
IF(DNEW.GT.2) GO TO 2746
A(IT)=TMP
DNEW=.2
GO TO 4219
2746 CONTINUE
T1=TMP
T2=TMP
T3=TMP
T4=TMP
U1=U(I,J)
U3=U(I+1,J)
V2=V(I,J)
V4=V(I,J+1)
IF(U1.GT.0.) T1=TOX(IT-1,JT)
IF(U3.LT.0.) T3=TOX(IT+1,JT)
IF(V2.GT.0.) T2=TOX(IT,JT-1)
IF(V4.LT.0.) T4=TOX(IT,JT+1)
IF(JT.EQ.2) EDGE=EDGE-V2*T2*DX
IF(IT.EQ.M1) EDGE=EDGE+U3*T3*DY
IF(IT.EQ.2) EDGE=EDGE-U1*T1*DY
IF(JT.EQ.N1) EDGE=EDGE+V4*T4*DX
TX=(U1*T1-U3*T3)/DX
TY=(V2*T2-V4*T4)/DY

IF((I,NE.IPI(LPI)).OR.(J,NE.JPI(LPI)).OR.(LPI.GT.NPP)) GO TO 24
IF(QIN(LPI).EQ.0.) QIN(LPI)=TMP*QPP(LPI)
TX=TX-QIN(LPI)
SORC=SORC-QIN(LPI)
LPI=LPI+1

24 IF((I,NE.IPO(LPO)).OR.(J,NE.JPO(LPO)).OR.(LPO.GT.NPP)) GO TO 23
IF(QIN(LPO).EQ.0.) QIN(LPO)=TOX(IPI(LPO)+1,JPI(LPO)+1)*QPP(LPO)
TX=TX-QIN(LPO)
SORC=SORC+QIN(LPO)
LPO=LPO+1

23 IF((I,NE.IS(L)).OR.(J,NE.JS(L)).OR.(L.GT.NSEW)) GO TO 22
TX=TX-QSOX(L)
SORC=SORC-QSOX(L)
L=L+1
GO TO 23

22 AD=OLDA(IT,JT)*DOLDS+TA(IT,JT)*DNEW
DEP=(DOLDS+DNEW)/2.
UVEL=((U1+U3)/2.)/DEP
VEL=((V2+V4)/2.)/DEP
VEL=SQR(UVEL*UVEL+VEL*VEL)
REAER=REAERA(VEL,DEP,WINDF,WINDCO)*TCOREA**(TEMP-20.)*DTD/2.
GAMMA=REAER

CPLUS=1.+GAMMA
CMINUS=1.-GAMMA
RES=OXRESP*DRESP*AD
OBOD=F1(IT,JT)*DDKC*(OLDC(IT,JT)*DOLDS+TC(IT,JT)*DNEW)
ONH3=F2(IT,JT)*OXNH3*DDKN2*(OLDNH3(IT,JT)*DOLDS+TNH3(IT,JT)*DNEW)
GR=OXGROW*GRO(IT,JT)*AD
REA=REAER*SATDOX*(DOLDS+DNEW)
EXCH=GR+REA-RES-Dbenhox(I,J)-ONH3-OBOD
A(IT)=((DOLDS*CMINUS*TMP+EXCH+TDT*(TX+TY))/(DNEW*GPLUS)
IF(A(IT).LT.0.) A(IT)=0.
REAR=-(TMP*DOLDS+A(IT)*DNEW)*GAMMA+REA
IF(REAR.LT.0.) GO TO 40
QREAR=QREAR+REAR
GO TO 41

40 Q02=Q02-REAR
41 CONTINUE
QGROW=QGROW+CR
QRESP=QRESP+RES
QOBEN=QOBEN+DBENOX(I,J)
QOBOD=QOBOD+OBOD
QONH3=QONH3+ONH3

4219 CONTINUE
29 IF(JT.GT.2) TOX(IT,J)=HOLD
30 CONTINUE
DO 31 IT=2,M1
31 TOX(IT,N1)=A(IT)
RETURN
END
C****
FUNCTION REAERA(VEL, DEP, WINDF, WINDCO)
DOUBLE PRECISION REAERA
REAERA = 5.0 * (11.6101 * (VEL ** 0.969) / (DEP ** 1.673)) + (WINDCO * WINDF / DEP)
REAERA2 = 1.0 * (7.6 * VEL / (DEP ** 1.333)) + (WINDCO * WINDF / DEP)
RETURN
END

SUBROUTINE MAPALG(NNAME, T, CHL, QINIT, QGRAZ, QRESP, QGROW, QSET, QGRAZ2, QEDGE2, QRESP2, QGROW2, QSET2, QRGRID2, KEY)
DOUBLE PRECISION QGRAZ, QEDGE, QGRAZ2, QEDGE2, QINIT, QRESP, QRESP2, QGROW, QGROW2, QSET, QSET2, QRGRID2
DOUBLE PRECISION CONS1, CONS2, QGRAZL, QGRIDL, DTSUM, QRGRID1, QEDGE1
COMMON /A/M, N, TIMECT, DX, DY, TDT, IPRDAY, KDPRT, TI, M1, N1, TDAY, DAYLEN
1 /U(69, 40), V(68, 41), DEPTH(68, 40)
2 /C/DOLD(68, 40), NSEW, IS(100), JS(100), QS(100), QSC(100), QSP(100),
3 QSP04(100), QSN(100), QSN03(100), QSOX(100),
4 /M/CONS1, CONS2, BODU5
DIMENSION TNT(102), T(70, 42)
REAL*8 STAR, BLANK, STARS(102)
REAL*4 CNAME(4, 2)
INTEGER*4 ITI, IDAY, ITPH, ITPM
DATA STAR, BLANK'/**********', '/
DATA CNAME/'ALGA', 'E BI', 'OMAS', 'S ', 'CHLO', 'ROPH', 'YLL ', 1 'A '/

200 FORMAT(48X, 'CONCENTRATIONS (MICROG/L)', '//)
201 FORMAT(50X, 'CONCENTRATIONS (MG/L)', '//)
202 FORMAT(1H1, 22X, 'TIME = ', 'I3, ' DAYS, ', I3, ' HOURS, ', I3, ' MINUTES', 1 10X, 4A4, '//)
203 FORMAT(' I J', 22(1X, I2, 2X), '//)
204 FORMAT(I3, 22F5.2)
205 FORMAT(2X, 'CHANGE', 9X, 8F10.0)
208 FORMAT(50X, 'TOTAL WEIGHT (POUNDS)', '//)
209 FORMAT(2X, 'TOTAL INITIALLY IN THE GRID ', F10.0, '//)
210 FORMAT(I3, 22F5.0)
1010 FORMAT(3X, 22A5)

ITI = TI + .9
IDAY = ITI / 1440
ITPH = ITI / 60 - IDAY * 24
ITPM = ITI - ITPH * 60 - IDAY * 1440
NN = NNAME
**Appendix D**

