

ANALYTICAL SOLUTIONS FOR SEQUENTIALLY COUPLED MULTI-SPECIES
REACTIVE TRANSPORT PROBLEMS

Except where reference is made to the work of others, the work described in this thesis is my own or was done in collaboration with my advisory committee. This thesis does not include proprietary or classified information.

Venkatraman Srinivasan

Certificate of Approval:

Mark O. Barnett
Associate Professor
Civil Engineering

T. Prabhakar Clement, Chair
Professor
Civil Engineering

Dongye Zhao
Associate Professor
Civil Engineering

George T. Flowers
Interim Dean
Graduate School

ANALYTICAL SOLUTIONS FOR SEQUENTIALLY COUPLED MULTI-SPECIES
REACTIVE TRANSPORT PROBLEMS

Venkatraman Srinivasan

A Thesis

Submitted to

the Graduate Faculty of

Auburn University

in Partial Fulfillment of the

Requirements for the

Degree of

Master of Science

Auburn, Alabama
December 17, 2007

ANALYTICAL SOLUTIONS FOR SEQUENTIALLY COUPLED MULTI-SPECIES
REACTIVE TRANSPORT PROBLEMS

Venkatraman Srinivasan

Permission is granted to Auburn University to make copies of this thesis at its discretion, upon request of individuals or institutions and at their expense. The author reserves all publication rights.

Signature of Author

Date of Graduation

THESIS ABSTRACT

ANALYTICAL SOLUTIONS FOR SEQUENTIALLY COUPLED MULTI-SPECIES
REACTIVE TRANSPORT PROBLEMS

Venkatraman Srinivasan

Master of Science, December 17, 2007
(B.Tech., Indian Institute of Technology Madras, Chennai, India, 2004)

216 Typed Pages

Directed by T. Prabhakar Clement

Multi-species reactive transport equations coupled through sorption and sequential first-order reactions are commonly used to model sites contaminated with radioactive wastes, chlorinated solvents and nitrogenous species. Although researchers have been attempting to solve various forms of this reactive transport problem for over fifty years, a general closed-form analytical solution to this problem is not available in the published literature. In the first part of this research work, a closed-form analytical solution to this problem is deduced involving a generic spatially-varying initial condition. Two distinct solutions are derived for Dirichlet and Cauchy boundary conditions each with Bateman-type source terms. The proposed solution procedure employs a combination of Laplace and linear transform methods to uncouple and solve the system of partial differential equations. The final solution is organized and presented in a

general format that represents the solutions to both the boundary conditions. In addition, the mathematical concepts for deriving the solution within a generic framework that can be used for solving similar transport problems are also presented.

In the second part of this research work, the computational techniques for implementing the new solutions are discussed. These techniques are then adopted to develop a general computer code which is used to verify the solutions. In addition, several special-case solutions for simpler transport problems involving zero initial condition, identical retardation factors, zero advection, zero dispersion and steady-state condition are also derived. Where ever possible, these special-case solutions are compared against previously published analytical solutions to establish the validity of the new solution. The performance of the new solution is tested against other published analytical and semi-analytical solutions using a set of example problems. Finally, an investigation into extending the general solution to multiple dimensions using the approximate Domenico solution is also presented.

This thesis has produced the following three journal publications:

- 1) V. Srinivasan, T.P. Clement, and K.K. Lee. “Domenico Solution – Is it Valid?”, *Ground Water*, 25(2): 136-146, May 2007.
- 2) V. Srinivasan, and T.P. Clement. “Analytical Solutions for Sequentially Coupled Reactive Transport Problems – Part I: Mathematical Derivations”, *Submitted May 2007*, *Advances in Water Resources*.
- 3) V. Srinivasan, and T.P. Clement. “Analytical Solutions for Sequentially Coupled Reactive Transport Problems – Part II: Special Cases, Implementation and Testing”, *Submitted May 2007*, *Advances in Water Resources*.

ACKNOWLEDGMENTS

I would like to express my sincere gratitude to my advisor Dr. T. Prabhakar Clement for his valuable guidance and constant encouragement during my graduate studies here at Auburn University. I would like to thank my committee members Dr. Mark O. Barnett and Dr. Dongye Zhao for their support in the preparation of my thesis. I would also like to thank Dr. Kang-Kun Lee, Dr. Rein van Genuchten, Dr. Joel Melville, Dr. Jacob Dane and Dr. Willie Harper and for their guidance throughout my research work.

Funding for this research work was, in part, supported by the Korean Research Council's Frontier project, funded through the Seoul National University, Seoul, Korea, and by the Office of Science (BER), U.S. Department of Energy Grant no. DE-FG02-06ER64213.

Style manual of journal used Advances in Water Resources

Computer software used Microsoft Office Word 2003

TABLE OF CONTENTS

LIST OF TABLES	xi
LIST OF FIGURES	xii
I. INTRODUCTION.....	1
1.1 Background.....	1
1.2 Literature Review.....	2
1.3 Scope and Objective	7
1.4 Organization.....	7
II. GOVERNING EQUATIONS AND SOLUTION DERIVATION.....	9
2.1 Governing Equations	9
2.2 Derivation of the Solution for the Dirichlet Boundary condition.....	10
2.3 Derivation of the Analytical Solution for the Cauchy Boundary Condition	27
2.4 Discussion.....	30
III. TESTING AND VALIDATION OF THE ANALYTICAL SOLUTIONS	33
3.1 Introduction.....	33
3.2 Implementation of the Solution	34
3.3 Special Cases	38
3.3.1 Zero Initial Condition	38
3.3.2 Identical Retardation Factors	39
3.3.3 Zero Advection Velocity.....	40

3.3.4	Steady State.....	42
3.3.5	Zero Dispersion Coefficient.....	44
3.4	Example Problems	46
3.5	Salient Features of the General Solution	55
3.6	Limitations of the General Solution.....	56
IV. INVESTIGATION OF ERROR ASSOCIATED WITH THE DOMENICO		
APPROXIMATION		59
4.1	Introduction.....	59
4.2	Governing Equations	61
4.3	Review of the Domenico Solution.....	62
4.4	A Rigorous Approach to Derive the Domenico Solution	67
4.5	Mathematical Analysis of the Validity of the Approximation Involved in the Domenico Solution	69
4.6	Analysis of the Error Associated With the Domenico Solution	71
4.7	Example Problem.....	73
4.7.1	Plume Comparison Analysis Behind the Advective Front	73
4.7.2	Plume Comparison Analysis Beyond the Advective Front	78
4.8	Discussions and Recommendations	84
V. SUMMARY AND CONCLUSIONS		88
REFERENCES		91
APPENDIX A.....		99
APPENDIX B		113
APPENDIX C		118

APPENDIX D.....	127
APPENDIX E.....	140
APPENDIX F.....	154
APPENDIX G.....	162
APPENDIX H.....	182
APPENDIX I.....	184
APPENDIX J.....	198
APPENDIX K.....	200

LIST OF TABLES

Table 1: Parameters used in the Cho [9] example problem	47
Table 2: Parameters used in the modified Higashi and Pigford [24] example problem as given by van Genuchten [45].....	50
Table 3: Parameters used in the new 10 species example problem	52
Table 4: Summary of parameter limitations	58
Table5: Parameters used in the Domenico and Robbins [17] example problem	74

LIST OF FIGURES

Fig. 1: Comparison of the concentration profiles of the general solution with the Cho [9] solution (3 Species, Dirichlet boundary condition)	48
Fig. 2: Comparison of the concentration profiles of the general solution with the van Genuchten [45] solution (4 Species, Cauchy boundary condition)	51
Fig. 3: Comparison of the concentration profiles of the general solution with the Quezada et al. [38] semi-analytical solution (10 Species, Dirichlet boundary condition)	53
Fig. 4: Comparison of the concentration profiles of the general solution with the Quezada et al. [38] semi-analytical solution (10 Species, Cauchy boundary condition)	54
Fig. 5: Concentration contours predicted by the Domenico and Wexler solutions for the base case: solutions behind the advective front for (a) X-Y plane (b) X-Z plane	75
Fig. 6: Sensitivity results for variations in the longitudinal dispersivity value; solutions behind the advective front for (a) α_x*10 (b) $\alpha_x/10$	77
Fig. 7: Sensitivity results for variations in the transport velocity; solutions behind the advective front for (a) $v*10$ (b) $v/10$	79
Fig. 8: Concentration contours predicted by the Domenico and Wexler solutions; solutions include concentration contours beyond the advective front	80
Fig. 9: Sensitivity results for variations in the longitudinal dispersivity value; solutions include concentration contours beyond the advective front for (a) α_x*10 (b) $\alpha_x/10$	82

Fig. 10: Sensitivity results for variations in the transport velocity; solutions include concentration contours beyond the advective front for (a) $v*10$ (b) $v/10$ 83

Fig. 11: Concentration profiles predicted by the Domenico solution compared with the Wexler solution at $x = 1000\text{m}$ and 1500m (a) along Y axis (b) along Z axis..... 86

CHAPTER I

INTRODUCTION

1.1 Background

Transport problems involving sequentially decaying contaminants are frequently analyzed by groundwater hydrologists to assess water quality issues associated with environmental and health hazards. Examples of sequentially decaying contaminants include radioactive waste materials, chlorinated solvents, and nitrogenous species [4, 11, 45]. Several types of models, using both analytical and numerical procedures, have been formulated for solving these sequentially coupled reactive transport problems [12, 26]. Although numerical models are capable of solving complex and heterogeneous problems, their performance often needs to be tested against experimental datasets or analytical models. Experimental simulations of complex reactive transport problems are not only time consuming but can also be expensive. Therefore, analytical models provide a convenient, cost-effective alternative to test and validate numerical formulations [18, 30, 44]. Furthermore, analytical models also provide computationally efficient screening tools for simulating the fate and transport of reactive contaminants in groundwater systems [3, 13].

1.2 Literature Review

The analytical solution given by McLaren [32] and McLaren [33], which describes the steady-state, one-dimensional transport of a five species nitrogen chain, is one of the first multi-species solutions derived for solving sequentially coupled reactive transport problems. This work assumed that the transport was only governed by advection, and the effects of dispersion and sorption were ignored.

Cho [9] developed explicit analytical solutions to a three species transport problem that was subjected to advection, dispersion, linear equilibrium sorption, coupled through sequential first-order reactions. Explicit analytical solutions were obtained using Laplace transform procedures for the Dirichlet boundary condition. One of the limitations of this solution is that only the first species in the chain was subjected to sorption. Later, Misra et al. [34] derived semi-analytical solutions to a problem similar to Cho [9] using a pulse source boundary.

Burkholder and Rosinger [7] and Lester et al. [29] developed solutions for the advective dispersive transport of radionuclide chains subjected to linear equilibrium sorption. Explicit analytical solutions were presented for a three species problem involving distinct retardation factors for each species, for both impulse and decaying-band release boundary conditions. In addition, they also provided solutions for the case of no dispersion and for the case of identical retardation factors.

Harada et al. [23] published a research report presenting general semi-analytical solutions to sequentially coupled one dimensional reactive transport problems of arbitrary chain lengths subjected to arbitrary release modes. However, one of the major limitations of the solution strategy was that, the semi-analytical solution for a given species in the

chain required the computation of its entire predecessor species. This would result in computationally inefficient algorithms especially when analyzing transport problems involving long reactive chains. Harada et al. [23] and Higashi and Pigford [24] also provided explicit closed-form solutions for a set of purely advective (no dispersion) transport problems with various types of boundary conditions.

Gurehian and Jansen [21] presented an analytical solution to a transport problem involving a three member, first-order decay chain in a multi-layered system, subjected to advection, dispersion and linear equilibrium sorption processes for both continuous and band source release conditions. Convolution theorems and Laplace transform techniques were used to obtain semi-analytical solutions for the case involving both advective and dispersive transport, and explicit closed-form analytical solutions for the case involving non-dispersive transport.

van Genuchten [45] developed explicit analytical solutions to model a sequentially coupled four species transport problem governed by advection, dispersion and linear equilibrium sorption processes involving, first-order reactions. It was assumed that all the species had distinct retardation factors. One of the key contributions of this work is that it considered both Dirichlet and Cauchy boundary conditions. Furthermore, van Genuchten [45] developed a robust computer code (CHAIN) for implementing his analytical solution.

Angelakis et al. [2] developed a semi-analytical solution to a sequentially coupled two-species transport problem governed by advection, dispersion and linear equilibrium sorption subjected to Dirichlet boundary condition. The transport problem assumed that the reactions were first-order and each of the species had different dispersion coefficients

and distinct retardation factors. The authors also demonstrated that when the dispersion coefficients of both the species were equal, their solution reduced to the closed-form solution similar to the solutions presented by Cho [9] and Misra et al. [34]. Furthermore, the authors also provided solutions for the no dispersion (pure advection) case. Angelakis et al. [1] developed an interesting semi-analytical solution for a problem involving the coupled transport of two solutes and a gaseous product in soils. The solute migration was governed by advection, dispersion, linear equilibrium sorption and sequential first-order reaction, whereas the gas migration was governed by diffusive transport coupled with reversible linear equilibrium dissolution.

Lunn et al. [30] solved a three-species transport problem, which was similar to the Cho [9] problem, using the Fourier transform method. The authors demonstrated that the use of Fourier transforms enabled them to solve problems having non-zero initial conditions by solving two special case problems.

Khandelwal and Rabideau [27] developed semi-analytical solutions for a three species, sequentially-coupled, first-order reactive transport problem. The key contribution of this work was that they addressed cases involving linear, non-equilibrium sorption mechanisms.

Eykholt and Li [18] developed a solution method based on kinetic response functions to solve a linearly coupled non-sequential reactive transport problem having different retardation factors. Although, there was no restriction on the number of species in the system, this method required numerical procedures to evaluate the final solution. Furthermore, for the case of the non-ideal plug flow scenarios (advective dispersive

transport), the accuracy of this method appears to decrease with decrease in Peclet number.

Sun et al. [42] developed a method that can solve multi-species advective dispersive transport equations coupled with sequential first-order reactions involving arbitrary number of species for different types of initial and boundary conditions. Their method was based on the use of a transformation procedure to uncouple the system of equations, which could then be solved analytically in the transformed domain. The final solutions are obtained by retransforming the solutions to the original domain. Later, Sun et al. [43] extended the transformation format to solve problems involving a combination of serial and parallel reactions. Clement [11] presented a more general and fundamental approach to derive the Sun et al. [43] solution by employing the similarity transformation method. The approach presented by Clement [11] can also be used to solve problems involving serial, parallel, converging, diverging and/or reversible first-order reaction network. However, all of these methods are only applicable for solving problems involving identical retardation factors.

Bauer et al. [4] presented a method to solve one, two, and three-dimensional sequentially coupled reactive transport problems with distinct retardation factors. This method was based on transforming the system of equations to a Laplace domain and then obtaining a set of fundamental solutions to each of the equations in the transformed domain. The specific solutions in the Laplace domain can then be obtained through a linear combination of the fundamental solutions provided the fundamental solutions are linearly independent. Finally, the Laplace domain solutions can be transformed back to the time domain using the inverse Laplace transform procedure, which could be

accomplished either analytically or numerically. Although this method can be applied to solve different types of boundary conditions, the solution procedure is mathematically tedious; specifically obtaining analytical inverse transform expressions for long chain lengths can be a challenge.

Montas [35] developed an analytical procedure to solve a three species, multi-dimensional transport problem coupled by a first-order, non-sequential reaction network subject to a pulse type boundary condition. This procedure involved obtaining a basis solution of a convoluted form for the transport equation and then evaluating the basis solution using Laplace transforms. One of the key advantages of this procedure is that it can model transport problems with distinct retardation factors. However, as mentioned earlier this solution was limited to a three species system.

Quezada et al. [38] extended the approach given by Clement [11] and developed a method that can solve multi-species transport equations coupled with a network of first-order reactions involving distinct retardation factors. This method involves transforming the system of governing equations to a Laplace domain and then solving the transformed system of equations using the Clement [11] approach. The solutions in the Laplace domain are then retransformed to the time domain using an inverse Laplace transform procedure. One of the key limitations of this approach is that, except for a simple two species transport problem, the solutions are in general semi-analytical since they require a numerical inverse Laplace transform routine to evaluate them.

1.3 Scope and Objective

The above literature review indicates that one-dimensional reactive transport equations coupled through sorption and sequential first-order reactions have explicit closed-form analytical solutions only for short chains up to four species. To model transport problems involving longer reaction chains, one has to either use semi-analytical solutions or purely numerical solutions.

This objective of this work is to develop a general closed-form analytical solution to the sequential transport problem involving arbitrary number of species subjected to a generic exponentially decaying Bateman-type source boundary, for a spatially varying initial condition. The solutions derived in this study are then implemented in a computational platform and tested and validated against other analytical and semi-analytical solutions published in the literature.

1.4 Organization

This thesis is organized into five chapters. Chapter I presents a brief introduction with a detailed literature review of analytical solutions to coupled reactive transport problems.

Chapter II details the solution derivation of the sequentially coupled one dimensional reactive transport problem. Chapter III discusses some of the techniques in implementing the new solution on a computational platform and testing the new solution through a set of example problems. Chapters II and III are draft versions of manuscripts that have been submitted to *Advances in Water Resources* journal.

Chapter IV investigates the validity of approximations involved in extending the one-dimensional solutions derived in this study into three-dimensions. The contents of this chapter are extracted from the author's publication titled "Domenico Solution – Is it Valid?" in the Groundwater journal.

Chapter V summarizes the key findings of this study, and also discusses some recommendations for future work.