**Ecological Model**

```
CH=1.
IF(NNAME.EQ.1) GO TO 80
NN=2

80 CONTINUE
IF(KEY.EQ.5) GO TO 89
DO 304 NSTRT=1,N,22
NSTOP=NSTRT+21
IF(NSTOP.GT.N) NSTOP=N
WRITE(6,202) IDAY,ITPH,ITPM,(CNAME(L,NN),L=1,4)
IF(NN.EQ.1) WRITE(6,201)
IF(NN.EQ.2) WRITE(6,200)
WRITE(6,203) (J, J=NSTRT,NSTOP)
DO 303 I=1,M
DO 301 J=NSTRT,NSTOP
STARS(J)=BLANK
TNT(J)=T(I+1,J+1)*CH
IF(T(I+1,J+1).GT.100.) STARS(J)=STAR
301 CONTINUE
WRITE(6,1010) (STARS(J),J-NSTRT,NSTOP)
IF(NN.EQ.1) WRITE(6,204) I,(TNT(J),J=NSTRT,NSTOP)
IF(NN.EQ.2) WRITE(6,210) I,(TNT(J),J=NSTRT,NSTOP)
WRITE(6,1010) (STARS(J),J=NSTRT,NSTOP)
304 CONTINUE
89 IF(NN.EQ.2) RETURN
WRITE(6,202) IDAY,ITPH,ITPM,(CNAME(L,NN),L=1,4)
WRITE(6,208)
QGRAZ1=QGRAZ*CONS2
QRESP1=QRESP*CONS2
QGROW1=QGROW*CONS2
QSET1=QSET*CONS2
QGRAZ=0.
QRESP=0.
QGROW=0.
QSET=0.
QGRAZ2=QGRAZ1+QGRAZ2
QRESP2=QRESP1+QRESP2
QGROW2=QGROW1+QGROW2
QSET2=QSET1+QSET2
QEDGE1=QEDGE*CONS1*TDT
QEDGE=0.
QEDGE2=QEDGE2+QEDGE1
QGRIDS=QGRID2
DTSUM=0.
DO 90 J=1,N
DO 90 I=1,M
IF(DEPTH(I,J).GT.100.) GO TO 90
DTSUM=DTSUM+DOLD(I,J)*T(I+1,J+1)
90 CONTINUE
QGRID2=DTSUM*CONS2
QGRID1=QGRID2-QGRIDS
TOTOU1=QGRAZ1+QEDGE1+QRESP1+QSET1
```
TOTOU2 = QGRAZ2 + QEDGE2 + QRESP2 + QSET2
TOTIN1 = QGROW1
TOTIN2 = QGROW2
WRITE(6, 209) QINIT
WRITE(6, 207) QGRID2, QGROW2, QRESP2, QSET2, QGRAZ2, QEDGE2, TOTIN2,
1 TOTOU2
WRITE(6, 205) QGRID1, QGROW1, QRESP1, QSET1, QGRAZ1, QEDGE1, TOTIN1,
1 TOTOU1
RETURN
END

C*****
C*****
C*****
SUBROUTINE MAPCON(NNAME, T, QINIT, QSORC, QDKAY, QPROD, QEDGE, QRESP,
1 QBEN, QGROW, QSET, QSORC2, QDKAY2, QPROD2, QEDGE2, QRESP2, QBEN2, QGROW2,
2 QSET2, QGRID2, KEY)
C*****
DOUBLE PRECISION QSORC, QDKAY, QPROD, QEDGE, QRESP,
1 QGRAZ, QGROW, QSET, QGRID2
DOUBLE PRECISION CONS1, CONS2, QDKAY1, QPROD1, QGRID1, DTSUM,
1 QGRID1, QGROW1, QSET1, TOTOU1, TOTOU2, TOTIN1, TOTIN2, QRESP1, QBEN1, QGROW1,
2 QSET1
COMMON/A/M, N, TIMECT, DX, DY, TDIR, KPDAY, TI, M1, N1, TDAY, DAYLEN
/BU/U(69, 40), V(68, 41), DEPTH(68, 40),
/C/DOLD(68, 40), NSEW, IS(100), JS(100), QS(100), QSC(100), QSP(100),
3 QSP04(100), QSN(100), QSNH3(100), QSN03(100), QSOX(100)
/M/CONS1, CONS2, BODU5
DIMENSION TNT(102), T(70, 42)
REAL*8 STAR,BLANK,STARS(102)
REAL*4 CNAME(9, 7)
INTEGER*4 ITI, IDAY, ITPH, ITPM
C*****
DATA STAR, BLANK/ '********',
1 'CARB', 'ONAC', 'EOUS', 'BOD', 'BC', 'DC', '', '',
2 'ORG', 'NIC', 'PHOS', 'PHOR', 'US A', 'S P', 'ORG', 'P-P',
3 'ORTH', 'OPOH', 'SPHA', 'TE A', 'S P', 'PO4', 'P-P',
4 'ORCA', 'NIC', 'NITR', 'OGEN', 'AS', 'N (O)', 'RG N', '-N',
5 'AMNO', 'NIA', 'AS N', '(NH', '3-N)', '', '',
6 'NITR', 'ATE', 'AS N', '(NO', '3-N)', '', '',
7 'FIVE', '-DAY', 'BOD', 'BO', 'D5', '',
/C*****
FORMAT(50X, 'CONCENTRATIONS (MG/L')
201 FORMAT(50X, 'TIME = ', I3, ' DAYS', I3, ' HOURS', I3, ' MINUTES',
1 10X, 9A4)
202 FORMAT(1H1, 22X, 'TOTAL INITIALLY IN THE GRID', F10.0)
203 FORMAT(1H1, 22X, 'TOTAL WEIGHT (POUNDS)', F10.0)
1010 FORMAT(3X,22A5)
C*****
ITI=TI+.9
IDAY=ITI/1440
ITPH=ITI/60-IDAY*24
ITPM=ITI-ITPH/60-IDAY*1440
NN=NNAME-1
IF(NN.EQ.8) NN=7
CH=1.
IF(NN.EQ.7) CH=1./BODU5
IF(KEI.EQ.5) GO TO 89
DO 304 NSTRT=1,N,22
NSTOP=NSTRT+21
IF(NSTOP.GT.N) NSTOP=N
WRITE(6,202) IDAY,ITPH,ITPM,(CNAME(L,NN),L=1,9)
WRITE(6,201)
WRITE(6,203) (J,J=NSTRT,NSTOP)
DO 303 I=1,M
DO 301 J=NSTRT,NSTOP
STARS(J)=BLANK
TNT(J)=T(I+1,J+1)*CH
IF(T(I+1,J+1).GT.1000.) STARS(J)=STAR
301 CONTINUE
WRITE(6,1010) (STARS(J),J=NSTRT,NSTOP)
WRITE(6,204) I,(TNT(J),J=NSTRT,NSTOP)
WRITE(6,1010) (STARS(J),J=NSTRT,NSTOP)
303 CONTINUE
WRITE(6,203) (J,J=NSTRT,NSTOP)
304 CONTINUE
89 IF(NN.EQ.7) RETURN
WRITE(6,202) IDAY,ITPH,ITPM,(CNAME(L,NN),L=1,9)
WRITE(6,208)
QSORC1=QSORC*CONS2*TDT
QDKAY1=QDKAY*CONS2
QPROD1=QPROD*CONS2
QRESP1=QRESP*CONS2
QBEN1=QBEN*CONS2
QGROW1=QGROW*CONS2
QSET1=QSET*CONS2
QSORC=0.
QDKAY=0.
QPROD=0.
QRESP=0.
QBEN=0.
QGROW=0.
QSET=0.
QSORC2=QSORC1+QSORC2
QDKAY2=QDKAY1+QDKAY2
QPROD2=QPROD1+QPROD2
QRESP2=QRESP1+QRESP2
QBEN2=QBEN1+QBEN2
QGROW2=QGROW1+QGROW2
QSET2=QSET1+QSET2
QGRIDS=QGRID2
Vol II 119

Appendix D
Ecological Model

DO 90 J=1,N
DO 90 I=1,M
IF(DEPTH(I,J).GT.100.) GO TO 90
DTSUM=DTSUM+DOLD(I,J)*T(I+1,J+1)
90 CONTINUE
QGRID2=DTSUM*CONS2
QGRID1=QGRID2-QGRIDS
QEDGE1=QEDGE*CONS1*TDT
QEDGE=0.
QEDGE2=QEDGE2+QEDGE1
TOTOU1=QDKAY1+QEDGE1+QGRW1+QSET1
TOTOU2=QDKAY2+QEDGE2+QGRW2+QSET2
TOTIN1=QSORC1+QPROD1+QRESP1+QBN1
TOTIN2=QSORC2+QPROD2+QRESP2+QBN2
WRITE(6,209) QINIT
WRITE(6,207) QGRID2,QSORC2,QPROD2,QRESP2,QBN2,QDKAY2,QGRW2,
1 QSET2,QEDGE2,TOTIN2,TOTOU2
WRITE(6,205) QGRID1,QSORC1,QPROD1,QRESP1,QBN1,QDKAY1,QGRW1,
1 QSET1,QEDGE1,TOTIN1,TOTOU1
RETURN

C*****
C*****
C*****
SUBROUTINE MAPDOX(NNAME,T,SATDOX,QINIT,QSORC,QOBOD,QREAR,QEDGE,
1 QBEN,QONH3,QO2,QGRW,QRESP,QSORC2,QOBOD2,QREAR2,QEDGE2,QBEN2,
2 QONH32,QO22,QGRW2,QRESP2,QGRID2,KEY)
C*****
DOUBLE PRECISION QSORC,QOBOD,QREAR,QEDGE,QSORC2,QOBOD2,QREAR2,
1 QEDGE2,QBEN,QONH3,QO2,QGRW,QRESP,QBN2,QONH32,QO22,QGRW2,QRESP2
2 ,QGRID2,QINIT
DOUBLE PRECISION CONS1,CONS2
DOUBLE PRECISION QSORC1,QOBOD1,QREAR1,QGRID1,QTOT1,QTOT2,QBEN1,QONH31,QO21,QGRW1,QRESP1
COMMON/A/M,N,TIMECT,DY,DY,TDT,IPRDAY,KIDDAY,ITIM1,ITIM2
1 /B/69,40),V(68,41),DEP(68,40)
2 /C/DOLD(68,40),NSEL,IS(100),JS(100),QS(100),QSC(100),QSP(100),
3 QSPQ(100),QSN(100),QSNH3(100),QSN03(100),QSOX(100)
4 /M/CONS1,CONS2,BODU5
DIMENSION CONC(102),T(70,42),INT(102)
REAL*4 CNAME(4,2)
REAL*8 BLANK,STARS(102),STAR
INTEGER*4 ITI,IDAY,ITPH,ITPM

DATA BLANK,STARS /"*******/
DATA CNAME/'DISS','OLVE','Dox','YGEN','OXYG','END','EFIC','IT'/'

15 FORMAT(4X,22A5)
20 FORMAT(1X,I3,(22F5.1))
201 FORMAT(50X,'CONCENTRATIONS (MG/L)'),//)
203 FORMAT(' I J',2(1X,I2,2X),/)
Vol II 120 Appendix D
Ecological Model

205 FORMAT(2X,'CHANGE',1X,11F10.0)
207 FORMAT(11X,'IN GRID',3X,'SOURCES',5X,'REAR.',4X,'GROWTH',3X, 
1 'BOD OX.',3X,'NH3 OX.',3X,'RESPIR.',4X,'O2 GAS',1X,'BEN. DEM.',2X 
2 , 'OFF EDGE',2X,'TOTAL IN',1X,'TOTAL OUT',//,2X,'TOTAL',2X,12F10.0 
3 ,) 
208 FORMAT(50X,'TOTAL WEIGHT (POUNDS)',//) 
209 FORMAT(2X,'TOTAL INITIALLY IN THE GRID ',F10.0,/) 
215 FORMAT(1H1 ,22X,'TIME = ',I3,' DAYS','I3,' HOURS','I3,' MINUTES',1 
10X,4A4,/) 

C*****

ITI=TI+.9 
IDAY=ITI/1440 
ITPH=ITI/60-IDAY*24 
ITPM=ITI-ITPH*60-IDAY*1440 
NN=NNAME-7 
IF(NN.EQ.4) NN=2 
IF(KEY.EQ.5) GO TO 89 
DO 304 NSTRT=1,N,22 
NSTOP=NSTRT+21 
IF(NSTOP.GT.N) NSTOP=N 
WRITE(6,215) IDAY,ITPH,ITPM,(CNAME(L,NN),L=1,4) 
WRITE(6,201) 
WRITE(6,203) (J,J=NSTRT,NSTOP) 
DO 303 I=1,M 
DO 301 J=NSTRT,NSTOP 
STARS(J)=BLANK 
CONC(J)=T(I+1,J+1) 
IF(NN.EQ.2) CONC(J)=SATDOX-CONC(J) 
301 IF(T(I+1,J+1).GT.100.) STARS(J)=STAR 
WRITE(6,15) (STARS(J),J=NSTRT,NSTOP) 
WRITE(6,20) I,( CONC(J),J=NSTRT,NSTOP) 
WRITE(6,15) (STARS(J),J=NSTRT,NSTOP) 
303 CONTINUE 
WRITE(6,203) (J,J=NSTRT,NSTOP) 
304 CONTINUE 

89 IF(NN.EQ.2) RETURN 
WRITE(6,215) IDAY,ITPH,ITPM,(CNAME(L,NN),L=1,4) 
WRITE(6,208) 
QSORC1=QSORC*CONS2*TDT 
QREAR1=QREAR*CONS2 
QOBOD1=QOBOD*CONS2 
QONH31=QONH3*CONS2 
QBEN1=QBEN*CONS2 
QO21=QO2*CONS2 
QGROW1=QGROW*CONS2 
QRESP1=QRESP*CONS2 
QSORC=0. 
QOBOD=0. 
QREAR=0. 
QONH3=0. 
QBEN=0. 
QO2=0. 
QGROW=0. 
QRESP=0.
Vol II

QSORC2 = QSORC1 + QSORC2
QOBOD2 = QOBOD1 + QOBOD2
QREAR2 = QREAR1 + QREAR2
QONH32 = QONH31 + QONH32
QBEN2 = QBEN1 + QBEN2
Q22 = Q21 + Q22
QGROW2 = QGROW1 + QGROW2
QRESP2 = QRESP1 + QRESP2
QGRIDS = QGRID2
DTSUM = 0.