CHAPTER II

GOVERNING EQUATIONS AND SOLUTION DERIVATION

2.1 Governing Equations

Consider, a one-dimensional transport problem involving ‘n’ sequentially decaying contaminants simultaneously subjected to advection, dispersion and linear adsorption processes. The general governing equation for this transport problem can be expressed as:

$$\begin{aligned} R_i \frac{\partial c_i(x,t)}{\partial t} + v \frac{\partial c_i(x,t)}{\partial x} - D_x \frac{\partial^2 c_i(x,t)}{\partial x^2} &= y_i k_{i-1} c_{i-1}(x,t) - k_i c_i(x,t) ; \forall i = 2, 3, \dots n \\ &= -k_i c_i(x,t) ; i = 1 \\ &; \forall t > 0 \text{ and } 0 < x < \infty \end{aligned} \quad (1)$$

where; ‘ c_i ’ is the concentration of species i [ML^{-3}]; ‘ R_i ’ is the retardation coefficient of species i [Dimensionless]; ‘ y_i ’ is the effective yield factor that describes the mass of a species i produced from species $i-1$ [MM^{-1}]; ‘ k_i ’ is the first-order decay rate constant of species i [T^{-1}]; ‘ v ’ is the transport velocity [LT^{-1}]; ‘ D_x ’ is the dispersion coefficient [L^2T^{-1}]; and ‘ n ’ is the total number of species in the reaction network. Equation(1) is solved for a generic exponentially distributed initial condition given by:

$$c_i(x,0) = c_i^0 e^{-\mu_i x}, \quad 0 < x < \infty ; \forall i = 1, 2, \dots n \quad (2)$$

where; ‘ c_i^o ’ is the initial source concentration of species i [ML^{-3}]; ‘ μ_i ’ is the first-order decay parameter of the initial distribution of species i [L^{-1}]. The boundary condition at ‘ ∞ ’ is given as:

$$\frac{\partial c_i(\infty, t)}{\partial x} = 0, t > 0 ; \forall i = 1, 2, \dots n \quad (3)$$

Explicit solutions including detailed derivation steps are provided for the following two inlet (source) boundary conditions: Dirichlet (*Section-3*) and Cauchy (*Section-4*) boundaries.

2.2 Derivation of the Solution for the Dirichlet Boundary condition

For the case of the Dirichlet boundary, the boundary condition at the source is described as follows:

$$c_i(0, t) = \begin{cases} \sum_{i=1}^i B_i^i e^{-\lambda_i t}, & 0 < t \leq t_o \\ 0, & t > t_o \end{cases} ; \forall i = 1, 2, \dots n \quad (4)$$

where; ‘ B_i^i ’ is the source boundary concentration of specie i_1 that contributes to species i [ML^{-3}]; ‘ λ_{i_1} ’ is the first-order decay of the corresponding ‘ B_i^i ’ term [T^{-1}].

Equation(4) can be conveniently re-written as:

$$c_i(0, t) = \sum_{i=1}^i B_i^i e^{-\lambda_{i_1} t} \{u(t) - u(t - t_o)\}, t > 0 ; \forall i = 1, 2, \dots n$$

where ; u is the unit step function given by

$$u(t - a) = \begin{cases} 0, & \text{if } t < a \\ 1, & \text{if } t \geq a \end{cases} \quad (5)$$

and a is an arbitrary positive constant

The system of equations given by (1) can be written in a matrix format as [11, 38]:

$$[R] \frac{\partial \{c\}}{\partial t} + v \frac{\partial \{c\}}{\partial x} - D_x \frac{\partial^2 \{c\}}{\partial x^2} = [K] \{c\} \quad (6)$$

where; ‘ $[]$ ’ denotes a square matrix and ‘ $\{ \}$ ’ denotes a column vector. The corresponding initial and boundary conditions can be written as:

$$\{c(x, 0)\} = \{c^o e^{-\mu x}\}, \quad 0 < x < \infty \quad (7)$$

$$\left\{ \frac{\partial c(\infty, t)}{\partial x} \right\} = 0, \quad t > 0 \quad (8)$$

$$\{c(0, t)\} = \{\omega\}, \quad t > 0$$

$$\text{where ; } \omega_i = \sum_{i=1}^i B_i^i e^{-\lambda_i t} \{u(t) - u(t - t_o)\}, \quad t > 0 ; \quad \forall i = 1, 2, \dots, n \quad (9)$$

The solution procedure used here is adopted from Quezada [38]. Applying Laplace transform to equation(6), we get:

$$[R]s \{p\} - [R] \{c(x, 0)\} + v \frac{d \{p\}}{dx} - D_x \frac{d^2 \{p\}}{dx^2} = [K] \{p\} \quad (10)$$

where; ‘ s ’ is the Laplace variable and ‘ p ’ is the Laplace transformed concentration.

Substituting equation(7) in (10) and rearranging we get:

$$\frac{d^2 \{p\}}{dx^2} - \left(\frac{v}{D_x} \right) \frac{d \{p\}}{dx} + \frac{1}{D_x} ([K] - [R]s) \{p\} = \frac{-1}{D_x} [R] \{c^o e^{-\mu x}\} \quad (11)$$

Now in order to uncouple the system of ordinary differential equations (ODEs) given by equation(11), we apply the linear transform procedure described by Clement [11], by performing the following matrix operation.

$$\{p\} = [A]\{b\} \quad (12)$$

where; ' $\{b\}$ ' is the concentration in the doubly transformed domain and $[A]$ is an arbitrary square matrix of order n . Applying this transformation equation(11) gets modified as:

$$\frac{\partial^2 [A]\{b\}}{\partial x^2} - \left(\frac{v}{D_x}\right) \frac{\partial [A]\{b\}}{\partial x} + \frac{1}{D_x} ([K] - [R]s)[A]\{b\} = \frac{-1}{D_x} [R]\{c^o e^{-\mu x}\} \quad (13)$$

Pre-multiplying equation(13) with ' $[A]^{-1}$ ', we get:

$$\frac{\partial^2 \{b\}}{\partial x^2} - \left(\frac{v}{D_x}\right) \frac{\partial \{b\}}{\partial x} + \frac{1}{D_x} [\tilde{K}]\{b\} = \frac{-1}{D_x} \{\tilde{C}\} \quad (14)$$

where ; $[\tilde{K}] = [A]^{-1} ([K] - [R]s)[A]$ and $\{\tilde{C}\} = [A]^{-1} [R]\{c^o e^{-\mu x}\}$

By forcing the columns of the ' $[A]$ ', matrix as the eigenvectors of the combined reaction coefficient matrix ' $\langle [K] - [R]s \rangle$ ', we can make the ' $[\tilde{K}]$ ' matrix a diagonal matrix and thus uncouple the system of equations; the details of this similarity transformation procedure are illustrated in Clement [11]. The corresponding ' $[A]$ ' matrix is:

$$[A] = \begin{bmatrix} \prod_{i=2}^n \frac{-(k_1 + sR_1 - k_i - sR_i)}{y_i k_{i-1}}, 0, 0, \dots, (n-1) \text{ times} \\ \prod_{i=3}^n \frac{-(k_1 + sR_1 - k_i - sR_i)}{y_i k_{i-1}}, \prod_{i=3}^n \frac{-(k_2 + sR_2 - k_i - sR_i)}{y_i k_{i-1}}, 0, 0, \dots, (n-2) \text{ times} \\ \cdot \\ \cdot \\ \prod_{i=n}^n \frac{-(k_1 + sR_1 - k_i - sR_i)}{y_i k_{i-1}}, \prod_{i=n}^n \frac{-(k_2 + sR_2 - k_i - sR_i)}{y_i k_{i-1}}, \dots, \prod_{i=n}^n \frac{-(k_{n-1} + sR_{n-1} - k_i - sR_i)}{y_i k_{i-1}}, 0 \\ 1, 1, \dots, n \text{ times} \end{bmatrix} \quad (15)$$

13

The ' $[A]^{-1}$ ', matrix is:

$$[A]^{-1} = \left[\begin{array}{c} \frac{\prod_{i=2}^n y_i k_{i-1}}{\prod_{i=1, (i \neq 1)}^n -(k_1 + sR_1 - k_i - sR_i)}, 0, 0, \dots, (n-1) \text{ times} \\ \frac{\prod_{i=2}^n y_i k_{i-1}}{\prod_{i=1, (i \neq 2)}^n -(k_2 + sR_2 - k_i - sR_i)}, \frac{\prod_{i=3}^n y_i k_{i-1}}{\prod_{i=2, (i \neq 2)}^n -(k_2 + sR_2 - k_i - sR_i)}, 0, 0, \dots, (n-2) \text{ times} \\ \cdot \\ \cdot \\ \frac{\prod_{i=2}^n y_i k_{i-1}}{\prod_{i=1, (i \neq n)}^n -(k_n + sR_n - k_i - sR_i)}, \frac{\prod_{i=3}^n y_i k_{i-1}}{\prod_{i=2, (i \neq n)}^n -(k_n + sR_n - k_i - sR_i)}, \dots, \frac{\prod_{i=n}^n y_i k_{i-1}}{\prod_{i=n-1, (i \neq n)}^n -(k_n + sR_n - k_i - sR_i)}, 1 \end{array} \right] \quad (16)$$

The corresponding ‘ $\left[\tilde{K} \right]$ ’, matrix is:

$$\left[\tilde{K} \right] = \begin{bmatrix} -k_1 - sR_1, 0, 0, \dots (n-1) \text{ times} \\ 0, -k_2 - sR_2, 0, 0, \dots (n-2) \text{ times} \\ \cdot \\ 0, 0, \dots (i-1) \text{ times}, -k_i - sR_i, 0, 0, \dots (n-i) \text{ times} \\ \cdot \\ 0, 0, \dots (n-1) \text{ times}, -k_n - sR_n \end{bmatrix} \quad (17)$$

The corresponding ‘ $\left\{ \tilde{C} \right\}$ ’, vector is:

$$\left\{ \tilde{C} \right\} = \left\{ \begin{array}{l} \frac{R_1 c_1^o e^{-\mu_1 x} \prod_{i=2}^n y_i k_{i-1}}{\prod_{i=1, (i \neq 1)}^n -(k_1 + sR_1 - k_i - sR_i)} \\ \frac{R_1 c_1^o e^{-\mu_1 x} \prod_{i=2}^n y_i k_{i-1}}{\prod_{i=1, (i \neq 2)}^n -(k_2 + sR_2 - k_i - sR_i)} + \frac{R_2 c_2^o e^{-\mu_2 x} \prod_{i=3}^n y_i k_{i-1}}{\prod_{i=2, (i \neq 2)}^n -(k_2 + sR_2 - k_i - sR_i)} \\ \cdot \\ \cdot \\ \frac{R_1 c_1^o e^{-\mu_1 x} \prod_{i=2}^n y_i k_{i-1}}{\prod_{i=1, (i \neq n)}^n -(k_n + sR_n - k_i - sR_i)} + \frac{R_2 c_2^o e^{-\mu_2 x} \prod_{i=3}^n y_i k_{i-1}}{\prod_{i=2, (i \neq n)}^n -(k_n + sR_n - k_i - sR_i)} + \dots + R_n c_n^o e^{-\mu_n x} \end{array} \right\} \quad (18)$$

The explicit expression for ‘ \tilde{C}_i ’ in equation(18) is:

$$\tilde{C}_i = \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o e^{-\mu_{i_1} x} \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right]; \forall i = 1, 2, \dots, n \quad (19)$$

Equation(14) describes a set of ‘ n ’ independent second-order non-homogeneous ODEs the boundary conditions of which are obtained by performing Laplace and linear transforms of the boundary conditions given by equations(8) and (9). Laplace transform of equations(8) and (9) yields:

$$\left\{ \frac{\partial p(\infty, s)}{\partial x} \right\} = 0 \quad (20)$$

$$\{p(0, s)\} = \{\xi\}$$

$$\text{where ; } \xi_i = \sum_{i_1=1}^i \frac{B_{i_1}^{i_1} \{1 - e^{-t_0(s+\lambda_{i_1})}\}}{(s + \lambda_{i_1})}; \forall i = 1, 2, \dots, n \quad (21)$$

To transform the boundary conditions from ‘ p ’ domain to the ‘ b ’ domain, we apply the linear transform given by equation(12). This yields:

$$\left\{ \frac{\partial b(\infty, s)}{\partial x} \right\} = 0 \quad (22)$$

$$\{b(0, s)\} = [A]^{-1} \{\xi\} \quad (23)$$

The explicit expression for ‘ $b_i(0, s)$ ’ in equation(23) is given as:

$$b_i(0, s) = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \sum_{i_2=1}^{i_1} \frac{B_{i_2}^{i_2} \{1 - e^{-t_0(s+\lambda_{i_2})}\}}{(s + \lambda_{i_2})} \right] \quad (24)$$

$$; \forall i = 1, 2, \dots, n$$

Since equation(14) is uncoupled, it can now be written as a set of ‘ n ’ independent equations as:

$$\frac{\partial^2 b_i(x,s)}{\partial x^2} - \left(\frac{v}{D_x}\right) \frac{\partial b_i(x,s)}{\partial x} + \frac{1}{D_x} (-k_i - sR_i) b_i(x,s) = \frac{-1}{D_x} \sum_{i_2=1}^i \left[\frac{R_{i_2} c_{i_2}^o e^{-\mu_{i_2} x} \prod_{i_2=i_2+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_2, (i_2 \neq i)}^n -(k_{i_2} + sR_{i_2} - k_{i_2-1} - sR_{i_2-1})} \right] ; \forall i = 1, 2, \dots, n \quad (25)$$

The general solution to equation(25) is given as:

$$b_i(x,s) = b_i^h(x,s) + b_i^p(x,s) ; \forall i = 1, 2, \dots, n \quad (26)$$

where; ‘ $b_i^h(x,s)$ ’ is the general solution of the homogeneous part of equation(25) and ‘ $b_i^p(x,s)$ ’ is a particular solution of equation(25). The general solution ‘ $b_i^h(x,s)$ ’ can be readily obtained as:

$$b_i^h(x,s) = \Psi_i^1 e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} + \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right]} + \Psi_i^2 e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right]} ; \forall i = 1, 2, \dots, n \quad (27)$$

where; Ψ_i^1 and Ψ_i^2 are constants. The particular solution ‘ $b_i^p(x,s)$ ’ is obtained by using the method of undetermined coefficients. The general form of the particular solution is given as:

$$b_i^p(x,s) = \frac{-1}{D_x} \sum_{i_2=1}^i \left[\frac{M_{i_2} R_{i_2} c_{i_2}^o e^{-\mu_{i_2} x} \prod_{i_2=i_2+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_2, (i_2 \neq i)}^n -(k_{i_2} + sR_{i_2} - k_{i_2-1} - sR_{i_2-1})} \right] \quad (28)$$

where; ‘ M_{i_1} ’ is a constant. Substituting equation(28) in the governing equation(25) and simplifying, we evaluate the constants ‘ M_{i_1} ’ as:

$$M_{i_1} = \frac{D_x}{(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_i - sR_i)} \quad (29)$$

; $\forall i = 1, 2, \dots, n$

Substituting the values of ‘ M_{i_1} ’ into equation(28) we get the particular solution ‘ $b_i^p(x, s)$ ’ to equation(25) as:

$$b_i^p(x, s) = - \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o e^{-\mu_{i_1} x} \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_i - sR_i) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right] \quad (30)$$

; $\forall i = 1, 2, \dots, n$

Substituting equations(27) and (30) in equation(26), we get the general solution to equation(25) as:

$$b_i(x, s) = \Psi_i^1 e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} + \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right]} + \Psi_i^2 e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right]}$$

$$- \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o e^{-\mu_{i_1} x} \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_i - sR_i) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right] \quad (31)$$

; $\forall i = 1, 2, \dots, n$

In order to apply the boundary condition given by equation(22) we differentiate the general solution with respect to x. Differentiation of equation(31) with respect to x yields:

$$\begin{aligned}
\frac{\partial b_i(x,s)}{\partial x} = & \Psi_i^1 \left[\frac{1}{2} \left\{ \frac{v}{D_x} + \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right] e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} + \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right]} \\
& + \Psi_i^2 \left[\frac{1}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right] e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right]} \\
& - \sum_{i_1=1}^i \left[\frac{(-\mu_{i_1}) R_{i_1} c_{i_1}^o e^{-\mu_{i_1} x} \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_{i_1} - sR_{i_1}) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_{i_1} + sR_{i_1} - k_{i_2} - sR_{i_2})} \right] \\
& ; \forall i = 1, 2, \dots, n
\end{aligned} \tag{32}$$

To satisfy the boundary condition given by equation(22), i.e. when ‘ x ’ tends to ‘ ∞ ’; the exponential function in the first term tends to ‘ ∞ ’, hence Ψ_i^1 must vanish.