DO 90 J = 1, N
DO 90 I = 1, M
IF(DEPTH(I, J).GT.100.) GO TO 90
DTSUM = DTSUM + DOLD(I, J) * T(I+1, J+1)
90 CONTINUE
QGRID2 = DTSUM * CONS2
QGRID1 = QGRID2 - QGRIDS
QEDGE1 = QEDGE * CONS1 * TDT
QEDGE = 0.
QEDGE2 = QEDGE2 + QEDGE1
TOTIN1 = QSORC1 + QREAR1 + QGROW1
TOTIN2 = QSORC2 + QREAR2 + QGROW2
TOTOU1 = QRESP1 + QEDGE1 + Q021 + QOBOD1 + QONH31 + QBEN1
TOTOU2 = QRESP2 + QEDGE2 + Q022 + QOBOD2 + QONH32 + QBEN2
WRITE(6, 209) QINIT
WRITE(6, 207) QGRID2, QSORC2, QREAR2, QGROW2, QOBOD2, QONH32, QRESP2, Q22
WRITE(6, 205) QGRID1, QSORC1, QREAR1, QGROW1, QOBOD1, QONH31, QRESP1, Q21
RETURN

C*****
C*****
C*****

END

SUBROUTINE PLOTBX(QINIT, QGRID, QSORC, QDKAY, QPROD, QEDGE, QGROW2, QRESP2, QSET2, QBEN2, Q022, QONH32, CROMAX, RESP, GRAZE, ZOOEFF, HSATF, HSATN, PHOTO, EXTL1, EXTL2, XLIGHT, XSSATLI, CHL, BODU5, DCK, DKP, DKN, DKNH3, SETA, SETC, SETN, SETNH3, BENP04, BENNH3, BENN03, PC, FP, FN, PRF, OXGROW, OXRESP, OXN, TCOGRO, TCORSP, TCOGRZ, TCOC, TCOP, TCON, TCONH3, TCOREA, WIND, WINDCO, TEMP, SALIN, TI, TITLE1, TITLE2, AVGCON)

C*****
C*****
C*****

DOUBLE PRECISION QINIT(8), QGRID(8), QSORC(8), QDKAY(8), QPROD(8), QEDGE(8), QGROW2(8), QRESP2(8), QSET2(8), QBEN2(8), Q022, QONH32
REAL*4 TITLE1(4), TITLE2(4), AVGCON(8)
REAL*4 BOXX(8), BOXY(8), BOXHT(8), BOXWID(8), XARR(3, 41), YARR(3, 41)
REAL*4 BOXNAM(2, 8), RATENA(2, 46), RATEN(2)
REAL*4 HTNAME, HTWORD, HTNUM
REAL*4 QBEG, QEND, QSORC, QDKAY, QPROD, QGROW, QRESP, RATE(46)
INTEGER*2 NABOXL(8), MODARR(41), NARR(41), NL
INTEGER*4 IT, IDAY, IHOUR, IMIN

C*****

DATA BOXX/6.8, 2.0, 3.6, 5.2, 2.0, 3.6, 5.2, 3.5/,
1  BOXY/5.8, 1.6, 3.6, 3.6, 7.8, 7.8, 7.8, 5.3/,
2  BOXHT/7.8, 3.0/ , BOXWID/5.2, 6*1.2, 1.4/,
ECOLOGICAL MODEL

DATA MODARR/8*1,2,10*1,7*2,5,3*1,3,3,4,3,4,4,4,1,3/;
NARR/2,4,4,3*2,3,4*2,3*4,2,4,6,3,4,2,3,0,3*3,4,3*3,1,2,4,2,
4,1,3/;
XARR/1.8,1.8,0.,3.4,3.4,0.,5.0,5.0,0.,1.8,1.8,0.,3.4,3.4,0.,
5:0:5:0:0.,1.:4,2,0,0.,7:0:7:0:0.,2:4,2,6:2,6:3,6:3,6:0.,
5:2:5:2:0,0,2,0,2,0,0,3,6,3,6,0,5,2,5,2,0,4,4,4,4,0,;
6:8,6:8,0,0,8,2,0,0,7,2,8,0,0,2,4,8,0,0,3:8,3,8:8,0,;
7:2,2,2,8,0,3,8,3,8,0,8,0,5,4,5,4,8,0,4,0,4,0,;
8:0,5:4,5:4,2,8,2,8,8,0,2:1,2,1,0,,4,0,4,8,0,,;
2,4,3,2,0,4,0,4,8,0,4,4,4,0,,5,6,6,4,0,,3,6,3,6,4,,
1,5,6,6,4,0,,6,4,5:0,0,,6,4,3,6,3,6,6,4,2,0,2,0,;
6,4,2:4,2,4:4,8,6,6,6,6,6,2,0,0,8,0,,5,0,6,4,0,,;
YARR/2,8,2,2,0,,1:4,3,0,0,,1,4,3,0,0,,1,0,2,8,4,0,,
10,2,8,4,0,,10,2,8,4,0,,5,6,5,6,0,,3,2,1,2,0,;
1,4,1,4,6,3,0,1,2,0,0,3,0,1,2,0,,8,4,10,4,0,;
6,8,4,10,4,0,8,4,10,4,0,4,6,1,2,0,8,4,10,2,0,;
5,0,5,0,0,5,8,5,8,0,1,6,1,6,0,3,0,2,4,2,4,;
8,4,10,0,1,0,,8,4,9,6,9,6,2,8,2,8,3,0,9,2,9,2,8,4,;
8,8,8,8,8,4,4,6,2,0,2,0,2,2,4,6,0,3,6,3,6,0,;
7,8,7,8,0,0,7,8,7,8,0,0,6,0,7,8,0,0,3,6,3,6,0,,
2,7,2,6,8,6,8,7,8,7,8,0,0,5,0,5,0,0,0,4,4,4,4,4,4,2,;
6,4,6,4,7,2,4,5,4,5,2,2,8,8,3,2,4,8,4,8,0,,5,6,5,6,0,;
DATA RATENA/GROM,'AX = ',' RE ', ' SP = ', ' GRO ', ' ZE = ',
'ZOOE', ' FF = ', ' HSA ',
'TP = ', ' HSA ', ' TN = ', ' PHO ', ' TO = ', ' EXT ', ' L1 = ',
' EXT ', ' L2 = ',
'XLIG', ' HT = ', ' XSAT ', ' LI = ', ' C ', ' HL = ',
' BOD ', ' U5 = ', ' D ',
'KC = ', ' D ', ' KP = ', ' D ', ' KN = ', ' DKN ', ' H3 = ',
' SE ', ' TA = ', ' SE ', ' TC = ', ' SE ', ' TP = ', ' SE ',
'TN = ', ' SETN ', ' H3 = ', ' BENP ', ' O4 = ', ' BEN ', ' H3 = ',
'BENN ', ' O3 = ', ' FC = ', ' FP = ',
'FN = ', ' P ',
'RF = ', ' OXGR ', ' OW = ', ' OXRE ', ' SP = ', ' O ', ' XC = ',
'OXN ', ' H3 = ',
'BEN ', ' OX = ', ' TC ', ' RO = ', ' TCOR ', ' SP = ', ' TC ',
'Z = ', ' TC ', ' OC = ', ' TC ', ' OP = ', ' TC ', ' ON = ',
'TCON ', ' H3 = ', ' TCOR ', ' EA = ', ' WI ', ' ND = ', ' WIND ',
'CO = ', ' TE ', ' MP = ', ' SAL ', ' IN = '/

C****

XYCENT(NL,XY1,XY2,HT)=(XY1+XY2-NL*HT*8./9.)/2.
NUMRAT=46
NUMARR=41
RATE(1)=GROMAX
RATE(2)=RESP
RATE(3)=GRAZE
RATE(4)=ZOOEFF
RATE(5)=HSATP
RATE(6)=HSATN
RATE(7)=PHOTO
Vol II

RATE(8) =EXTL1
RATE(9) =EXTL2
RATE(10) =XLIGHT
RATE(11) =XSATL1
RATE(12) =CHL
RATE(13) =BODU5
RATE(14) =DKC
RATE(15) =DKP
RATE(16) =DKN
RATE(17) =DKNH3
RATE(18) =SETA
RATE(19) =SETC
RATE(20) =SETP
RATE(21) =SETN
RATE(22) =SETNH3
RATE(23) =BENPO4
RATE(24) =BENNH3
RATE(25) =BENNO3
RATE(26) =FC
RATE(27) =FP
RATE(28) =FN
RATE(29) =PRF
RATE(30) =OXGROW
RATE(31) =OXRESP
RATE(32) =OC
RATE(33) =OXNH3
RATE(34) =BENOX
RATE(35) =TCOGRO
RATE(36) =TCORSP
RATE(37) =TCORZ
RATE(38) =TCOC
RATE(39) =TCOP
RATE(40) =TCON
RATE(41) =TCONH3
RATE(42) =TCOREA
RATE(43) =WIND
RATE(44) =WINDCO
RATE(45) =TEMP
RATE(46) =SALIN

CALL INITP2(1,2)
CALL PLOT(5.0,0.,-3)
HTNAME=.15
HTNUM=.1
HTWORD=.075
DO 1 NB=1,8
QBEG=QINIT(NB)*1.D-6
QEND=GRID(NB)*1.D-6
CALL BOX(BOXX(NB),BOXY(NB),BOXHT(NB),BOXWID(NB),HTNAME,HTNUM,
1 AVGCONE(NB),QBEG,QEND,NABOXL(NB),BOXNAM(1,NB),BOXNAM(2,NB))
1 CONTINUE
DO 2 NA=1,NUMARR
GO TO (21,22,23,24,25),MODARR(NA)
21 CALL ARROW1(XARR(1,NA),YARR(1,NA),XARR(2,NA),YARR(2,NA),NARR(NA))
GO TO 2
CALL ARROW2(XARR(1,NA), YARR(1,NA), XARR(2,NA), YARR(2,NA),
              1 XARR(3,NA), YARR(3,NA), NARR(NA))
GO TO 2

CALL ARROW3(XARR(1,NA), YARR(1,NA), XARR(2,NA), YARR(2,NA), NARR(NA))
GO TO 2

CALL ARROW4(XARR(1,NA), YARR(1,NA), XARR(2,NA), YARR(2,NA),
              1 XARR(3,NA), YARR(3,NA), NARR(NA))
GO TO 2

CALL ARROW5(XARR(1,NA), YARR(1,NA), XARR(2,NA), YARR(2,NA))

2 CONTINUE
CALL DASH(1.,0.,1.,11.,4)
CALL DASH(8.;0.;8.;11.;5)
CALL SYMBOL(.95,2.0,HTNAME,'Atmosphere',90.,10)
CALL SYMBOL(8.2,2.0,HTNAME,'Benthos',90.,7)

NL=7
DO 3 NA=1,6
XWORD=XARR(1,NA)-HTWORD/2.
YWORD=XYCENT(NL,YARR(1,NA),YARR(2,NA),HTWORD)
CALL SYMBOL(XWORD,YWORD,HTWORD,'Sources',90.,NL)
3 CONTINUE
NA=7
XWORD=(XARR(1,NA)+XARR(2,NA))/2.
YWORD=YARR(1,NA)+HTWORD/2.
CALL SYMBOL(XWORD,YWORD,HTWORD,'Sources',90.,NL)

NL=8
DO 4 NA=8,15
IF(NA.EQ.9) GO TO 4
XWORD=XARR(1,NA)-HTWORD/2.
YWORD=XYCENT(NL,YARR(1,NA),YARR(2,NA),HTWORD)
CALL SYMBOL(XWORD,YWORD,HTWORD,'Off Edge',90.,NL)
4 CONTINUE
NA=9
XWORD=XARR(2,NA)-HTWORD/2.
YWORD=XYCENT(NL,YARR(2,NA),YARR(3,NA),HTWORD)
CALL SYMBOL(XWORD,YWORD,HTWORD,'Off Edge',90.,NL)

NL=8
NA=16
XWORD=XARR(1,NA)-HTWORD/2.
YWORD=XYCENT(NL,YARR(1,NA),YARR(2,NA),HTWORD)
CALL SYMBOL(XWORD,YWORD,HTWORD,'Grazing',90.,NL)

NL=10
NA=17
XWORD=XARR(1,NA)+HTWORD
YWORD=YARR(1,NA)+HTWORD/2.
CALL SYMBOL(XWORD,YWORD,HTWORD,'Reaeration',90.,NL)

NL=6
NA=40
XWORD=XARR(2,NA)+HTWORD
YWORD=YARR(2,NA)-HTWORD-NL*HTWORD*8./9.
CALL SYMBOL(XWORD,YWORD,HTWORD,'O2 Gas',90.,NL)