Equation(32) now reduces to:

$$\begin{aligned}
b_i(x,s) = & \Psi_i^2 e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right]} \\
& - \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o e^{-\mu_{i_1} x} \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_{i_1} - sR_{i_1}) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_{i_1} + sR_{i_1} - k_{i_2} - sR_{i_2})} \right] \\
& ; \forall i = 1, 2, \dots, n
\end{aligned} \tag{33}$$

Applying the second boundary condition given by equation(24), we get:

$$\Psi_i^2 = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_o(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})} \right] + \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_i - sR_i \right) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right]; \quad \forall i = 1, 2, \dots, n \quad (34)$$

Therefore, the solution in the ‘ b ’ domain is:

$$b_i(x, s) = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_o(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})} \right] e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_i + sR_i)} \right\} \right]} + \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1} \left\{ e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_i + sR_i)} \right\} \right]} - e^{-\mu_{i_1} x} \right\}}{\left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_i - sR_i \right) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right] ; \quad \forall i = 1, 2, \dots, n \quad (35)$$

Inverse linear transform of equation(35) is done to obtain the solution in the Laplace domain (‘ p ’ domain) by using equation(12). The solution given by equation(35) can be split into two parts and represented as:

$$b_i(x, s) = b_i^1(x, s) + b_i^2(x, s)$$

where ;

$$b_i^1(x, s) = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_0(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})} \right] e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_i + sR_i)} \right\} \right]}$$

$$b_i^2(x, s) = \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1} \left\{ e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_i + sR_i)} \right\} \right]} - e^{-\mu_{i_1} x} \right\}}{\left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_i - sR_i \right) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right]$$

; $\forall i = 1, 2, \dots, n$
(36)

Using the distributive property of matrix addition, we can apply the inverse linear transform to each of the individual terms and then sum them to get the solution in the ‘ p ’ domain. This is expressed as:

$$\{p\} = [A]\{b\} = [A]\{b^1\} + [A]\{b^2\} \quad (37)$$

The first term ‘ $[A]\{b^1\}$ ’, can be evaluated as:

$$\{p^1\} = [A]\{b^1\} \quad (38)$$

The explicit expression for ‘ $p_i^1(x, s)$ ’ in equation(38) is given as:

$$p_i^1(x,s) = \sum_{i_1=1}^i \left[\frac{\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=1}^{i_1} \frac{B_{i_2}^{i_2} \left\{ 1 - e^{-t_0(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})}}{\sum_{i_2=i_1}^i \frac{e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]}}{\prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})}} \right] \quad (39)$$

; $\forall i = 1, 2, \dots, n$

Using a similar approach the second term ‘ $[A]\{b^2\}$ ’ is evaluated and the explicit expression for ‘ $p_i^2(x,s)$ ’ is:

$$p_i^2(x,s) = \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left\{ e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]} - e^{-\mu_{i_1} x} \right\}}{\sum_{i_2=i_1}^i \frac{(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_{i_2} - sR_{i_2})}{\prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})}} \right] \quad (40)$$

; $\forall i = 1, 2, \dots, n$

Substituting equations(39) and (40) into equation(37) we get the solution in the Laplace domain as:

$$\begin{aligned}
p_i(x, s) &= p_i^1(x, s) + p_i^2(x, s) = \\
&\sum_{i_1=1}^i \left[\frac{\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_0(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})}}{\sum_{i_2=i_1}^i \frac{e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]}}{\prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})}} \right] \\
&+ \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left\{ e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]} - e^{-\mu_{i_1} x} \right\}}{\sum_{i_2=i_1}^i \left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_{i_2} - sR_{i_2} \right) \prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})} \right] \\
&\quad ; \forall i = 1, 2, \dots, n
\end{aligned} \tag{41}$$

The final solution is obtained by taking an inverse Laplace transform of the solution given by equation(41). Inverse Laplace transform is performed as follows:

$$\begin{aligned}
c_i(x, t) &= c_i^1(x, t) + c_i^2(x, t) \\
&= \ell^{-1} \left\langle p_i^1(x, s) + p_i^2(x, s) \right\rangle \\
&= \ell^{-1} \left\langle p_i^1(x, s) \right\rangle + \ell^{-1} \left\langle p_i^2(x, s) \right\rangle \\
&\quad ; \forall i = 1, 2, \dots, n
\end{aligned} \tag{42}$$

In Appendix A, the terms ‘ $\ell^{-1} \left\langle p_i^1(x, s) \right\rangle$ ’, and ‘ $\ell^{-1} \left\langle p_i^2(x, s) \right\rangle$ ’, are evaluated.

Substituting equations (A.7) and (A.16) in equation(42) we obtain the final solution in the time domain as:

$$\begin{aligned}
c_i(x, t) = & \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \{ G_1^1 + h(G_1^1) G_2^1 \} \right] \\
& + \sum_{i_1=1}^i \left[R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \{ G_1^2 + h(G_1^2) G_2^2 \} \right] \quad (43) \\
& ; \quad \forall i = 1, 2, \dots, n
\end{aligned}$$

where; the ‘ G ’ terms are defined as (See equations (A.7) and (A.16) in Appendix

A).

$$\begin{aligned}
G_1^1 &= \sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i \frac{B_{i_1}^{i_3} \left\langle \begin{aligned} &F_{i_2, i_3, 0}[x, t] - u(t - t_o) e^{(-\lambda_{i_3} t_o)} F_{i_2, i_3, 0}[x, (t - t_o)] \\ &- F_{i_2, i_2, i_4}[x, t] + u(t - t_o) e^{(-\lambda_{i_3} t_o)} F_{i_2, i_2, i_4}[x, (t - t_o)] \end{aligned} \right\rangle}{(a_{i_2, i_4} - \lambda_{i_3}) \left(\prod_{i_5=i_1, (i_5 \neq i_2, R_{i_5} = R_{i_2})}^i -k_{i_2, i_5} \right) (-R_{i_2, i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2, i_5} (a_{i_2, i_5} - a_{i_2, i_4})} \\
G_2^1 &= \frac{B_{i_1}^{i_3} \left\langle F_{i_2, i_3, 0}[x, t] - u(t - t_o) e^{(-\lambda_{i_3} t_o)} F_{i_2, i_3, 0}[x, (t - t_o)] \right\rangle}{\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4}} \\
G_1^2 &= \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i \frac{\left\langle F_{i_2, i_1, -i_2}[x, t] - e^{(-\mu_{i_1} x - a_{i_1, -i_2} t)} - F_{i_2, i_2, i_3}[x, t] + e^{(-\mu_{i_1} x - a_{i_2, i_3} t)} \right\rangle}{(a_{i_2, i_3} - a_{i_1, -i_2}) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (a_{i_2, i_4} - a_{i_2, i_3})} \\
G_2^2 &= \frac{-\left\langle F_{i_2, i_1, -i_2}[x, t] - e^{(-\mu_{i_1} x - a_{i_1, -i_2} t)} \right\rangle}{R_{i_2} \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3}}
\end{aligned} \tag{44}$$

where; the term ‘ F_{i_1, i_2, i_3} ’ is given by: (See equation(B.8) in Appendix B)

$$F_{i_1, i_2, i_3} [x, t] = \frac{e^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}}}{2} \left[e^{\frac{-x\omega_{i_1, i_2, i_3}}{2D_x}} \operatorname{erfc} \left\{ \frac{R_{i_1} x - \omega_{i_1, i_2, i_3} t}{2\sqrt{R_{i_1} D_x t}} \right\} + e^{\frac{x\omega_{i_1, i_2, i_3}}{2D_x}} \operatorname{erfc} \left\{ \frac{R_{i_1} x + \omega_{i_1, i_2, i_3} t}{2\sqrt{R_{i_1} D_x t}} \right\} \right] \quad (45)$$

$$\text{where ; } \omega_{i_1, i_2, i_3} = \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)}$$

The above solution is valid only for real values of ‘ ω_{i_1, i_2, i_3} ’. For problems involving complex values for ‘ ω_{i_1, i_2, i_3} ’ the ‘ F_{i_1, i_2, i_3} ’ terms are given as: (See equation(B.14) in Appendix B)

$$F_{i_1, i_2, i_3} [x, t] = e^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}} \left[A \operatorname{Cos} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) - B \operatorname{Sin} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) \right]$$

$$\text{where ; } \omega_{i_1, i_2, i_3}^* = \sqrt{\left| v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right) \right|} \quad (46)$$

$$\text{and } (A + iB) = \operatorname{erfc} \left\{ \frac{R_{i_1} x + i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\}$$

2.3 Derivation of the Analytical Solution for the Cauchy Boundary Condition

For the case of the Cauchy boundary, the boundary condition at the source is

described as follows:
$$-D_x \frac{\partial c_i(0,t)}{\partial x} + v c_i(0,t) = \begin{cases} \sum_{i=1}^i B_i^i v e^{-\lambda_i t}, & 0 < t \leq t_o \\ 0, & t > t_o \end{cases}; \forall i = 1, 2, \dots, n$$

(47)

Equation(47) can be conveniently re-written as:

$$-D_x \frac{\partial c_i(0,t)}{\partial x} + v c_i(0,t) = \sum_{i=1}^i B_i^i v e^{-\lambda_i t} \{u(t) - u(t - t_o)\}, \quad t > 0; \forall i = 1, 2, \dots, n$$

where ; u is the unit step function given by

$$u(t-a) = \begin{cases} 0, & \text{if } t < a \\ 1, & \text{if } t \geq a \end{cases} \quad (48)$$

and a is an arbitrary positive constant

Note that the governing equations, initial conditions and the boundary condition at ‘ ∞ ’ for the Cauchy boundary are identical to the Dirichlet boundary. Furthermore, the boundary conditions at the source for both these boundaries share a similar structure. Due to this structural similarity, the solution procedures for the Cauchy boundary will be analogous to that of the Dirichlet boundary. The details of the solution derivation are presented in Appendix C. The final solution for the Cauchy boundary can also be represented by equation(43). However, the ‘ G ’ terms associated with the Cauchy boundary are defined as (See equations (D.6) and (D.13) in Appendix D):

$$\begin{aligned}
G_1^1 &= \sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i \frac{B_{i_1}^{i_3} \left\langle \begin{aligned} &F_{i_2, i_3, 0}[x, t] - u(t-t_o) e^{(-\lambda_{i_3} t_o)} F_{i_2, i_3, 0}[x, (t-t_o)] \\ &- F_{i_2, i_2, i_4}[x, t] + u(t-t_o) e^{(-\lambda_{i_3} t_o)} F_{i_2, i_2, i_4}[x, (t-t_o)] \end{aligned} \right\rangle}{(a_{i_2, i_4} - \lambda_{i_3}) \left(\prod_{i_5=i_1, (i_5 \neq i_2, R_{i_5} = R_{i_2})}^i -k_{i_2, i_5} \right) (-R_{i_2, i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2, i_5} (a_{i_2, i_5} - a_{i_2, i_4})} \\
G_2^1 &= \frac{B_{i_1}^{i_3} \left\langle F_{i_2, i_3, 0}[x, t] - u(t-t_o) e^{(-\lambda_{i_3} t_o)} F_{i_2, i_3, 0}[x, (t-t_o)] \right\rangle}{\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4}} \\
G_1^2 &= \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i \frac{\left\langle \left(1 + \frac{\mu_{i_1} D_x}{\nu} \right) F_{i_2, i_1, -i_2}[x, t] - e^{(-\mu_{i_1} x - a_{i_1, -i_2} t)} - \left(1 + \frac{\mu_{i_1} D_x}{\nu} \right) F_{i_2, i_2, i_3}[x, t] + e^{(-\mu_{i_1} x - a_{i_2, i_3} t)} \right\rangle}{(a_{i_2, i_3} - a_{i_1, -i_2}) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (a_{i_2, i_4} - a_{i_2, i_3})} \\
G_2^2 &= \frac{-\left\langle \left(1 + \frac{\mu_{i_1} D_x}{\nu} \right) F_{i_2, i_1, -i_2}[x, t] - e^{(-\mu_{i_1} x - a_{i_1, -i_2} t)} \right\rangle}{R_{i_2} \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3}}
\end{aligned} \tag{49}$$

where; the term ‘ F_{i_1, i_2, i_3} ’ is given by: (See equation (E.20) in Appendix E)

$$\begin{aligned}
F_{i_1, i_2, i_3} [x, t] &= ve^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}} \left[\frac{e^{-\frac{x\omega_{i_1, i_2, i_3}}{2D_x}}}{(v + \omega_{i_1, i_2, i_3})} \operatorname{erfc} \left\{ \frac{R_{i_1} x - \omega_{i_1, i_2, i_3} t}{2\sqrt{R_{i_1} D_x t}} \right\} + \right. \\
&\quad \left. \frac{e^{\frac{x\omega_{i_1, i_2, i_3}}{2D_x}}}{(v - \omega_{i_1, i_2, i_3})} \operatorname{erfc} \left\{ \frac{R_{i_1} x + \omega_{i_1, i_2, i_3} t}{2\sqrt{R_{i_1} D_x t}} \right\} \right] \\
&+ \frac{2v^2}{(\omega_{i_1, i_2, i_3}^2 - v^2)} e^{\left[\frac{xv}{D_x} \frac{k_{i_1} t}{R_{i_1}} \right]} \operatorname{erfc} \left\{ \frac{R_{i_1} x + vt}{2\sqrt{R_{i_1} D_x t}} \right\} ; \text{ when } \frac{k_{i_1}}{R_{i_1}} \neq a_{i_2, i_3}, \text{ and} \\
&= e^{-\frac{k_{i_1} t}{R_{i_1}}} \left[\frac{1}{2} \operatorname{erfc} \left(\frac{R_{i_1} x - vt}{2\sqrt{R_{i_1} D_x t}} \right) + \sqrt{\frac{v^2 t}{\pi R_{i_1} D_x}} e^{-\frac{(R_{i_1} x - vt)^2}{4R_{i_1} D_x t}} \right. \\
&\quad \left. - \frac{1}{2} \left(1 + \frac{xv}{D_x} + \frac{v^2 t}{R_{i_1} D_x} \right) e^{\frac{xv}{D_x}} \operatorname{erfc} \left(\frac{R_{i_1} x + vt}{2\sqrt{R_{i_1} D_x t}} \right) \right] ; \text{ when } \frac{k_{i_1}}{R_{i_1}} = a_{i_2, i_3} \\
\text{where ; } \omega_{i_1, i_2, i_3} &= \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} \\
&\quad (50)
\end{aligned}$$

The above solution is valid only for real values of ‘ ω_{i_1, i_2, i_3} ’. For problems involving complex values for ‘ ω_{i_1, i_2, i_3} ’ in the case when ‘ $\frac{k_{i_1}}{R_{i_1}} \neq a_{i_2, i_3}$ ’, the ‘ F_{i_1, i_2, i_3} ’ terms are given as: (See equation (E.26) in Appendix E) Note: when ‘ $\frac{k_{i_1}}{R_{i_1}} = a_{i_2, i_3}$ ’, the ‘ F_{i_1, i_2, i_3} ’ terms are unchanged.

$$\begin{aligned}
F_{i_1, i_2, i_3} [x, t] &= \frac{2v}{\left(v^2 + \omega_{i_1, i_2, i_3}^*\right)^2} e^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}} \left[\begin{aligned} &\left(Av - B\omega_{i_1, i_2, i_3}^*\right) \text{Cos} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) \\ &-\left(A\omega_{i_1, i_2, i_3}^* + Bv\right) \text{Sin} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) \end{aligned} \right] \\
&+ \frac{v^2}{2R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} e^{\left[\frac{xv}{D_x} - \frac{k_{i_1} t}{R_{i_1}} \right]} \text{erfc} \left\{ \frac{R_{i_1} x + vt}{2\sqrt{R_{i_1} D_x t}} \right\} \\
\text{where ; } \omega_{i_1, i_2, i_3}^* &= \sqrt{\left| v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right) \right|} \\
\text{and } (A + iB) &= \text{erfc} \left\{ \frac{R_{i_1} x + i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\} \tag{51}
\end{aligned}$$

Equation(43) along with equations (44), (45) and (46) and (49) ,(50)and (51) give the complete explicit general solutions to the transport problem described by equation(1) subject to the initial condition given by equation(2) and the boundary condition at ∞ given by equation(3) for the Dirichlet and the Cauchy source boundaries given by equations(4) and (47) respectively.

2.4 Discussion

In the original governing equation given by equation(1), it was assumed that the degradation occurs in the liquid phase only. However, in several real life contaminant transport scenarios such as radioactive transport, decay occurs in both the liquid and solid phases. Under this condition the governing equation should be modified as:

$$\begin{aligned}
R_i \frac{\partial c_i(x,t)}{\partial t} + v \frac{\partial c_i(x,t)}{\partial x} - D_x \frac{\partial^2 c_i(x,t)}{\partial x^2} &= y_i R_{i-1} k_{i-1} c_{i-1}(x,t) - R_i k_i c_i(x,t) ; \forall i = 2, 3, \dots n \\
&= -R_i k_i c_i(x,t) ; i = 1 \\
&; \forall t > 0 \text{ and } 0 < x < \infty \\
&(52)
\end{aligned}$$

Note the additional parameters in the right side of equation(52). The solution to the above equation can be readily obtained from the previous solution given by equation(43) by substituting the value of ‘ k ’ with ‘ Rk ’.

From Sections 2.2 and 2.3, it can be seen that the solutions for the Dirichlet and the Cauchy boundaries share a common structure. However, careful observation indicates that the ‘ G_1^2 ’ and ‘ G_2^2 ’ terms for the Dirichlet boundary (see equation(44)) and the Cauchy Boundary (see equation(49)) are different. Furthermore, from equations (45) and (46) and equations (50) and (51) it can be observed that the ‘ F_{i_1, i_2, i_3} ’ terms involved in the ‘ G ’ terms are distinctly different for the two boundary conditions. The solutions are presented in a format wherein they share a very similar structure. One of the key advantages of this type of presentation is that the implementation of the solution for one type of boundary condition becomes relatively easy if we have the algorithm for the other boundary implemented. Furthermore, the general solution is presented in a format which enables us to directly obtain explicit solutions for any of the species in the chain without involving the computations of its parent chain members. This unique feature makes the general solution computationally efficient. The solutions previously published in the literature have either been restricted to small chain lengths or have been semi analytical solutions for longer chain lengths. The solutions derived in this study overcome both these difficulties and making them highly suitable for screening level models.

In the following chapter, the general solutions are first implemented in a computational platform and then tested against other analytical and semi-analytical solutions using a set of example problems.

CHAPTER III

TESTING AND VALIDATION OF THE ANALYTICAL SOLUTIONS

3.1 Introduction

To test the general solution, one can either compare it analytically against other closed-form solutions or compare the results of its simulation against other analytical or numerical solutions. While testing the general solution using example simulations requires its implementation in a computational platform, analytical comparison is a viable verification option only when we have suitable solutions that solve an identical transport problem. Since explicit closed-form solutions are available only for simplified transport scenarios, the general solution is first simplified to these special cases and then compared against previously published closed-form solutions. The objective of this chapter is to: 1) discuss the computational implementation of the new general analytical solution; 2) test and validate the solution against published analytical and numerical results; and 3) evaluate the salient features and limitations of the solution. In the following sections the general solutions are first validated analytically against the solutions presented by van Genuchten [45] and Higashi and Pigford [24] for simplified transport scenarios. To further test the general solutions for a more general case, example simulations are performed and the results of the general solution are compared against the results generated from the semi-analytical solution given by Quezada et al. [38].

3.2 Implementation of the Solution

To develop a computer code for the general solution, one has to formulate an algorithm based on the looping structure presented in equation(43). In order to evaluate the summation loops, one has to compute all the ‘ G ’ terms in equation(43), which are defined in equations (44) and (49). From these, equations one can observe that the ‘ G ’ terms require the evaluation of ‘ F_{i_1, i_2, i_3} ’ terms. These ‘ F_{i_1, i_2, i_3} ’ terms, which are defined in equations (45), (46), (50) and (51), involve computations of the product of complementary error functions and exponential functions. After evaluating the ‘ F_{i_1, i_2, i_3} ’ terms, one can then compute the ‘ G ’ terms. The general solution can then be implemented by substituting the values for the corresponding ‘ G ’ terms in the summation loop described by equation(43). Although the above procedure appears to be a straightforward task, round-off errors and underflow/overflow errors that occur during the computations of the ‘ G ’ and ‘ F_{i_1, i_2, i_3} ’ terms, can cause severe stability problems even for short chain lengths involving four species [45]. These computational errors are further aggravated when solving problems involving the Cauchy boundary condition for long chain lengths and/or large simulation times.

As pointed out by van Genuchten [45] the round-off errors arise due to the approximations in the computation of the complementary error functions (ERFC) involved in the ‘ F_{i_1, i_2, i_3} ’ terms, especially when employing the Cauchy boundary condition, for large retardation factors and long simulation distances or simulation times. Typically ERFC is evaluated by approximating it either as a closed-form analytical

expression [20] or as an infinite summation series [19]. Van Genuchten [45] suggested that this round-off error could be considerably limited by first substituting the approximate expressions for the ERFCs into the ‘ F_{i_1, i_2, i_3} ’ terms and then combining and simplifying the resulting terms. The mathematical details of this strategy can be obtained from the computer code CHAIN [45]. Although this strategy significantly improves the round-off errors, it involves extensive book keeping procedures. It must be noted that van Genuchten [45] used the Gautschi [20] and Gautschi [19] approximations for computing the ERFC terms. Using a more accurate approximation for the ERFC function, such as the Cody [14] Chebyshev approximation, would provide additional improvement to the round-off errors.

Another important computational challenge encountered in implementing this solution is the handling of very large and/or very small numbers involving the computations of the exponential and ‘ Π ’ functions. These numbers arise when solving problems involving long chain lengths and/or large simulation times. For example, consider the computations involved in the ‘ G_1^2 ’ term in equation(44).

$$G_1^2 = \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i \frac{\left\langle F_{i_2, i_1, -i_2} [x, t] - e^{(-\mu_1 x - a_{i_1, -i_2} t)} - F_{i_2, i_2, i_3} [x, t] + e^{(-\mu_1 x - a_{i_2, i_3} t)} \right\rangle}{\left(a_{i_2, i_3} - a_{i_1, -i_2} \right) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} \left(a_{i_2, i_4} - a_{i_2, i_3} \right)} \quad (53)$$

From equation(53) one can observe that the denominator term involves two independent product loops that have to be evaluated. Under certain parameter combinations, each of the arguments within these product loops can assume very small values (e.g., less than 1E-5) and when this loop is run for long chain lengths (e.g., greater than 10) the product of these arguments results in an extremely small number (lesser than 1E-50). Performing division by a small number of this order, results in very large arguments which cannot be represented accurately by using finite computer precision. Furthermore, mathematical operations (additions and/or subtractions) of these large arguments can result in severe round-off errors. Note that this problem is not only restricted to the ‘ G_1^2 ’ term, but can also apply to the other ‘ G ’ terms. Similarly, one can also observe that the numerator of the ‘ G_1^2 ’ terms in equation(53) involves the computation of exponential functions. While performing simulations for large times and small distances, one may encounter very large arguments within these exponential functions that cannot be represented by finite computer precision. Therefore, it is common to observe overflow errors when solving problems involving large simulation times.

A powerful solution to tackle these two problems involving long chain lengths and large simulation times is to use a log based formulation. The log based formulation involves transforming the arguments whose products have to be evaluated in to the log space. Within the log space the products are evaluated by performing a set of additions. Finally, the evaluated product is inverse transformed from the log scale to the linear scale. By employing the proposed log based formulation, the underflow and/or overflow errors related to evaluating the ‘ Π ’ terms can be virtually eliminated. However, the proposed log based formulation can only be applied to compute product terms and cannot be used to compute summation terms. To tackle the problem of overflow errors while evaluating summation terms, one can make use of the symmetrical property of the ‘ F_{i_1, i_2, i_3} ’ terms. This involves combining the ‘ F_{i_1, i_2, i_3} ’ terms that have identical ‘ i_2 ’ and ‘ i_3 ’ values and then evaluating the sum of these combination terms separately and then substituting them in the main solution [45]. Although this method seems to have a direct approach, the algebraic manipulation involved in combining the various ‘ F_{i_1, i_2, i_3} ’ terms requires the reformulation of the final solution in a different format. This reformulation would be mathematically tedious involving extensive book keeping, especially for problems having long chain lengths.

One can use the log formulation and other computational techniques suggested above for developing a computer code for the general solution. In addition to this, in the section below, another alternative is presented where solutions for several simpler and more practical cases that are computationally less challenging to implement are deduced. These special-case solutions are particularly useful for developing screening level

models. Furthermore, some of these simple solutions can be directly compared against analytical solutions presented in the literature to provide powerful arguments for demonstrating the validity of the general solution.

3.3 Special Cases

In this section solutions to several simplified transport scenarios are presented and where ever possible, these solutions are validated against previously published analytical solutions. It should be noted that for some of these special cases, a more rigorous mathematical analysis (rather than simple substitution) is required to derive these simplified solutions. Furthermore, unlike the general solution these special case solutions impose fewer restrictions on the transport parameter values (Appendices A and D)

3.3.1 Zero Initial Condition

The general solution can be readily simplified to solve transport scenarios where the initial concentrations of all the contaminants are zero. This is done by substituting the value of ‘ c^o ’ as zero. For this case, the general solution simplifies to:

$$c_i(x, t) = \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \{ G_1^1 + h(G_1^1) G_2^1 \} \right] \quad (54)$$

; $\forall i = 1, 2, \dots n$

where; ‘ a_{i_1, i_2} ’ is given by:

$$a_{i_1, i_2} = \begin{cases} \frac{k_{i_1, i_2}}{R_{i_1, i_2}} ; & \text{when } i_2 > 0 \\ \lambda_{i_1} ; & \text{when } i_2 = 0 \end{cases} \quad (55)$$

Note that the ‘ G_1^1 ’ and ‘ G_2^1 ’ terms in equation(54) defined in Appendices A and D remain unchanged. It can be shown that the expressions for the first four species of this special case solution match the solutions given by van Genuchten [45]. Furthermore, from equation(54) one can observe that, for the zero initial condition case, the second term of the general solution given by equation(43) is absent. This not only relaxes some of the restrictions on the transport parameter values but also directly helps in improving the round off errors especially during large simulation times.

3.3.2 Identical Retardation Factors

Special cases arise when the retardation factors of all the species in the transport problem are identical (for example, the transport of non-sorbing set of sequentially decaying contaminants, where the retardation factors of all the species will be 1). One can obtain the solution for this problem from the general solution by simple substitution. The modified solution for this special case problem for both types of boundary conditions is given as:

$$\begin{aligned}
 c_i(x,t) = & \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \{G_2^1\} \right] \\
 & + \sum_{i_1=1}^i \left[R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \{G_2^2\} \right] \quad (56) \\
 & ; \forall i = 1, 2, \dots n
 \end{aligned}$$

Note that the ‘ G_2^1 ’ and ‘ G_2^2 ’ terms defined in Appendices A and D remain unchanged. The exclusion of the ‘ G_1^1 ’ and ‘ G_1^2 ’ terms helps in obtaining a

computationally more stable solution for longer chain lengths. Additionally this solution imposes lesser restriction on the transport parameter values. Sun et al. [42] solved a transport problem similar to the above problem assuming zero initial conditions. However, unlike the solution presented in equation(56), Sun et al. [42] did not provide an explicit closed-form expression, instead they only present a computational algorithm to compute the concentration profiles.

3.3.3 Zero Advection Velocity

In some situations the transport of contaminants is governed mostly by dispersion. The solution for this condition is identical to the general solution given by equation(43), except that the ‘ F_{i_1, i_2, i_3} ’ terms will be modified. For the Dirichlet boundary condition, the

‘ F_{i_1, i_2, i_3} ’ term is given as:

$$F_{i_1, i_2, i_3} [x, t] = \frac{e^{-a_{i_2, i_3} t}}{2} \left[e^{\frac{-x\omega_{i_1, i_2, i_3}}{2D_x}} \operatorname{erfc} \left\{ \frac{R_{i_1} x - \omega_{i_1, i_2, i_3} t}{2\sqrt{R_{i_1} D_x t}} \right\} + e^{\frac{x\omega_{i_1, i_2, i_3}}{2D_x}} \operatorname{erfc} \left\{ \frac{R_{i_1} x - \omega_{i_1, i_2, i_3} t}{2\sqrt{R_{i_1} D_x t}} \right\} \right] \quad (57)$$

$$\text{where ; } \omega_{i_1, i_2, i_3} = \sqrt{4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)}$$

The above expression is valid only for real values of ‘ ω_{i_1, i_2, i_3} ’. If ‘ ω_{i_1, i_2, i_3} ’ is complex the ‘ F_{i_1, i_2, i_3} ’ term is modified as:

$$F_{i_1, i_2, i_3} [x, t] = e^{-a_{i_2, i_3} t} \left[A \cos \left(\frac{x \omega_{i_1, i_2, i_3}^*}{2D_x} \right) - B \sin \left(\frac{x \omega_{i_1, i_2, i_3}^*}{2D_x} \right) \right]$$

$$\text{where ; } \omega_{i_1, i_2, i_3}^* = \sqrt{4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} \quad \text{and ; } (A + iB) = \operatorname{erfc} \left\{ \frac{R_{i_1} x + i \omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\} \quad (58)$$

For the case of the Cauchy boundary condition forcing the advection velocity to be zero would result in a zero boundary condition. To avoid this, it is assumed that the boundary condition at the source does not involve the ‘ v ’ term and modify Cauchy source boundary condition as:

$$-D_x \frac{\partial c_i(0, t)}{\partial x} = \begin{cases} \sum_{i=1}^i B_i e^{-\lambda_i t}, & 0 < t \leq t_o \\ 0, & t > t_o \end{cases} ; \forall i = 1, 2, \dots, n \quad (59)$$

Under this condition, the ‘ F_{i_1, i_2, i_3} ’ term is modified as:

$$F_{i_1, i_2, i_3} [x, t] = e^{-a_{i_2, i_3} t} \left[\frac{e^{-\frac{x \omega_{i_1, i_2, i_3}}{2D_x}}}{\omega_{i_1, i_2, i_3}} \operatorname{erfc} \left\{ \frac{R_{i_1} x - \omega_{i_1, i_2, i_3} t}{2\sqrt{R_{i_1} D_x t}} \right\} + \frac{e^{\frac{x \omega_{i_1, i_2, i_3}}{2D_x}}}{-\omega_{i_1, i_2, i_3}} \operatorname{erfc} \left\{ \frac{R_{i_1} x + \omega_{i_1, i_2, i_3} t}{2\sqrt{R_{i_1} D_x t}} \right\} \right] ; \text{ when } \frac{k_{i_1}}{R_{i_1}} \neq a_{i_2, i_3}, \text{ and}$$

$$= e^{-\frac{k_{i_1} t}{R_{i_1}}} \left[\sqrt{\frac{t}{\pi R_{i_1} D_x}} e^{-\frac{(R_{i_1} x - vt)^2}{4R_{i_1} D_x t}} - \frac{x}{2D_x} \operatorname{erfc} \left(\frac{R_{i_1} x}{2\sqrt{R_{i_1} D_x t}} \right) \right] ; \text{ when } \frac{k_{i_1}}{R_{i_1}} = a_{i_2, i_3}$$

$$\text{where ; } \omega_{i_1, i_2, i_3} = \sqrt{4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} \quad (60)$$

The above expression is valid only for real values of ' ω_{i_1, i_2, i_3} '. If ' ω_{i_1, i_2, i_3} ' is complex the ' F_{i_1, i_2, i_3} ' term is modified as:

$$F_{i_1, i_2, i_3} [x, t] = \frac{2}{\omega_{i_1, i_2, i_3}^*} e^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}} \left[\begin{array}{c} -B\omega_{i_1, i_2, i_3}^* \text{Cos} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) \\ -A\omega_{i_1, i_2, i_3}^* \text{Sin} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) \end{array} \right]$$

$$\text{where ; } \omega_{i_1, i_2, i_3}^* = \sqrt{\left| 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right) \right|} \text{ and } (A + iB) = \text{erfc} \left\{ \frac{R_{i_1} x + i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\}$$

(61)

The solutions obtained for the zero advection velocity case are specialized solutions that can be directly applied to model reactive transport scenarios involving chemical and nuclear repository leakage through diffusion.

3.3.4 Steady State

The behavior of a contaminant plume under steady state conditions is of special interest especially in analyzing monitored natural attenuation (MNA) problems. Steady state solutions avoid the problems related to overflow errors which occur when employing the general solution to solve for large simulation times. Although this problem can be tackled by using the algebraic manipulation technique suggested in Section 3.2, it can be a tedious effort. Furthermore, numerical codes (e.g., RT3D [12]) and semi-analytical solutions (e.g., Quezada et al. [38]), require large amounts of computational resources to accurately model steady state solutions. For these reasons, it is very attractive to obtain explicit solutions for steady state conditions. It must be noted

that unlike previous special cases, it is not possible to deduce the steady state solution by direct substitution. Therefore, the steady state solution is derived by solving the steady state governing equations assuming a constant source for the Dirichlet and Cauchy boundaries conditions. Since the time term is absent in the governing equations, Laplace transform techniques are not involved, and one can directly use the linear transform method given by Clement [11] to uncouple the system of equations. The detailed solution procedure is given in Appendix F. The steady state solution for the Dirichlet boundary is given as:

$$c_i(x) = \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left(\sum_{i_2=1}^{i_1} B_{i_1}^{i_2} \right) \sum_{i_2=i_1}^i \frac{e^{\left[\frac{x}{2D_x} \{v - \sqrt{v^2 + 4D_x k_{i_2}}\} \right]}}{\prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} - k_{i_3})} \right] \quad (62)$$

; $\forall i = 1, 2, \dots, n$

The corresponding steady state solution for the Cauchy boundary is given as:

$$c_i(x) = \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left(\sum_{i_2=1}^{i_1} B_{i_1}^{i_2} \right) \sum_{i_2=i_1}^i \frac{v e^{\left[\frac{x}{2D_x} \{v - \sqrt{v^2 + 4D_x k_{i_2}}\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x k_{i_2}} \right\} \prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} - k_{i_3})} \right] \quad (63)$$

; $\forall i = 1, 2, \dots, n$

From equations(62) and (63) it can be observed that the steady state solutions are less complex compared to the generic transient solutions since they do not involve the ERFC terms.