NL=5
NA=39
XWORD=XARR(1,NA)-HTWORD/2.
YWORD=XYCENT(NL,YARR(1,NA),YARR(2,NA),HTWORD)
CALL SYMBOL(XWORD,YWORD,HTWORD,'Light',90.,NL)
NLB=6
DO 5 NA=1,6
IF((NA.EQ.2).OR.(NA.EQ.3)) GO TO 5
XW=XARR(1,NA)+HTNUM/2.
YW=YARR(1,NA)+HTNUM/2.
SORC=QSORC(NA+1)*1.D-6
CALL NUMBER(XW,YW,HTNUM,SORC,90.,3)
5 CONTINUE
DO 6 NA=2,3
XW=XARR(1,NA)+HTNUM/2.
YW=YARR(1,NA)-(.5+NLB)*HTNUM*8./9.
SORC=QSORC(NA+1)*1.D-6
CALL NUMBER(XW,YW,HTNUM,SORC,90.,3)
6 CONTINUE
NA=7
XW=XARR(1,NA)-HTNUM/2.
YW=YARR(1,NA)-NLB/2.*HTNUM*8./9.
SORC=QSORC(NA+1)*1.D-6
CALL NUMBER(XW,YW,HTNUM,SORC,90.,3)
7 CONTINUE
NA=16
XW=XARR(1,NA)+HTNUM/2.
YW=YARR(2,NA)+HTNUM/2.
DKAY=QDKAY(1)*1.D-6
CALL NUMBER(XW,YW,HTNUM,DKAY,90.,3)
NA=27
XW=XARR(1,NA)-HTNUM/2.
YW=(YARR(1,NA)+YARR(2,NA))/2.-NLB/2.*HTNUM*8./9.
DKAY=QDKAY(2)*1.D-6
CALL NUMBER(XW,YW,HTNUM,DKAY,90.,3)
NA=17
XW=XARR(1,NA)-HTNUM/2.
YW=YARR(1,NA)
PROD=QPROD(8)*1.D-6
CALL NUMBER(XW,YW,HTNUM,PROD,90.,3)
NA=40
XW=XARR(2,NA)-HTNUM/2.
YW=YARR(2,NA)-NLB*HTNUM*8./9.-HTNUM/2.
O2=QO22*1.D-6
CALL NUMBER(XW,YW,HTNUM,O2,90.,3)
NQ=4
DO 8 NA=32,35
IF(NA.EQ.33) GO TO 8
XW=(XARR(1,NA)+XARR(2,NA))/2.
YW = YARR(1, NA) + HTNUM/2.
QGROW = QGROW2(NQ) * 1.D-6
CALL NUMBER(XW, YW, HTNUM, QGROW, 90., 3)
NQ = NQ + 1
IF(NQ.EQ.5) NQ = NQ + 2

8 CONTINUE
NA = 33
XW = XARR(1, NA) - HTNUM/2.
YW = YARR(1, NA) - NLB*HTNUM*8./9.
QGROW = QGROW2(6) * 1.D-6
CALL NUMBER(XW, YW, HTNUM, QGROW, 90., 3)
NA = 36
XW = XARR(1, NA) - .5
YW = YARR(1, NA) + HTNUM/2.
QRESP = QRESP2(3) * 1.D-6
CALL NUMBER(XW, YW, HTNUM, QRESP, 90., 3)
NA = 37
XW = XARR(2, NA) - HTNUM/2.
YW = YARR(3, NA) - (1+NLB)*HTNUM*8./9.
QRESP = QRESP2(5) * 1.D-6
CALL NUMBER(XW, YW, HTNUM, QRESP, 90., 3)
NA = 38
XW = XARR(2, NA) - HTNUM/2.
YW = (YARR(2, NA) + YARR(3, NA))/2. - NLB/2.*HTNUM*8./9.
QRESP = QRESP2(2) * 1.D-6
CALL NUMBER(XW, YW, HTNUM, QRESP, 90., 3)
DO 9 NA = 8, 15
XW = XARR(2, NA) + HTNUM/2.
IF(NA.EQ.9) XW = XARR(2, NA) + 1.5*HTNUM
YW = YARR(2, NA) - (.5 + NLB)*HTNUM*8./9.
IF((NA.GE.12).AND.(NA.LE.14)) YW = YARR(2, NA) + HTNUM/2.
EDGE = QEDGE(NA-7) * 1.D-6
CALL NUMBER(XW, YW, HTNUM, EDGE, 90., 3)
9 CONTINUE
NA = 31
XW = XARR(1, NA) - HTNUM/2.
YW = YARR(1, NA) + HTNUM/2.
QONH3 = QONH32 * 1.D-6
CALL NUMBER(XW, YW, HTNUM, QONH3, 90., 3)
NQ = 1
XD = 5*.125
DO 10 NA = 18, 22
XW = XARR(2, NA) - XD
IF(NA.GE.20) XW = XARR(3, NA) - XD
YW = YARR(2, NA) + HTNUM/2.
IF(NA.GE.20) YW = YARR(3, NA) + HTNUM/2.
QSET = QSET2(NQ) * 1.D-6
CALL NUMBER(XW, YW, HTNUM, QSET, 90., 3)
NQ = NQ + 1
IF(NQ.EQ.4) NQ=NQ+1
XD=XD-.125
10 CONTINUE
NQ=4
XD=5*.125
DO 11 NA=23,25
XW=XARR(1,NA)-XD
YW=YARR(1,NA)+HTNUM/2.
QBEN=QBEN2(NQ)*1.D-6
CALL NUMBER(XW,YW,HTNUM,QBEN,90.,3)
NQ=NQ+1
11 CONTINUE
NA=26
XW=XARR(3,NA)-.125
YW=YARR(3,NA)+HTNUM/2.
QBEN=QBEN2(8)*1.D-6
CALL NUMBER(XW,YW,HTNUM,QBEN,90.,3)
NL=7
XW=BOXX(1)
YWS=BOXY(1)-BOXWID(1)/2.+2.
CALL SYMBOL(XW,YWS,HTNUM,'Growth=',90.,NL)
YWN=YWS+NL*HTNUM*8./9.
QGROW=QGROW2(1)*1.D-6
CALL NUMBER(XW,YWN,HTNUM,QGROW,90.,3)
NL=12
YWS=BOXY(1)+BOXWID(1)/2.-.2-(NL+NLB)*HTNUM*8./9.
CALL SYMBOL(XW,YWS,HTNUM,'Respiration=',90.,NL)
YWN=YWS+NL*HTNUM*8./9.
QRESP=QRESP2(1)*1.D-6
CALL NUMBER(XW,YWN,HTNUM,QRESP,90.,3)
HTWORD=.1
NL=76
XWORD=9.2
YWORD=XYCENT(NL,0.,11.,HTWORD)
CALL SYMBOL(XWORD,YWORD,HTWORD,'Relationships of Water Quality Parameters in the U.S.F. Ecological Bay Model',90.,NL)
NL=35
XWORD=XWORD+.2
YWORD=XYCENT(NL,0.,11.,HTWORD)
CALL SYMBOL(XWORD,YWORD,HTWORD,'Total millions of pounds in the bay',90.,NL)
NL=41
XWORD=XWORD+.2
YWORD=XYCENT(NL,0.,11.,HTWORD)
CALL SYMBOL(XWORD,YWORD,HTWORD,'(Average concentration in the bay 1(mg/l))',90.,NL)
CALL DASH(0.,0.,0.,13.,0)
CALL DASH(9.,0.,9.,13.,0)
CALL DASH(0.,0.,9.,0.,0)
CALL DASH(0.,11.,9.,11.,0)
CALL DASH(0.,13.,9.,13.,0)
CALL DASH(.75,11.,.75,13.,0)
CALL DASH(8.11,8.13,0)
HTWORD=.125
NL=16
XNAME=.4
XD=.2
YNAME=XYCENT(NL,11,13,HTWORD)
CALL SYMBOL(XNAME,YNAME,HTWORD,TITLE1,90,NL)
XNAME=XNAME+XD
YNAME=XYCENT(NL,11,13,HTWORD)
CALL SYMBOL(XNAME,YNAME,HTWORD,TITLE2,90,NL)
HTWORD=.125
XNAME=.95
YNAME=11.1
CALL SYMBOL(XNAME,YNAME,HTWORD,'Rates & Constants',90,17)
HTRATE=1
XD=.15
NL=8
YRATES=11.2
YRATEN=YRATES+(1+NL)*HTRATE*8.9.
DO 12 NR=1,NUMRAT
RATEN(1)=RATENA(1,NR)
RATEN(2)=RATENA(2,NR)
XRATE=.05+XD*NR
CALL SYMBOL(XRATE,YRATES,HTRATE,RATEN,90,NL)
CALL NUMBER(XRATE,YRATEN,HTRATE,RATE(NR),90,-1)
12 CONTINUE
IT=IT
IDAY=IT/1440
IHOUR=(IT-IDAY*1440)/60
IMIN=IT-IDAY*1440-IHOUR*60
DDAY=IDAY
DHOUR=IHOUR
DMIN=IMIN
HTWORD=.1
XW=.2
YW=11.1
XD=.125
CALL SYMBOL(XW,YW,HTWORD,'Elapsed Time Since',90,18)
XW=XW+XD
CALL SYMBOL(XW,YW,HTWORD,'Startup',90,7)
NL=9
XW=XW+XD
YWS=YWS+(1+NL)*HTWORD*8.9.
CALL SYMBOL(XW,YWS,HTWORD,' Days =',90,NL)
CALL NUMBER(XW,YWS,HTWORD,DDAY,90,-1)
XW=XW+XD
CALL SYMBOL(XW,YWS,HTWORD,' Hours =',90,NL)
CALL NUMBER(XW,YWS,HTWORD,DHOUR,90,-1)
XW=XW+XD
CALL SYMBOL(XW,YWS,HTWORD,'Minutes =',90,NL)
CALL NUMBER(XW,YWS,HTWORD,DMIN,90,-1)
CALL PLOT(6,0,-3)
CALL ENDPLT
SUBROUTINE BOX(X, Y, HT, WID, HTNAME, HTNUM, TEND, QBEG, QEND, NL, BOXN)