3.3.5 Zero Dispersion Coefficient

If we ignore the effects of dispersion, the governing transport simplifies to:

$$\begin{aligned} R_i \frac{\partial c_i(x,t)}{\partial t} + v \frac{\partial c_i(x,t)}{\partial x} &= y_{i-1} k_{i-1} c_{i-1}(x,t) - k_i c_i(x,t) ; \forall i = 2, 3, \dots n \\ &= -k_i c_i(x,t) ; i = 1 \\ &; \forall t > 0 \text{ and } 0 < x < \infty \end{aligned} \quad (64)$$

Note that in the absence of dispersion, the Cauchy boundary condition will be identical to the Dirichlet boundary condition. In Appendix G the above problem is solved using the generic exponential initial condition given by equation(2). The solution for the no dispersion condition can also be represented by equation(43). However, the ‘*G*’ terms associated with this solution are defined as: (See equations (G.29) and (G.38) in Appendix G)

$$G_1^1 = \sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i \frac{B_{i_1}^{i_3} \left\langle \begin{array}{l} F_{i_2, i_3, 0} [x, t] - e^{-t_0 \lambda_{i_3}} F_{i_1, i_2, i_3} [x, (t-t_0)] \\ -F_{i_2, i_2, i_4} [x, t] + e^{-t_0 \lambda_{i_3}} F_{i_2, i_2, i_4} [x, (t-t_0)] \end{array} \right\rangle}{(a_{i_2, i_4} - \lambda_{i_3}) \left(\prod_{i_5=i_1, (i_5 \neq i_2, R_{i_5} = R_{i_2})}^i -k_{i_2, i_5} \right) (-R_{i_2, i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2, i_5} (a_{i_2, i_5} - a_{i_2, i_4})}$$

$$G_2^1 = \frac{B_{i_1}^{i_3} \left\langle F_{i_2, i_3, 0} [x, t] - e^{-t_0 \lambda_{i_3}} F_{i_1, i_2, i_3} [x, (t-t_0)] \right\rangle}{\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4}}$$

$$G_1^2 = \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i \frac{\left\langle F_{i_2, i_1, -i_2} [x, t] - e^{(-\mu_{i_1} x - a_{i_1, -i_2} t)} - F_{i_2, i_2, i_3} [x, t] + e^{(-\mu_{i_1} x - a_{i_2, i_3} t)} \right\rangle}{(a_{i_2, i_3} - a_{i_1, -i_2}) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (a_{i_2, i_4} - a_{i_2, i_3})}$$

$$G_2^2 = \frac{-\left\langle F_{i_2, i_1, -i_2} [x, t] - e^{(-\mu_{i_1} x - a_{i_1, -i_2} t)} \right\rangle}{R_{i_2} \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3}}$$

(65)

where; the term ‘ F_{i_1, i_2, i_3} ’ is given by: (See equation (H.5) in Appendix H)

$$F_{i_1, i_2, i_3} [x, t] = u \left(t - \frac{R_i x}{v} \right) e^{-a_{i_2, i_3} \left(t - \frac{R_i x}{v} \right)} \quad (66)$$

It can be shown that in the absence of initial contaminant distributions the solutions derived in this section are identical to those published in Harada et al. [23] and Higashi and Pigford [24]. Note that, although the solution for the no dispersion case appears to be identical in structure to the general solution, the ‘ F_{i_1, i_2, i_3} ’ terms involved in these ‘ G ’ terms, defined in equation (H.5) in equation(66), are distinctly different from that of the general solution. These new ‘ F_{i_1, i_2, i_3} ’ terms do not involve the ERFC terms and hence are easier to compute.

3.4 Example Problems

A general computer code (FORTRAN) that solves equation(43) was developed based on the computational algorithm provided in Section 3.2. Appendix I gives the details of the code along with a sample input and output files. Using this code, the general solution is first tested against two published examples problems: one involving the Dirichlet boundary and the other involving the Cauchy boundary. The example problem for the Dirichlet boundary condition is taken from Cho [9]. The parameters used in this problem are summarized in Table 1. Note: it is assumed that the decay occurs in the liquid phase only. The results of the general solution are tested against the results of the analytical solution given by Cho [9]. Fig.1 compares the concentration profiles of the two solutions. It can be observed that the two solutions give identical results.

Parameter	Value
Velocity; v (cm/h)	1
Dispersion coefficient; D_x (cm ² /h)	0.18
Simulation Time; t (h)	200
Pulse Time; t_0 (h)	200
Retardation factors; R , 1 to 3	2, 1, 1
First-order decay coefficients k , 1 to 3 (1/h)	0.01, 0.1, 0
Yield coefficients; y , 1 to 2	1,1
First-order source decay coefficients; λ , 1 to 3 (1/h)	0,0,0
Boundary constant value species 1; B , 1 to 1	1
Boundary constant value species 2; B , 1 to 2	0, 0
Boundary constant value species 3; B , 1 to 3	0, 0, 0
Initial condition constant; c^0 , 1 to 3	0, 0, 0
Initial condition Exponent; μ , 1 to 3	0, 0, 0

Table 1: Parameters used in the Cho [9] example problem

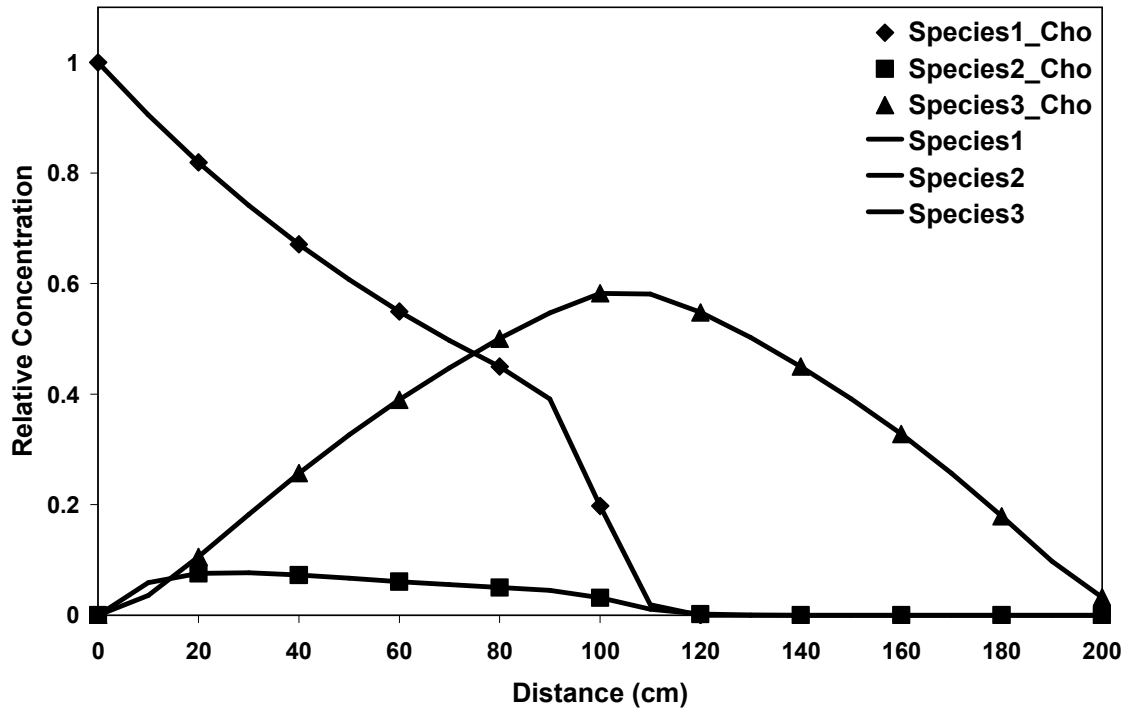


Fig. 1: Comparison of the concentration profiles of the general solution with the Cho [9] solution (3 Species, Dirichlet boundary condition)

The Cauchy boundary example problem is taken from van Genuchten [45]. This problem is a modified version of the problem given by Higashi and Pigford [24]. Here the source is modeled as a step source and decay is assumed to occur in both phases. The transport parameters used in this problem are summarized in Table 2. The results of the general solution are compared against the van Genuchten [45] solution and are summarized in Fig. 2. Observation of the concentration profiles given in Fig. 2 indicates that there is an excellent match between the two solutions.

To further test the performance of the general solution for long chain lengths, a hypothetical 10 species problem was formulated that included an exponentially decaying boundary condition along with spatially varying initial contaminant distributions. Decay is assumed to occur only in the liquid phase and furthermore, the retardation factors of some of the species were made equal. Table 3 summarizes the model parameters used in this problem. The above problem was solved using a Dirichlet boundary condition assuming a pulse time for the source release ' t_o ' as 10 years. The results of the general solution were compared against the semi-analytical solution given by Quezada et al. [38]. Fig. 3 shows the concentration distributions of all the species. These results indicate that the two solutions match well. An identical problem with a source pulse time ' t_o ' of 20 yrs was formulated using a Cauchy boundary. The concentration distributions of all the species were compared against the Quezada et al. [38] solution. Fig. 4 gives the comparative profiles of both these solutions. Again, it can be seen that the profiles predicted by the two solutions have a good match. These simulations further validate the performance of the general solution for transport problems involving long chain lengths.

Parameter	Value
Velocity; v (m/yr)	100
Dispersion coefficient; D_x (m ² /yr)	10
Simulation Time; t (yr)	1.0E+5
Pulse Time; t_0 (yr)	1.0E+4
Retardation factors; R , 1 to 4	1.0E+4, 1.4E+4, 5.0E+4, 5.0E+2
First-order decay coefficients k , 1 to 4 (1/yr)	7.9E-3, 2.8E-6, 8.71E-6, 4.3E-4
Yield coefficients; y , 1 to 3	1,1,1
First-order source decay coefficients; λ , 1 to 4 (1/yr)	8.9E-3, 1.0028E-3, 1.0087E-3, 1.43E-3
Boundary constant value species 1; B , 1 to 1	1.25
Boundary constant value species 2; B , 1 to 2	-1.2504, 1.2504
Boundary constant value species 3; B , 1 to 3	4.43684E-4, 5.93431E-1, -5.93874E-1
Boundary constant value species 4; B , 1 to 4	-5.1674E-6, 1.20853E-2, -1.22637E-2, 1.78958E-4
Initial condition constant; c^0 , 1 to 4	0, 0, 0, 0
Initial condition Exponent; μ , 1 to 4	0, 0, 0, 0

Table 2: Parameters used in the modified Higashi and Pigford [24] example problem as given by van Genuchten [45]

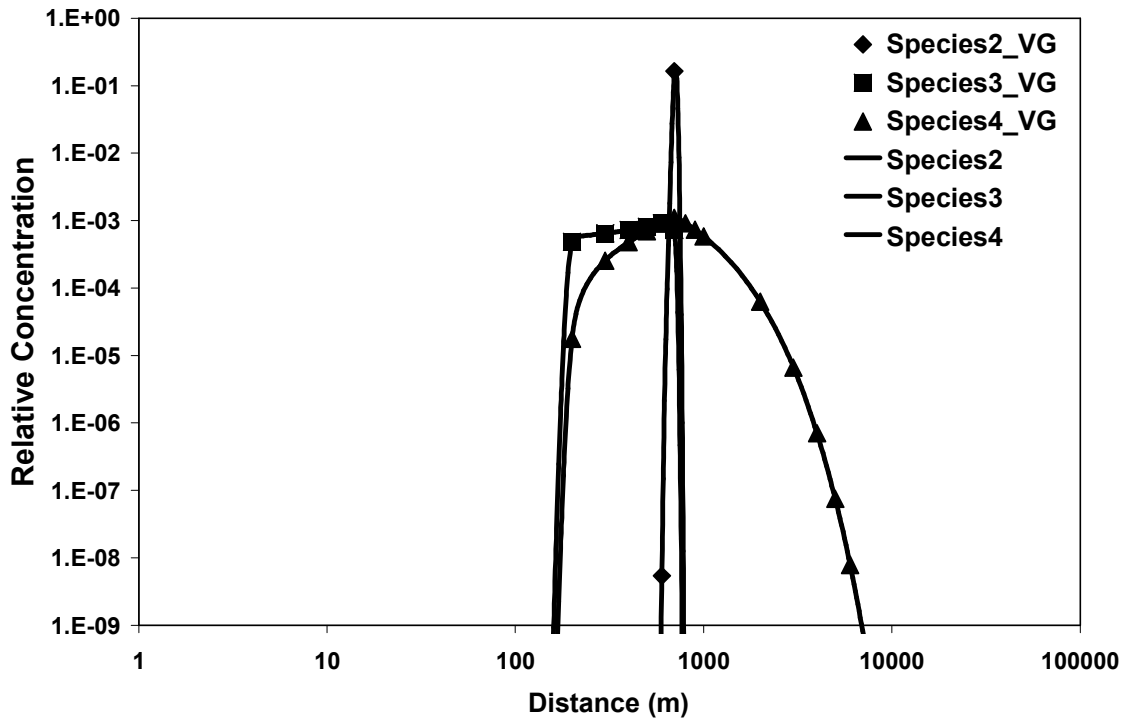


Fig. 2: Comparison of the concentration profiles of the general solution with the van Genuchten [45] solution (4 Species, Cauchy boundary condition)

Parameter (Should be updated)	Value
Velocity; v (m/yr)	5
Dispersion coefficient; D_x (m^2/yr)	50
Simulation Time; t (yr)	20
Pulse Time; t_0 (yr)	10 (For Dirichlet) and 20 (For Cauchy)
Retardation factors; R , 1 to 10	1.9, 1, 1.4, 1, 5, 8, 1.4, 3.1, 1, 1
First-order decay coefficients k , 1 to 10 (1/yr)	3, 2, 1.5, 1.25, 2.75, 1, 0.75, 0.5, 0.25, 0.1
Yield coefficients; y , 1 to 9	1, 2, 1.5, 0.4, 1, 1, 0.7, 0.9, 1
First-order source decay coefficients; λ , 1 to 10 (1/yr)	0.1, 0.75, 0.5, 0.25, 0, 0, 0.3, 1, 0, 0.65
Boundary constant value species 1; B , 1 to 1	10
Boundary constant value species 2; B , 1 to 2	0, 5
Boundary constant value species 3; B , 1 to 3	0, 0, 2.5
Boundary constant value species 4; B , 1 to 4	0, 0, 0, 0
Boundary constant value species 5; B , 1 to 5	0, 0, 0, 0, 10
Boundary constant value species 6; B , 1 to 6	0, 0, 0, 0, 0, 5
Boundary constant value species 7; B , 1 to 7	0, 0, 0, 0, 0, 0, 2.5
Boundary constant value species 8; B , 1 to 8	0, 0, 0, 0, 0, 0, 0, 0
Boundary constant value species 9; B , 1 to 9	0, 0, 0, 0, 0, 0, 0, 0, 0
Boundary constant value species 10; B , 1 to 10	0, 0, 0, 0, 0, 0, 0, 0, 0, 0
Initial condition constant; c^0 , 1 to 10	0, 0.1, 0.2, 0, 0.25, 0.3, 0.15, 0, 0, 0
Initial condition Exponent; μ , 1 to 10	0, 0.01, 0, 0, 0.02, 0.01, 0.1, 0, 0, 0

Table 3: Parameters used in the new 10 species example problem

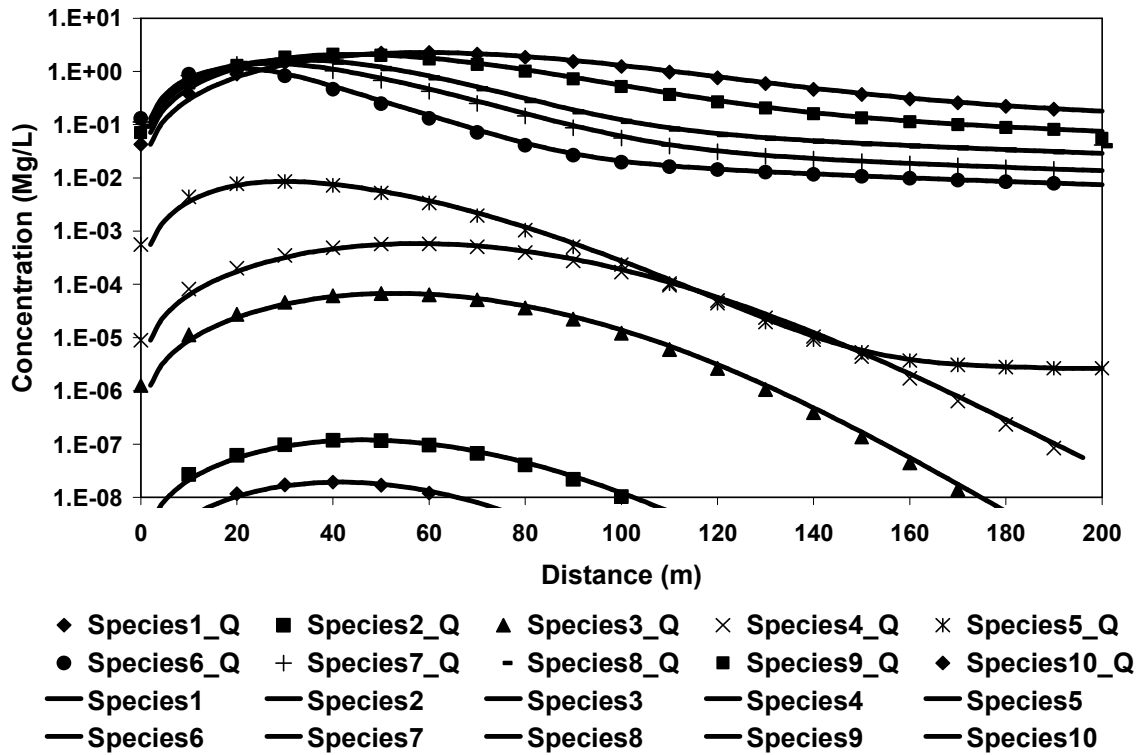


Fig. 3: Comparison of the concentration profiles of the general solution with the Quezada et al. [38] semi-analytical solution (10 Species, Dirichlet boundary condition)

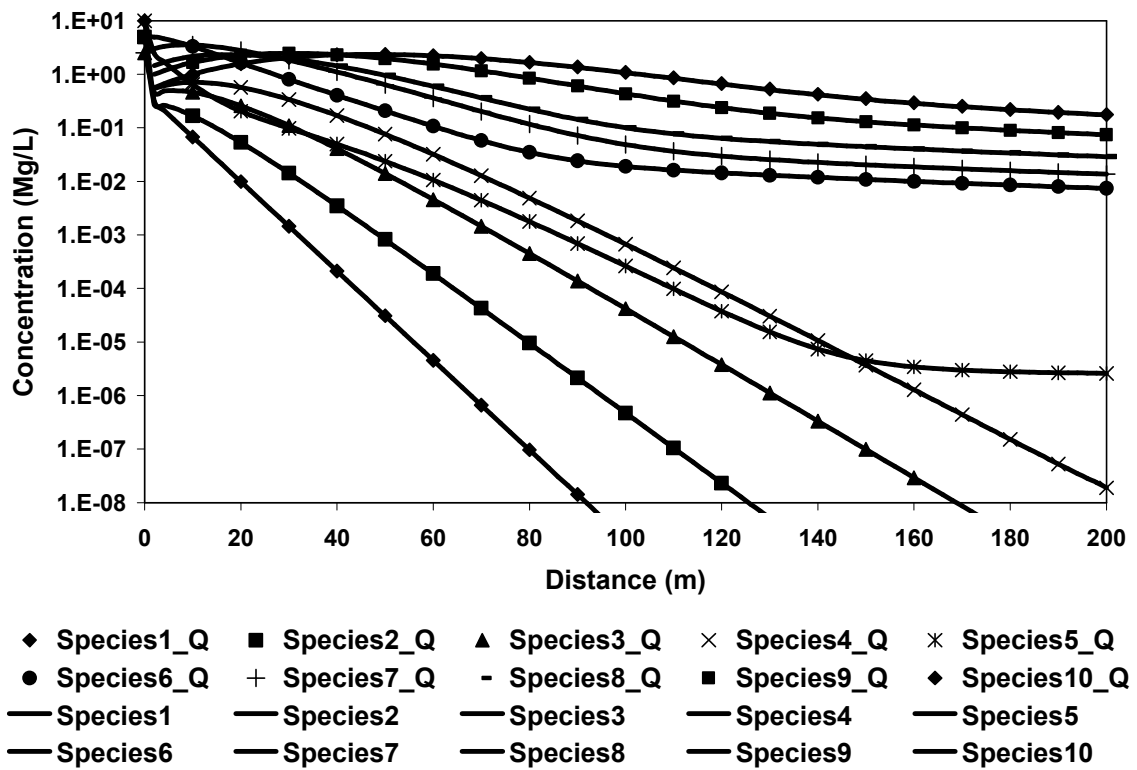


Fig. 4: Comparison of the concentration profiles of the general solution with the Quezada et al. [38] semi-analytical solution (10 Species, Cauchy boundary condition)

3.5 Salient Features of the General Solution

One of the most important features of the general solution is that it is a generic solution that can solve an arbitrary number of species. Furthermore, the final solution is given in an explicit closed-form format that avoids numerical methods. In addition to this, the general solution is formulated and presented in format such that solutions for the Dirichlet and the Cauchy boundaries share a common structure. This enables the convenient implementation of both these solutions by designing one code that shares several common routines having the capability to easily switch between the two boundary conditions.

Another interesting advantage of this solution is that it can be used to directly obtain the solution for a particular species in the chain without involving the computations of other species. This is extremely efficient when modeling scenarios where there the daughter product would be more toxic or more mobile than the parent compound. In this case, one would be interested in obtaining the concentration profiles for the daughter product only. Solving this problem numerically would involve the computation of all the predecessor species and this can be time consuming. A similar situation also arises when employing the recursive type semi-analytical solutions presented by Harada et al. [23] (See section4, p6, eq4.20). Their solution expression for any species ' $i > 1$ ' involves the concentration term of its predecessor ' $i-1$ 'th species.

The third feature of this solution is that it can be readily extended to a general diverging reaction network [43]. One can conceptualize a diverging network as a superposition of a series of parallel reaction networks each of which can be solved independently. It must be noted that one diverging species will have no effect on the

other diverging species and hence they can be considered independent of each other. Hence in order to solve a diverging reaction network, one has to split the network into a set of parallel sequential reaction networks and solve each sequential chain independently.

The one dimensional solutions presented in this work can be readily extended to solve multi-dimensional transport problems involving transverse dispersion terms using the approximate Domenico solution [16]. The mathematical details of this extension strategy are provided in Quezada [37]. As pointed out by Quezada [37], this strategy yields an approximate solution. In the following chapter this approximation is investigated to identify transport situations where the Domenico solution can be advantageously applied with high accuracy [41].

3.6 Limitations of the General Solution

Despite its advantages, the general solution and its simplified solutions have some key limitations. As with the case of all analytical solutions, these solutions can model idealized transport problems only. Situations involving heterogeneity, variation in the flow field, presence of source/sink terms, etc. cannot be modeled using these solutions. Furthermore, these solutions are restricted to modeling first-order kinetic reactions only. Although this is highly suitable for modeling radioactive waste transport and certain types of simplified chemical reactions, these solutions cannot capture the more generic reaction kinetics such as the Monod kinetics.

The general solution places several limitations on the values of the transport parameters which are summarized in Table 4 (Appendices A and D). These limiting

conditions arise when factorizing the denominator of the ‘ G ’ terms during the computation of the inverse Laplace transforms. An easy approach to tackle this problem would be to marginally perturb the individual model parameters such that the limiting conditions are averted [45]. This would enable us to tackle the problem using the general code. However, this technique can aggravate the existing computational problems related to overflow/under flow errors. Another more rigorous approach to tackle this problem would be to perform a mathematical analysis incorporating these special cases and evaluating the corresponding alternate expressions, similar to the one performed on the case of identical retardation factors (i.e. when ‘ $R_{i_1, i_2} = 0 ; \forall i_1, i_2 = 1, 2, \dots n ; where ; i_1 \neq i_2$ ’). This analysis would involve modifying the factorization of the ‘ G ’ terms to include the limiting conditions and performing generalized inverse Laplace transform operations on the alternate expressions.

Finally, it must be noted that unlike the semi-analytical solutions presented by Clement [11] or Quezada et al. [38] which can model arbitrary reaction networks, the general solution can only model sequential and diverging reaction networks. However, it is possible to extend this solution to solve systems involving complex reaction networks using a similar approach. The key step required to perform this extension would be to obtain generalized expressions for the linear transform matrices used to uncouple the coupled system of equations.

$k_{i_1, i_2} \neq 0 ; \forall i_1, i_2 = 1, 2, \dots, n$ <i>where ; $i_1 \neq i_2$</i>
$a_{i_1, i_2} \neq a_{i_1, i_3} ; \forall i_1, i_2, i_3 = 1, 2, \dots, n$ <i>where ; $i_1 \neq i_2$, $i_1 \neq i_3$, $i_2 \neq i_3$, $R_{i_1} \neq R_{i_2}$ and $R_{i_1} \neq R_{i_3}$</i>
$a_{i_1, i_2} \neq \lambda_{i_3} ; i \leq i_1, i_2 \leq n ; 1 \leq i_3 \leq i ; \forall i = 1, 2, \dots, n$ <i>where ; $i_1 \neq i_2$ and $R_{i_1} \neq R_{i_2}$</i>
$a_{i_1, i_2} \neq a_{i_1, -i_1} ; i \leq i_1, i_2 \leq n ; \forall i = 1, 2, \dots, n$ <i>where ; $i_1 \neq i_2$ and $R_{i_1} \neq R_{i_2}$</i>

Table 4: Summary of parameter limitations

CHAPTER IV
INVESTIGATION OF ERROR ASSOCIATED WITH THE DOMENICO
APPROXIMATION

4.1 Introduction

This chapter investigates some of the approximations involved in extending the one-dimensional solutions derived earlier into multiple-dimensions. One of the most popular multi-dimensional analytical solutions used for modeling multi-dimensional ground water contaminant plumes is the Domenico solution [16]. The Domenico solution is an approximate three-dimensional solution that describes the fate and transport of a decaying contaminant plume evolving from a finite patch source. This solution was based on an approach previously published by Domenico and Robbins [17] for modeling a non-decaying contaminant plume. Prior to this work, Cleary and Ungs [10] presented an analytical solution to a similar three dimensional transport problem for a domain finite in y and z directions. Later, Sagar [39] published an exact analytical solution to the transport problem considered by Domenico and Robbins [17]. Wexler [48] extended the Sagar [39] solution to include the effects of reaction and presented an exact analytical solution to the transport problem considered by Domenico [16]. However, these solutions are not closed form expressions since they involve numerical evaluation of a definite integral. This numerical integration step can be computationally demanding and can also introduce numerical errors. The key advantage of the Domenico and Robbins

[17] approach is that it provides a closed form solution without involving numerical integration procedures. Due to this computational advantage, the Domenico solution has been widely used in several public domain design tools including the USEPA tools BIOCHLOR and BIOSCREEN [3, 36].

Although the Domenico solution is extensively employed in several ground water transport models, its approximate nature has received mixed reviews over the years. For example, West and Kueper [47] compared the BIOCHLOR model against a more rigorous analytical solution and observed considerable discrepancies. By comparing the near field concentration profiles they concluded that the Domenico solution can produce errors up to 50%. Guyonnet and Neville [22] compared the Domenico solution against the Sagar [39] solution and presented the results in a non-dimensional form. They concluded that for ground water flow regimes dominated by advection and mechanical dispersion the discrepancies between the two solutions can be considered negligible along the plume centerline. They further added that the errors may increase significantly outside the plume centerline.

The above review indicates that there are conflicting opinions regarding the performance of the Domenico solution. Furthermore, since the development of the Domenico solution was based on a heuristic approach, researchers have expressed skepticism regarding its validity [47]. Presently, there are several unanswered issues related to the performance of this solution that include: Is there a mathematical basis for deriving the Domenico solution? If so, what are the approximations involved in deriving the solution? What are the errors associated with these approximations? And finally, under what conditions are these approximations valid? To answer these questions, we

need a fundamental understanding of the nature of the approximations involved in the Domenico solution. The focus of this chapter is to perform a rigorous mathematical analysis on the origin and development of the Domenico solution. The outcomes of this analysis are used to develop some general guidelines for the appropriate use of the solution.

4.2 Governing Equations

The transport problem considered by Domenico [16] assumes a patch source of constant concentration ‘ c_0 ’ located at the origin in a clean, semi-infinite aquifer. The contaminant is subjected to advection in the x direction and dispersive mixing in all three directions. Furthermore, it is assumed that the contaminant decays through a first order process. The governing transport equation considered by Domenico [16] is:

$$\frac{\partial c}{\partial t} = -v \frac{\partial c}{\partial x} + D_x \frac{\partial^2 c}{\partial x^2} + D_y \frac{\partial^2 c}{\partial y^2} + D_z \frac{\partial^2 c}{\partial z^2} - kc \quad (67)$$

The initial and boundary conditions are:

$$\begin{aligned} c(x, y, z, 0) &= 0 \quad \forall \quad 0 < x < \infty, \quad -\infty < y < \infty, \quad -\infty < z < \infty \\ c(0, y, z, t) &= c_0 \quad -\frac{Z}{2} < z < \frac{Z}{2}, \quad -\frac{Y}{2} < y < \frac{Y}{2}, \quad \forall \quad t > 0 \\ &= 0 \quad \text{otherwise,} \quad \forall \quad t > 0 \\ \lim_{x \rightarrow \infty} \frac{\partial c(x, y, z, t)}{\partial x} &= 0 \\ \lim_{y \rightarrow \pm\infty} \frac{\partial c(x, y, z, t)}{\partial y} &= 0 \\ \lim_{z \rightarrow \pm\infty} \frac{\partial c(x, y, z, t)}{\partial z} &= 0 \end{aligned} \quad (68)$$

where; ‘ c ’ is the concentration of the contaminant [ML^{-3}]; ‘ c_o ’ is the concentration at the source [ML^{-3}]; ‘ Y ’ and ‘ Z ’ are the source dimensions in y- and z- directions, respectively [L]; ‘ D_x ’, ‘ D_y ’, and ‘ D_z ’ are the dispersion coefficients in x, y and z directions, respectively [L^2T^{-1}]; ‘ v ’ is the advection velocity in the x direction [LT^{-1}]; and ‘ k ’ is the first-order decay coefficient [T^{-1}].

4.3 Review of the Domenico Solution

The Domenico [16] solution was based on an approximate approach given by Domenico and Robbins [17]. Therefore, we first present a detailed review of the development of the Domenico and Robbins solution. Domenico and Robbins [17] began their analysis by presenting the following exact analytical solution that describes the transport of an instantaneous pulse source in a three-dimensional domain [25]:

$$\begin{aligned}
 c(x, y, z, t) &= \frac{c_o}{8} \tilde{f}_x(x, t) \tilde{f}_y(y, t) \tilde{f}_z(z, t) \\
 \text{where ; } \tilde{f}_x(x, t) &= \left[\text{erf} \left\{ \frac{x - vt + \frac{X}{2}}{2(D_x t)^{1/2}} \right\} - \text{erf} \left\{ \frac{x - vt - \frac{X}{2}}{2(D_x t)^{1/2}} \right\} \right] \\
 \tilde{f}_y(y, t) &= \left[\text{erf} \left\{ \frac{y + \frac{Y}{2}}{2(D_y t)^{1/2}} \right\} - \text{erf} \left\{ \frac{y - \frac{Y}{2}}{2(D_y t)^{1/2}} \right\} \right] \\
 \text{and } \tilde{f}_z(z, t) &= \left[\text{erf} \left\{ \frac{z + \frac{Z}{2}}{2(D_z t)^{1/2}} \right\} - \text{erf} \left\{ \frac{z - \frac{Z}{2}}{2(D_z t)^{1/2}} \right\} \right]
 \end{aligned} \tag{69}$$

They then present the following one-dimensional analytical solution [15]:

$$c(x,t) = \frac{c_0}{2} f_x(x,t)$$

$$\text{where ; } f_x(x,t) = \text{erfc} \left\{ \frac{x-vt}{2(D_x t)^{1/2}} \right\} \quad (70)$$

Note that the above expression is the solution to the standard one-dimensional advection dispersion equation for an instantaneous source extending from zero to negative infinity [5].

To account for the transverse dispersion due to a finite sized two-dimensional source they employed the following two analytical solutions [15]:

$$c(y,t) = \frac{c_0}{2} f_y(y,t) \quad \text{and} \quad c(z,t) = \frac{c_0}{2} f_z(z,t)$$

$$\text{where ; } f_y(y,t) = \left[\text{erf} \left\{ \frac{y + \frac{Y}{2}}{2(D_y t)^{1/2}} \right\} - \text{erf} \left\{ \frac{y - \frac{Y}{2}}{2(D_y t)^{1/2}} \right\} \right] \quad (71)$$

$$\text{and } f_z(z,t) = \left[\text{erf} \left\{ \frac{z + \frac{Z}{2}}{2(D_z t)^{1/2}} \right\} - \text{erf} \left\{ \frac{z - \frac{Z}{2}}{2(D_z t)^{1/2}} \right\} \right]$$

Note that $c(y,t)$ and $c(z,t)$ are solutions to two independent one-dimensional transient diffusion equations subjected to an instantaneous line source of widths Y and Z respectively. Further, it can be observed that the terms $f_y(y,t)$ and $f_z(z,t)$ in equation(71) are identical to the terms $\tilde{f}_y(y,t)$ and $\tilde{f}_z(z,t)$ in the Hunt [25] solution.

Domenico and Robbins [17] multiplied the one-dimensional solution $f_x(x,t)$ with these “transverse spreading terms” $f_y(y,t)$ and $f_z(z,t)$ and presented the following expression.

$$c(x, y, z, t) = \frac{c_o}{8} f_x(x, t) f_y(y, t) f_z(z, t) \quad (72)$$

However, the authors did not justify this superposition step. Note that the Hunt solution [25] was never used in this analysis. At this stage, Domenico and Robbins presented the following arguments: *“The product of these three integral solutions [shown in equation(72)] describes a semi-infinite contaminated parcel which moves in the positive x direction with a one-dimensional velocity but which continuously expands in size in directions transverse to x throughout the whole domain of x, i.e., in the positive and negative regions. This is because the time t in the transverse spreading terms is interpreted as running time. Reinterpreting this time as x/v for a moving coordinate system, as is common in all transverse spreading models (Bruch and Street, 1967; Ogata, 1970; Domenico and Palciauskas, 1982) has the effect of maintaining the original source dimensions at x=0 so that the condition C=C_o is maintained at x=0 for t>0.”* Using these arguments they reinterpret the time term t in the transverse spreading terms $f_y(y, t)$ and $f_z(z, t)$ as x/v. However, the authors did not provide a mathematical reasoning for this time reinterpretation step. Further, all the references cited in the above text solve fundamentally different problems and we will address this issue in a later section. Using this time reinterpretation step, equation(72) was modified as:

$$c(x, y, z, t) = \frac{c_0}{8} f_x(x, t) f_y(y, x) f_z(z, x)$$

where ; $f_x(x, t) = \operatorname{erfc} \left\{ \frac{x - vt}{2(D_x t)^{1/2}} \right\}$

$$f_y(y, x) = \left[\operatorname{erf} \left\{ \frac{y + \frac{Y}{2}}{2(D_y x / v)^{1/2}} \right\} - \operatorname{erf} \left\{ \frac{y - \frac{Y}{2}}{2(D_y x / v)^{1/2}} \right\} \right] \quad (73)$$

and $f_z(z, x) = \left[\operatorname{erf} \left\{ \frac{z + \frac{Z}{2}}{2(D_z x / v)^{1/2}} \right\} - \operatorname{erf} \left\{ \frac{z - \frac{Z}{2}}{2(D_z x / v)^{1/2}} \right\} \right]$

Equation(73) was presented as the final solution to the continuous finite patch source problem considered by Domenico and Robbins [17].