REAL*4 BOXN(2)
REAL*4 X, Y, HT, WID, HTNAME, HTNUM, QBEG, QEND
INTEGER*2 NL, NLB, NLE, NLC

XTOP = X - HT/2.
XBOT = X + HT/2.
YLEFT = Y - WID/2.
YRIGHT = Y + WID/2.
CALL PLOT(XTOP, YLEFT, 3)
CALL PLOT(XTOP, YRIGHT, 2)
CALL PLOT(XBOT, YRIGHT, 2)
CALL PLOT(XBOT, YLEFT, 2)
CALL PLOT(XTOP, YLEFT, 2)
XNAME = X - HT/4.
YNAME = Y - NL/2 * (HTNAME * 8./9.)
CALL SYMBOL(XNAME, YNAME, HTNAME, BOXN, 90., NL)

NLB = 6
XBEG = X + 1.5 * HTNUM
YBEG = Y - (NLB + 6) * HTNUM * 8./9. / 2.
CALL SYMBOL(XBEG, YBEG, HTNUM, 'Start=', 90., NLB)
YBEGIN = YBEG + NLB * HTNUM * 8./9.
CALL NUMBER(XBEG, YBEGIN, HTNUM, QBEG, 90., 3)
XEND = XBEG + 1.5 * HTNUM
NLE = 4
YEND = YBEG + (NLB - NLE) * HTNUM * 8./9.
CALL SYMBOL(XEND, YEND, HTNUM, 'End=', 90., NLE)
YEND = YEND + NLE * HTNUM * 8./9.
CALL NUMBER(XEND, YEND, HTNUM, QEND, 90., 3)
NLC = 7
XCON = X
YCONS = Y - (NLC * HTNUM * 8./9.) / 2.
CALL SYMBOL(XCON, YCONS, HTNUM, '(' , 90., NLC)
YCON = YCONS + HTNUM * 8./9.
CALL NUMBER(XCON, YCON, HTNUM, TEND, 90., 2)
RETURN
END

SUBROUTINE ARROW1(X1, Y1, X2, Y2, NARR)

CALL PLOT(X1, Y1, 3)
CALL PLOT(X2, Y2, 2)
GO TO (1, 2, 3, 4), NARR
1 X3 = X2 + 1
   GO TO 30
2 Y3 = Y2 + 1

GO TO 40
3 X3 = X2 -.1
30 X4 = X3
Y3 = Y2 -.1
Y4 = Y2 +.1
GO TO 5
4 Y3 = Y2 -.1
40 Y4 = X3
X3 = X2 +.1
X4 = X2 -.1
5 CALL PLOT(X3, Y3, 2)
CALL PLOT(X4, Y4, 2)
RETURN
END

SUBROUTINE ARROW2(X1, Y1, X2, Y2, X3, Y3, NARR)

CALL PLOT(X1, Y1, 3)
CALL PLOT(X2, Y2, 2)
CALL ARROW1(X2, Y2, X3, Y3, NARR)
RETURN
END

SUBROUTINE ARROW3(X1, Y1, X2, Y2, NARR)

CALL DASH(X1, Y1, X2, Y2, 2)
GO TO (1, 2, 3, 4), NARR
1 X3 = X2 +.1
GO TO 30
2 Y3 = X2 +.1
GO TO 40
3 X3 = X2 -.1
30 X4 = X3
Y3 = Y2 -.1
Y4 = Y2 +.1
GO TO 5
4 Y3 = Y2 -.1
40 Y4 = X3
X3 = X2 +.1
X4 = X2 -.1
5 CALL DASH(X2, Y2, X3, Y3, 2)
CALL DASH(X3, Y3, X4, Y4, 2)
CALL DASH(X4, Y4, X2, Y2, 2)
RETURN
END

SUBROUTINE ARROW4(X1, Y1, X2, Y2, X3, Y3, NARR)
CALL DASH(X1,Y1,X2,Y2,2)
CALL ARROW3(X2,Y2,X3,Y3,NARR)
RETURN
END

SUBROUTINE ARROW5(X1,Y1,X2,Y2)

CALL DASH(X1,Y1,X2,Y2,2)
X3=X1
Y3=Y1+.3
CALL ARROW3(X1,Y1,X3,Y3,4)
X4=X2
Y4=Y2-.3
CALL ARROW3(X2,Y2,X4,Y4,2)
RETURN
END

SUBROUTINE DASH(X1,Y1,X2,Y2,MODE)