Later, Domenico [16] incorporated the effects of first-order decay by replacing the $f_x(x, t)$ term with an analytical solution for the semi-infinite pulse source problem with a decay term, presented by Bear [5]. The final solution was given as [16]:

$$c(x, y, z, t) = \frac{c_0}{8} f_x(x, t) f_y(y, x) f_z(z, x)$$

where ; $f_x(x, t) = \left[\exp \left\{ \frac{x}{2\alpha_x} \left[1 - \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2} \right] \right\} \operatorname{erfc} \left\{ \frac{x - vt \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2}}{2(\alpha_x vt)^{1/2}} \right\} \right]$

$$f_y(y, x) = \left[\operatorname{erf} \left\{ \frac{y + \frac{Y}{2}}{2(\alpha_y x)^{1/2}} \right\} - \operatorname{erf} \left\{ \frac{y - \frac{Y}{2}}{2(\alpha_y x)^{1/2}} \right\} \right] \quad (74)$$

and $f_z(z, x) = \left[\operatorname{erf} \left\{ \frac{z + \frac{Z}{2}}{2(\alpha_z x)^{1/2}} \right\} - \operatorname{erf} \left\{ \frac{z - \frac{Z}{2}}{2(\alpha_z x)^{1/2}} \right\} \right]$

where; $\alpha_x = \frac{D_x}{v}$, $\alpha_y = \frac{D_y}{v}$ and $\alpha_z = \frac{D_z}{v}$ are the dispersivities in the x, y and z

directions respectively [L].

Martyn-Hayden and Robbins [31] later modified the Domenico [16] solution, referred in this work as modified-Domenico solution, by incorporating the following one-dimensional solution (which describes a constant source boundary) in the $f_x(x, t)$ term [5]:

$$f_x = \left(\exp \left\{ \frac{x}{2\alpha_x} \left[1 - \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2} \right] \right\} \operatorname{erfc} \left\{ \frac{x - vt \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2}}{2(\alpha_x vt)^{1/2}} \right\} \right) + \left(\exp \left\{ \frac{x}{2\alpha_x} \left[1 + \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2} \right] \right\} \operatorname{erfc} \left\{ \frac{x + vt \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2}}{2(\alpha_x vt)^{1/2}} \right\} \right) \quad (75)$$

As pointed out by Bear [5] if the value of $\frac{x}{\alpha_x}$ is sufficiently large, a condition

usually satisfied in practice, the additional term in the above equation can be safely ignored.

The above review shows that the development of various forms of the Domenico solution does not have a rigorous mathematical basis. The empirical arguments provided by the authors are vague because the mathematical procedures implied by these arguments are inexplicit and nebulous. In the following section a more rigorous approach to derive the Domenico solution that clearly stated the approximations involved is presented.

4.4 A Rigorous Approach to Derive the Domenico Solution

The exact semi-analytical solution for the three dimensional transport problem described by equations (67) and (68), considered by Domenico [16], was provided by Wexler [48] as:

$$c(x, y, z, t) = \frac{C_o}{8} \int_{\tau=0}^{\tau=t} f'_x(x, \tau) f'_y(y, \tau) f'_z(z, \tau) d\tau$$

$$\text{where ; } f'_x(x, \tau) = \frac{x}{\sqrt{\pi D_x}} \exp\left(\frac{vx}{2D_x}\right) \frac{\exp\left(\frac{-v^2}{4D_x}\tau - k\tau + \frac{-x^2}{4D_x\tau}\right)}{\tau^{3/2}}$$

$$f'_y(y, \tau) = \left[\operatorname{erf}\left\{\frac{y + \frac{Y}{2}}{2(D_y\tau)^{1/2}}\right\} - \operatorname{erf}\left\{\frac{y - \frac{Y}{2}}{2(D_y\tau)^{1/2}}\right\} \right] \quad (76)$$

$$\text{and } f'_z(z, \tau) = \left[\operatorname{erf}\left\{\frac{z + \frac{Z}{2}}{2(D_z\tau)^{1/2}}\right\} - \operatorname{erf}\left\{\frac{z - \frac{Z}{2}}{2(D_z\tau)^{1/2}}\right\} \right]$$

To obtain the Domenico solution from the above exact solution, we replace the value of τ in the transverse spreading terms $f'_y(y, \tau)$ and $f'_z(z, \tau)$ with x/v (the validity of this substitution will be discussed later). This yields the following expression:

$$c(x, y, z, t) = \frac{C_o}{8} f_y(y, x) f_z(z, x) \int_{\tau=0}^{\tau=t} f'_x(x, \tau) d\tau$$

$$\text{where ; } f_y(y, x) = \left[\operatorname{erf}\left\{\frac{y + \frac{Y}{2}}{2(\alpha_y x)^{1/2}}\right\} - \operatorname{erf}\left\{\frac{y - \frac{Y}{2}}{2(\alpha_y x)^{1/2}}\right\} \right] \quad (77)$$

$$\text{and } f_z(z, x) = \left[\operatorname{erf}\left\{\frac{z + \frac{Z}{2}}{2(\alpha_z x)^{1/2}}\right\} - \operatorname{erf}\left\{\frac{z - \frac{Z}{2}}{2(\alpha_z x)^{1/2}}\right\} \right]$$

where; $\alpha_y = \frac{D_y}{v}$ and $\alpha_z = \frac{D_z}{v}$. Note that by substituting $\tau = x/v$ we have made

the transverse spreading terms $f_y(y, x)$ and $f_z(z, x)$ independent of time and hence they will not participate in the integration process. Without the transverse terms, the definite integral can be evaluated analytically as shown in Appendix J. Therefore, the above equation can be simplified as:

$$c(x, y, z, t) = \frac{c_o}{8} f_x(x, t) f_y(y, x) f_z(z, x)$$

$$\text{where ; } f_x(x, t) = \left(\exp \left\{ \frac{x}{2\alpha_x} \left[1 - \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2} \right] \right\} \operatorname{erfc} \left\{ \frac{x - vt \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2}}{2(\alpha_x vt)^{1/2}} \right\} \right. \quad (78)$$

$$\left. + \exp \left\{ \frac{x}{2\alpha_x} \left[1 + \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2} \right] \right\} \operatorname{erfc} \left\{ \frac{x + vt \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2}}{2(\alpha_x vt)^{1/2}} \right\} \right)$$

Equation(78) is identical to the modified-Domenico solution shown in equation(75). If we set the first-order decay coefficient k to zero equation(78) reduces to:

$$c(x, y, z, t) = \frac{c_o}{8} f_x(x, t) f_y(y, x) f_z(z, x)$$

$$\text{where ; } f_x(x, t) = \operatorname{erfc} \left\{ \frac{x - vt}{2(\alpha_x vt)^{1/2}} \right\} + \exp \left\{ \frac{x}{\alpha_x} \right\} \operatorname{erfc} \left\{ \frac{x + vt}{2(\alpha_x vt)^{1/2}} \right\} \quad (79)$$

Equation(79) is similar to the Domenico and Robbins (1985) solution given by equation(73). The additional expression in the $f_x(x, t)$ term in equations (78) and (79) is due to the use of the expanded form of the one-dimensional solution that describes a constant concentration boundary condition instead of a semi-infinite pulse source boundary condition.

The above analysis shows that the only approximation required for rigorously deriving the Domenico solution is the time reinterpretation step, where τ is replaced by x/v in the transverse dispersion terms. In the following section, a detailed mathematical analysis that investigates the validity of this approximation is presented

4.5 Mathematical Analysis of the Validity of the Approximation Involved in the Domenico Solution

Review of transport modeling literature indicates that it is common to replace τ with x/v in the transverse dispersion terms when solving convection dominated problems that have low longitudinal mixing. For example, Bruch and Street [6] used a similar assumption to solve the advection-dispersion problem when the longitudinal mixing was smaller than the transverse mixing. Another example of a convection-dominated problem that employs this approximation is the air pollution model used for predicting the fate and transport of smoke plumes evolving from chimneys [46]. Here, the transport is dominated by convection along the wind direction and dispersive mixing is restricted to the transverse directions only. Neglecting the effects of longitudinal dispersion in such problems simplifies the governing transport equation as:

$$\frac{\partial c}{\partial t} = -v \frac{\partial c}{\partial x} + D_y \frac{\partial^2 c}{\partial y^2} + D_z \frac{\partial^2 c}{\partial z^2} - kc \quad (80)$$

Consider solution to the above transport problem subject to the following Domenico-type initial and boundary conditions:

$$\begin{aligned}
c(x, y, z, 0) &= 0 \quad \forall \quad 0 < x < \infty, \quad -\infty < y < \infty, \quad -\infty < z < \infty \\
c(0, y, z, t) &= c_0 \quad -\frac{Z}{2} < z < \frac{Z}{2}, \quad -\frac{Y}{2} < y < \frac{Y}{2}, \quad \forall \quad t > 0 \\
&= 0 \quad \text{otherwise,} \quad \forall \quad t > 0
\end{aligned} \tag{81}$$

$$\lim_{y \rightarrow \pm\infty} \frac{\partial c(x, y, z, t)}{\partial y} = 0$$

$$\lim_{z \rightarrow \pm\infty} \frac{\partial c(x, y, z, t)}{\partial z} = 0$$

In Appendix K, Laplace transform techniques are used to solve the above problem and the resulting exact analytical solution is:

$$\begin{aligned}
c(x, y, z, t) &= \frac{c_0}{8} f_x^o(x, t) f_y(x, y) f_z(x, z) \\
\text{where ; } f_x^o(x, t) &= 2 \exp\left(-\frac{kx}{v}\right) u\left\{t - \frac{x}{v}\right\} \\
\text{and } u\left\{t - \frac{x}{v}\right\} &\text{ is the step function given by;}
\end{aligned} \tag{82}$$

$$u\left\{t - \frac{x}{v}\right\} = \begin{cases} 0 & \text{if } t \leq \frac{x}{v} \\ 1 & \text{if } t > \frac{x}{v} \end{cases}$$

where; $f_y(x, y)$ and $f_z(x, z)$ are identical to the expressions given in equation(77).

Since the Domenico approach approximates τ as x/v in the transverse dispersion terms one can hypothesize that the Domenico approximation must be valid when α_x is zero. To test this hypothesis, a limiting analysis is performed on the modified-Domenico solution by forcing α_x to zero; this is expressed as:

$$c(x, y, z, t) = \lim_{\alpha_x \rightarrow 0} \frac{c_0}{8} f_x(x, t) f_y(y, x) f_z(z, x) \tag{83}$$

The mathematical details of this limiting analysis are given in Srinivasan et al. [41]. The analysis shows that when α_x approaches zero the modified-Domenico solution relaxes to the exact analytical solution given by equation(82). This proves that the Domenico approximation indeed yields an exact analytical solution when α_x is equal to zero.

4.6 Analysis of the Error Associated With the Domenico Solution

The mathematical analysis presented in the previous section demonstrates that the time reinterpretation step, where τ is replaced with x/v , is exactly valid when $\alpha_x = 0$. From these results one could also infer that this time reinterpretation process provides a reasonable approximation when longitudinal dispersion plays an insignificant role in the overall transport. Hence, the Domenico solution can be expected to produce reasonable estimates for advection-dominated problems; however, it can introduce significant errors for longitudinal dispersion-dominated problems.

Another important feature of the time reinterpretation step is that it forces a quasi steady-state condition along the transverse directions at all times. In other words, the ‘conceptual’ residence time (x/v value) associated with a point located at the centerline to disperse contaminant mass in the transverse directions is independent of the simulation time. Further, this residence time is also assumed to increase linearly with respect to x . These unrealistic assumptions regarding residence times will lead to erroneous predictions, especially beyond the advective front. For example, consider a problem where $v = 50$ m/year and we are interested in predicting the concentration distribution of

a two year plume ($t = 2$ years) at a location $x = 200$ meters. The Domenico solution will estimate the residence time τ for our location of interest $x = 200$ meters as $\tau = x/v = 4$ years; this in fact is greater than the total simulation time it self! This is an unrealistic assumption since a 2 year old plume simply cannot have the time to disperse for 4 years! For a particle located at the advective front the residence time assumed by the Domenico solution is 2 years (the simulation time), and for all the particles located behind the advective front the residence time assumed by the Domenico solution will be equal to x/v (which will be less than 2 years); these seem to be reasonable estimates. However, for all points beyond the advective front i.e. $x > 100$ meters, the Domenico solution will assign unrealistic conceptual residence times which will be greater than the simulation time $t = 2$ years. It must be noted that this incorrect behavior will vanish when α_x is zero because, for this case, the plume will abruptly end at the advective front and the residence time for each particle located at or behind the advective front will in fact be equal to x/v .

When solving steady state problems, the assumption related to residence time should be a reasonable approximation. This is because, at steady state, the theoretical advective front will be at infinity. Therefore, the time reinterpretation should be reasonable for any finite domain. Hence, the performance of the Domenico solution under steady state conditions can be expected to be better. However, it is important to note that even under steady state conditions the solution will not be exact because it will still ignore the transport due to longitudinal mixing. In general, it can be concluded that the Domenico solution can be expected to perform better behind the advective front. In the following section we use an example problem to illustrate the implication of these theoretical results.

4.7 Example Problem

The example problem presented by Domenico and Robbins [17] is considered in this analysis. The transport parameters used in the problem are summarized in Table 5. The performance of the modified-Domenico solution was tested by comparing its results against those generated using the exact solution given by Wexler [48].

It has been established in the previous sections that the Domenico approximation makes unreasonable assumptions regarding the residence time beyond the advective front, and reasonable assumptions behind the front. Therefore we analyze the results of this comparison in two parts-- one behind the advective front and the other beyond the advective front (note that for our base case, the front is at $x = 1100$ meters).

4.7.1 Plume Comparison Analysis Behind the Advective Front

Figures 5a and 5b compare the two-dimensional concentration contours of both solutions on the X-Y and X-Z planes (Note: an aspect ratio of '2.2 : 1' was maintained for the X-Y plots and an aspect ratio of '55 : 1' was maintained for the X-Z plots). Since the problem is symmetric about the X axis, only half of the plume is presented. It can be observed from Figure 5 that the modified-Domenico solution is reasonably close to the true solution, though there are some noticeable discrepancies. To explore the limits of these discrepancies, we performed a series of sensitivity simulations.

Parameter	Value
Longitudinal dispersivity (α_x)	42.58 m
Transverse dispersivity (α_y)	8.43 m
Transverse dispersivity (α_z)	0.00642 m
Velocity (v)	0.2151 m/day
Source width in Y direction (Y _o)	240.0 m
Source width in Z direction (Z _o)	5.0 m
Source concentration (C _o)	850 mg/L
Simulation Time (T)	5110 days

Table5: Parameters used in the Domenico and Robbins [17] example problem

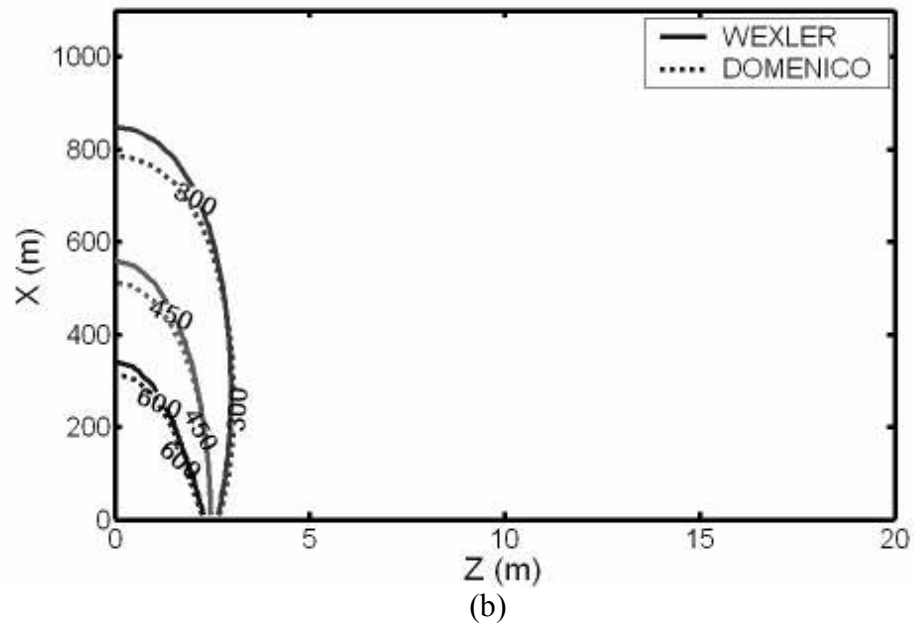
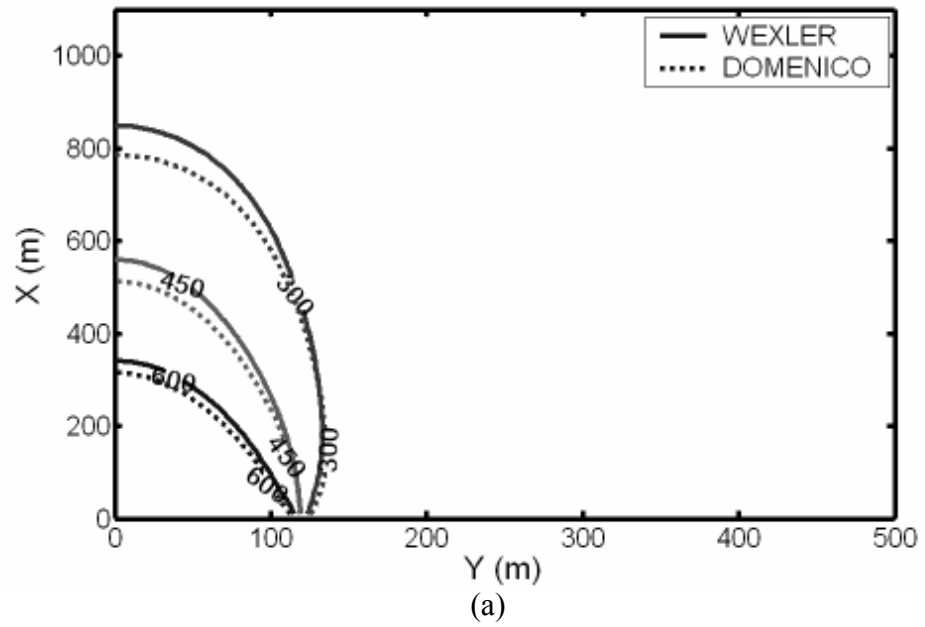


Fig. 5: Concentration contours predicted by the Domenico and Wexler solutions for the base case: solutions behind the advective front for (a) X-Y plane (b) X-Z plane

In the first set of sensitivity simulations, we varied the value of the longitudinal dispersivity (α_x) by an order of magnitude above and below the assumed baseline value. Figures 6a and 6b compare the two-dimensional concentration contours of the solutions for both the cases. Comparison of the data shown in Figures 5a and 6 indicate that the discrepancies between the two solutions were large when the value of longitudinal dispersivity was high. Also, as expected, when the longitudinal dispersivity was low there was an excellent match between the solutions. Similar trends were also observed in the concentration contours predicted on the X-Z plane. Since the spreading terms in the y and z directions are identical in structure, the contours in the X-Y and X-Z planes will exhibit identical trends. Therefore, from this point onwards the sensitivity analysis will be restricted to X-Y contours.