IF(MODE.EQ.0) GO TO 10
DELEX=X2-X1
DELEY=Y2-Y1
RLEN=SQRT(DELEX**2+DELEY**2)
S=DELEY/RLEN
C=DELEX/RLEN
CALL PLOT(X1,Y1,3)
GO TO (10,20,30,40,50,60,70,80,90,100), MODE+1
10 CALL PLOT(X1,Y1,3)
CALL PLOT(X2,Y2,2)
RETURN
20 D=.02
22 NPTS=RLEN/D
X=X1
Y=Y1
DO 28 I=1,NPTS
CALL PLOT(X,Y,3)
CALL PLOT(X,Y,2)
X=X+C*D
Y=Y+S*D
28 CONTINUE
RETURN
30 D=.04
GO TO 22
40 D=.1
GO TO 22
50 D=.2
GO TO 22
60 D=.0625
62 SP=D/2.
Vol II

Ecoctical Model

Appendix D

DOUBLE PRECISION RDT, GR0, DGR0MX, DRESP, DSETA, DGRAZE, DDKC, DSETC, 1
DDKP1, DSETP1, DBENP2, DDKN1, DSETN1, DDKN2, DSETN2, DBENN2, DBENN3,
2
DBNOX
DOUBLE PRECISION CONS1, CONS2
COMMON/B/U(69,40), V(68,41), DEPTH(68,40)
1 /C/DOLD(68,40), NSEW, IS(100), JS(100), QS(100), QSC(100), QSP(100),

XD=D*C
YD=D*S
XSP=SP*C
YSP=SP*S
IF((D*2.+SP).LE.RLEN) GO TO 66
NPTS=1
CONTINUE
DO 65 I=1,NPTS
CALL RELPT(XD,YD,2)
CALL RELPT(XSP,YSP,3)
CONTINUE
RETURN
66 NPTS=(RLEN-D)/(D+SP)+1
SP=(RLEN-D*NPTS)/(NPTS-1)
XSP=SP*C
YSP=SP*S
GO TO 64
70 D=.125
GO TO 62
80 D=.25
GO TO 62
90 D=.5
GO TO 62
100 D=1.0
GO TO 62
END
SUBROUTINE RELPT(RELX,RELY,IPEN)
INTEGER*2 IPEN
REAL RELX,RELY
REAL ROTATX, ROTATY
THETA=0.
TSIN=SIND(THETA)
TCOS=COSD(THETA)
ROTATX=((RELX*TCOS)-(RELY*TSIN))
ROTATY=((RELX*TSIN)+(RELY*TCOS))
CALL RPLDT(ROTATX,ROTATY,IPEN)
RETURN
END

BLOCK DATA

DOUBLE PRECISION RDT, GR0, DGR0MX, DRESP, DSETA, DGRAZE, DDKC, DSETC,
1 DDKP1, DSETP1, DBENP2, DDKN1, DSETN1, DDKN2, DSETN2, DBENN2, DBENN3,
2 DBNOX
DOUBLE PRECISION CONS1, CONS2
COMMON/B/U(69,40), V(68,41), DEPTH(68,40)
1 /C/DOLD(68,40), NSEW, IS(100), JS(100), QS(100), QSC(100), QSP(100),
Vol II 133 Appendix D

Ecological Model

2  QSP04(100), QSN(100), QSNH3(100), QSN03(100), QSOX(100)
3  /D/TA(70,42), OLDA(70,42), GRO(70,42), FC, FP, FN, DGROMX, HSATP, HSATN,
4  PHOTO, EXTILI, EXTL2, XLIGHT, XSATLI, RDT, DRESP, DSETA, DGRAZE, ZOEOFF
5  /E/TC(70,42), OLDC(70,42), DDKC, DSETC, OXC
6  /F/TP(70,42), OLDP(70,42), DDP1, DSETP1
7  /G/TP04(70,42), DBENP2(68,40)
8  /H/TN(70,42), OLDN(70,42), DDKN1, DSETN1
9  /I/TNH3(70,42), OLDNH3(70,42), PREF(70,42), DDKN2, PRF, DSETN2,
10  DBENN2(68,40)
11  /J/TN03(70,42), DBENN3(68,40)
12  /K/TOX(70,42), WINDF, WINDCO, TCOREA, TEMP, OXGROW, OXRESP, OXNH3,
13  DBENOX(68,40)
14  /L/NPP, IPI(10), JPI(10), IPO(10), JPO(10), QPO(10)
15  /M/CONS1, CONS2, BODU5

C*****
DATA U/2760*0./, V/2788*0./, DEPTH/2720*0./
DATA DOLD/2720*0./, NSEW/0./, IS, JS/200*0./, QS, QSC, QSP, QSP04, QSN,
1  QSNH3, QSN03, QSOX/800*0./
DATA TA, OLDA/5880*0./, GRO/2940*0./
DATA TC, OLDC/5880*0./, TP, OLDP/5880*0./, TPO4/2940*0./
1  TN, OLDN/5880*0./
DATA TNH3, OLDNH3/5880*0./, PREF/2940*0./, TNO3/2940*0./
DATA TOX/2940*0./, DBENOX/2720*0./
DATA DBEP2/2720*0./, DBENN2/2720*0./, DBENN3/2720*0./
DATA NPP/0./, IPI, JPI, IPO, JPO/40*0./, QPO/10*0./
END
C*****
C*****
C*****
SUBROUTINE OPENLH ( XLENTH, YLENTH ) /* OPEN & INIT LARGE HORIZ PLOT
REAL  XLENTH, YLENTH
C Prompts the user for the plot file name, and opens the file.
C Initializes a DIPLOT LARGE HORIZONTAL Plot File. (X26", Y13")
C Returns default values for the axis lengths.
C Requires loading Primos Applications Library. Use LI VAPPLE
C 22 JAN 84, CIVIL DEPT. USF. - Larry Sanders
C
$INSERT SYSCOM>A$KEYS
LOGICAL SUCCES
INTEGER*2 FILNAM(17), NAMLEN, WATIME, RETRY, DEVICE, PFUNIT,
$  FILOUT, TTY, RSVD, DEVICE
DATA NAMLEN /32/, WATIME /1/, RETRY /1/, FILOUT /3/, TTY /0/
C CALL TNOU ('>> OPENLH >>', 12 )
10 SUCCESS = OPVP$A('ENTER output PLOT file name. ', 29,
$  A$WRIT+A$SAFM+A$GETU, FILNAM, NAMLEN, PFUNIT,
$  A$VNEW, WATIME, RETRY)
  IF( .NOT. SUCCESS ) GOTO 10
C
DEVICE = 7 /* DISK FILE COMPRESSED ASCII
10 SUCCESS = OPVP$A('ENTER output PLOT file name. ', 29,
$  A$WRIT+A$SAFM+A$GETU, FILNAM, NAMLEN, PFUNIT,
$  A$VNEW, WATIME, RETRY)
  IF( .NOT. SUCCESS ) GOTO 10
C
DEVICE = 5 /* LARGE HORIZONTAL PLOT
CALL INIT$$ ( FILOUT, PFUNIT, TTY, TTY, DEVICE, RSVD )
XLENTH = 26.0
YLENTH = 13.0

C
CALL TNOU ('$$ OPENLH $$', 12)
1000 RETURN

C******************************************************************************
END /******************************************************************************

OPENLH.FTN *****