In the second set of sensitivity simulations, we varied the value of the transverse dispersivity (α_y) by an order of magnitude above and below the base line value. These results (refer supplementary material for figures) indicated that the transverse dispersivity in the y-direction does not play a significant role in influencing the error associated with the modified-Domenico solution. Similar sensitivity analysis performed on other transport parameters including the transverse dispersivity α_z and the source dimensions Y and Z also showed minimal sensitivity.

A third set of sensitivity simulations were completed for a decaying contaminant plume by assuming various first-order rate coefficients (k). Comparison of the concentration contours for k values of 0.0001 day^{-1} and 0.001 day^{-1} indicated that the presence of a decay term does not introduce any significant additional error [41].

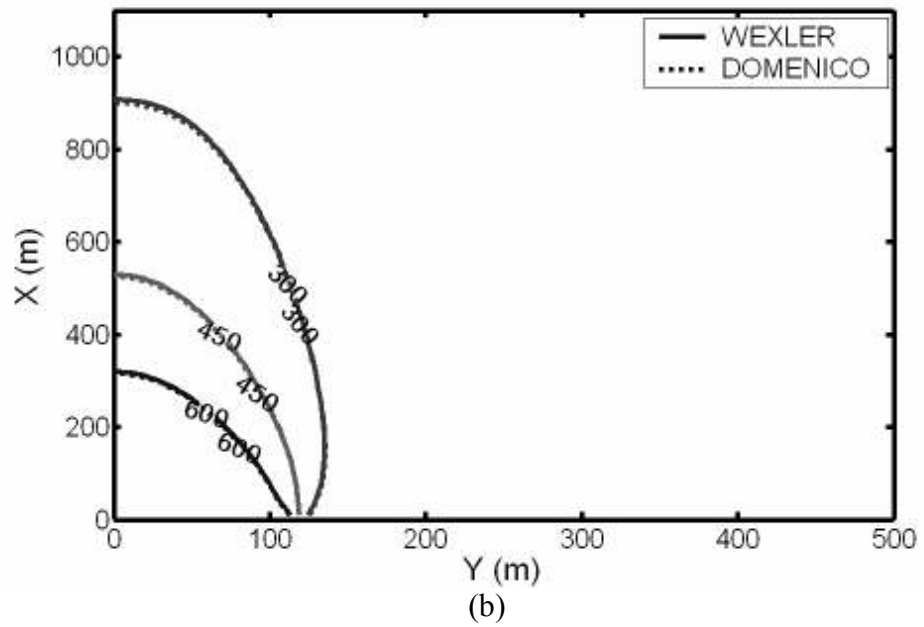
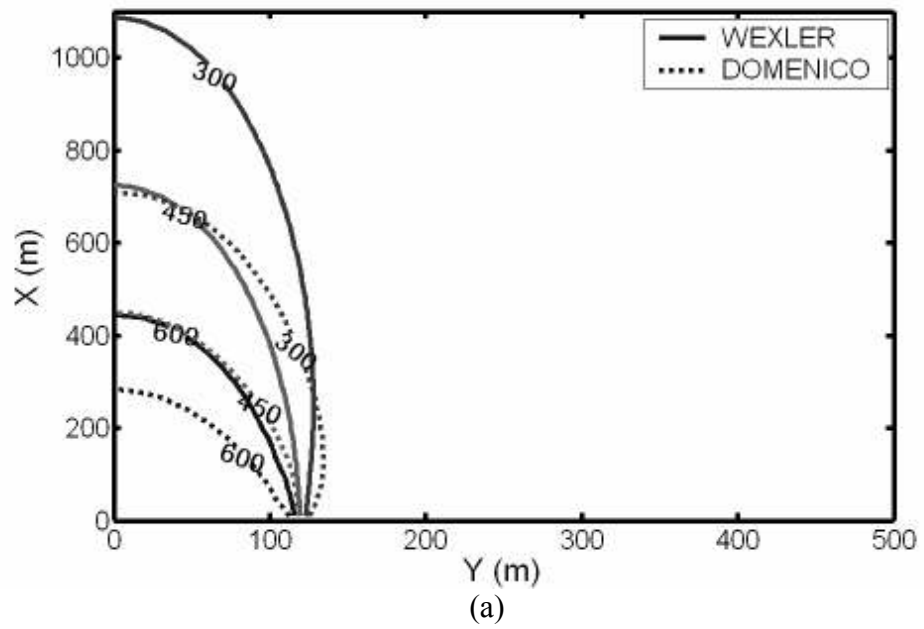


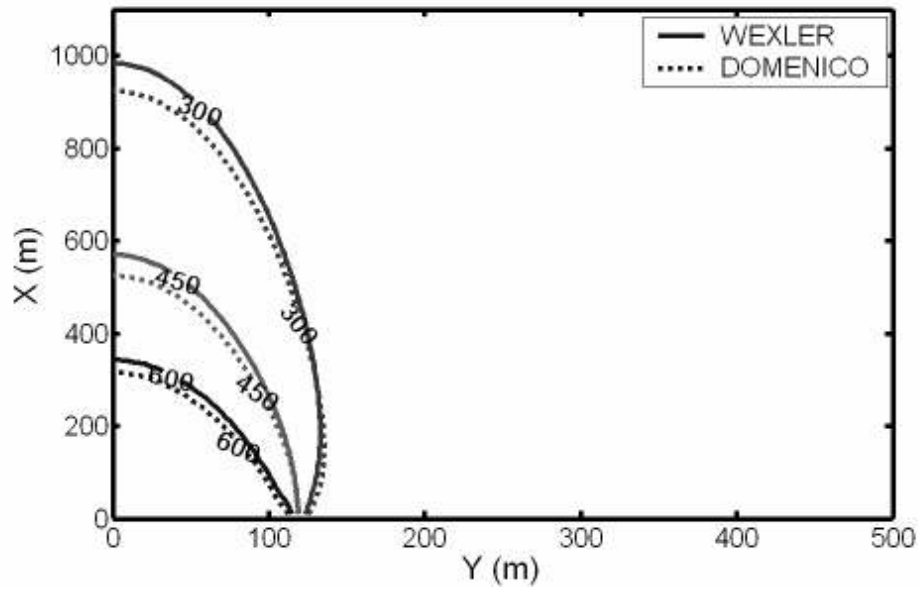
Fig. 6: Sensitivity results for variations in the longitudinal dispersivity value; solutions behind the advective front for (a) $\alpha_x \cdot 10$ (b) $\alpha_x / 10$

The fourth set of sensitivity simulations involved varying the advection velocity (v) by an order of magnitude above and below the baseline value. Figures 7a and 7b compare the concentration contours of the solutions for the two cases. From these figures it can be concluded that the advection velocity has very little effect in determining the accuracy of the solution. Note that in the absence of first order decay, varying the advection velocity will have the same effect as varying the total simulation time (t). Since decay does not play any significant role in determining the accuracy of the modified-Domenico solution, it can be safely concluded that variations in the total simulation time will have little sensitivity on its accuracy.

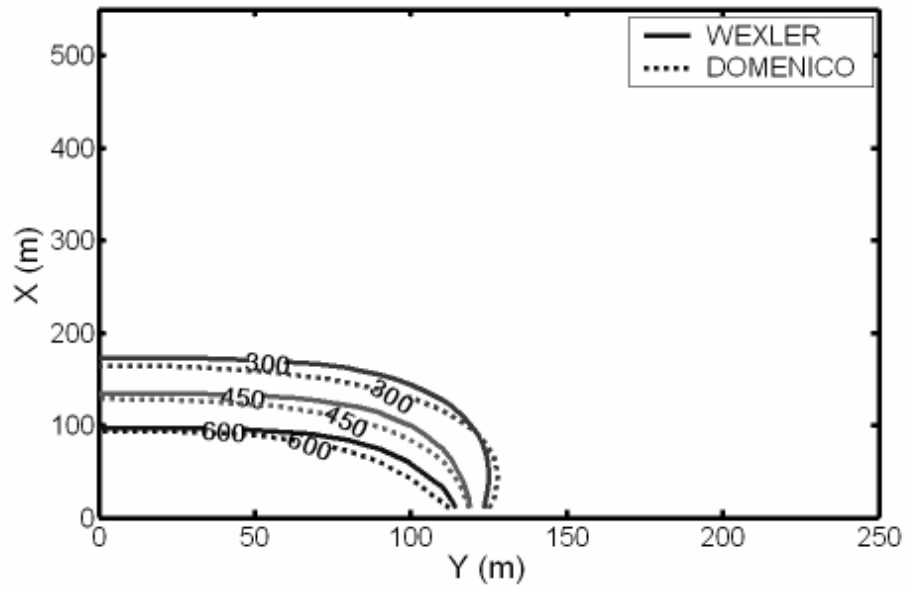
The above results indicate that within the advective front the longitudinal dispersivity plays a very important role in determining the accuracy of the modified-Domenico solution. All the other transport parameters have negligible effect on the accuracy of the solution.

4.7.2 Plume Comparison Analysis Beyond the Advective Front

Figure 8 compares the concentration contours of the two solutions in the X-Y plane for the base case parameters (Note: here an aspect ratio of '4:1' is maintained for the X:Y plane to capture the plume beyond the advective front; also the location of the advective front is indicated by an arrow on the x axis). It can be observed from Figure 8 that, as we move beyond the advective front, the accuracy of the modified-Domenico solution reduces rapidly. As pointed out in the earlier sections, this is due to the unrealistic assumptions made by the Domenico solution when computing the conceptual residence times beyond the advective front.



(a)



(b)

Fig. 7: Sensitivity results for variations in the transport velocity; solutions behind the advective front for (a) $v \cdot 10$ (b) $v/10$

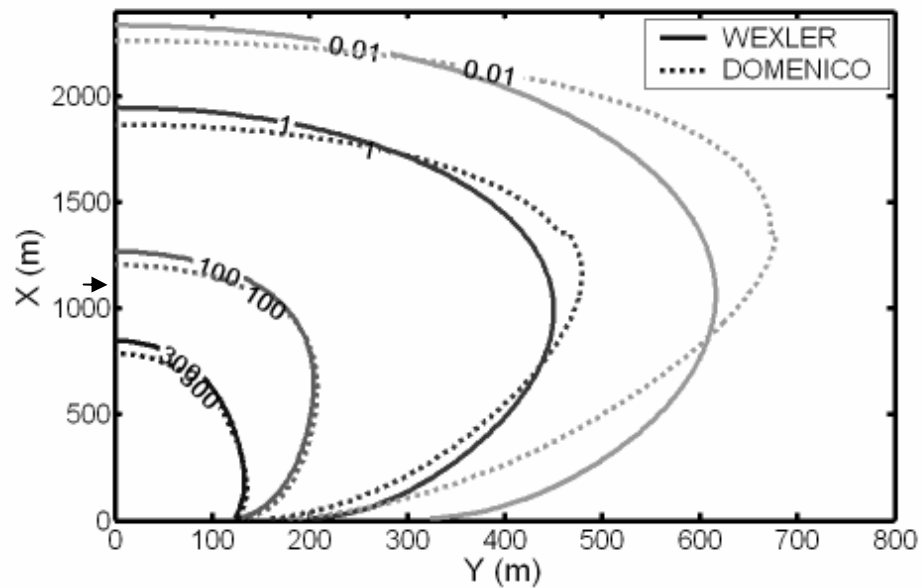


Fig. 8: Concentration contours predicted by the Domenico and Wexler solutions; solutions include concentration contours beyond the advective front

The results of a sensitivity analysis performed on the parameter α_x are summarized in Figures 9a and 9b. These figures indicate a trend similar to those present for regions within the advective front. Higher the value of the longitudinal dispersivity, greater the error associated with the modified-Domenico solution. Further, it can be observed that the error systematically increases when the contaminant is transported beyond the advective front.

A similar set of sensitivity simulations were performed on different transport parameters including α_y , α_z , Y, Z and K for regions beyond the advective front as well. As expected, the results indicated that these parameters had negligible contribution in determining the accuracy of the solution.

A final set of sensitivity simulations were performed by varying the value of the advection velocity (v) by an order of magnitude above and below the base line value. The results of this sensitivity analysis are summarized in Figures 10a and 10b. Initial observations of these figures may indicate that at higher velocities, the modified-Domenico solution appears to perform better. However, a closer analysis of these figures with respect to their respective advective front locations indicates that at higher velocities a greater portion of the plume is behind the advective front whereas, at lower velocities a relatively lesser portion of the plume is behind the advective front confirming that the advection velocity has little effect in determining the accuracy of the solution. However, it must be noted that the advection velocity itself plays an important role in determining the location of the advective front, which is one of the key parameters that affects the performance of the solution. Variations in the total simulation time (t) will have a similar effect as that of the advection velocity.

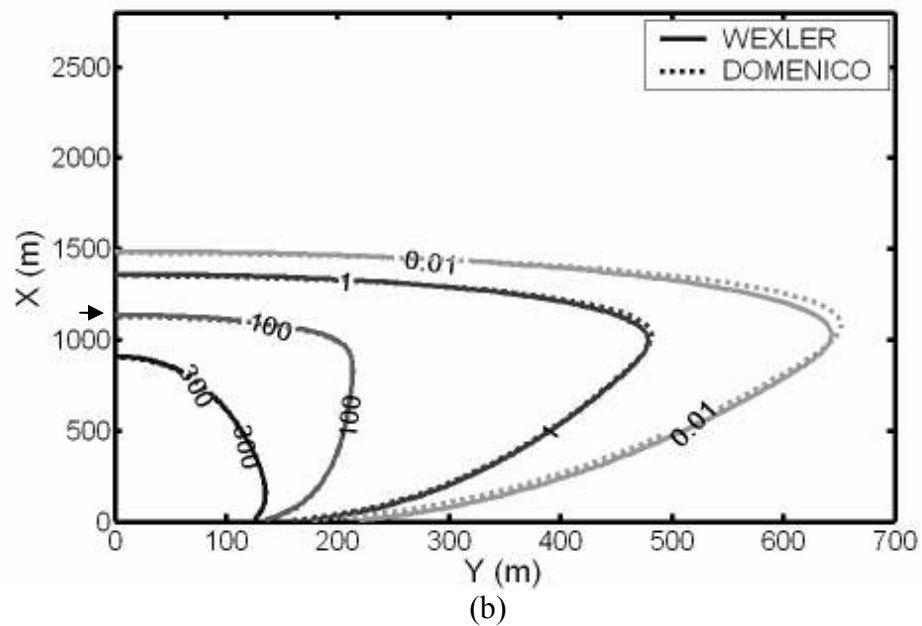
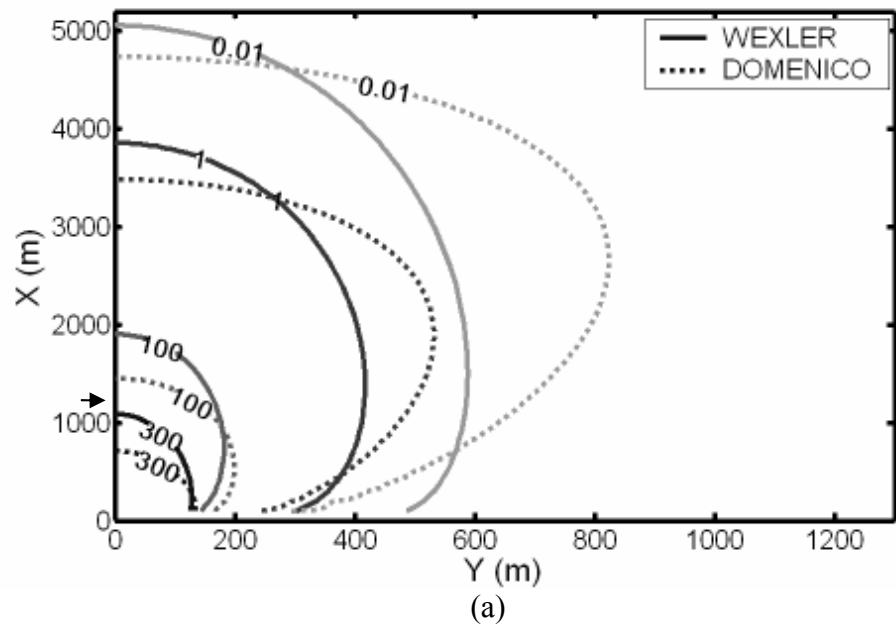


Fig. 9: Sensitivity results for variations in the longitudinal dispersivity value; solutions include concentration contours beyond the advective front for (a) $\alpha_x \cdot 10$ (b) $\alpha_x / 10$

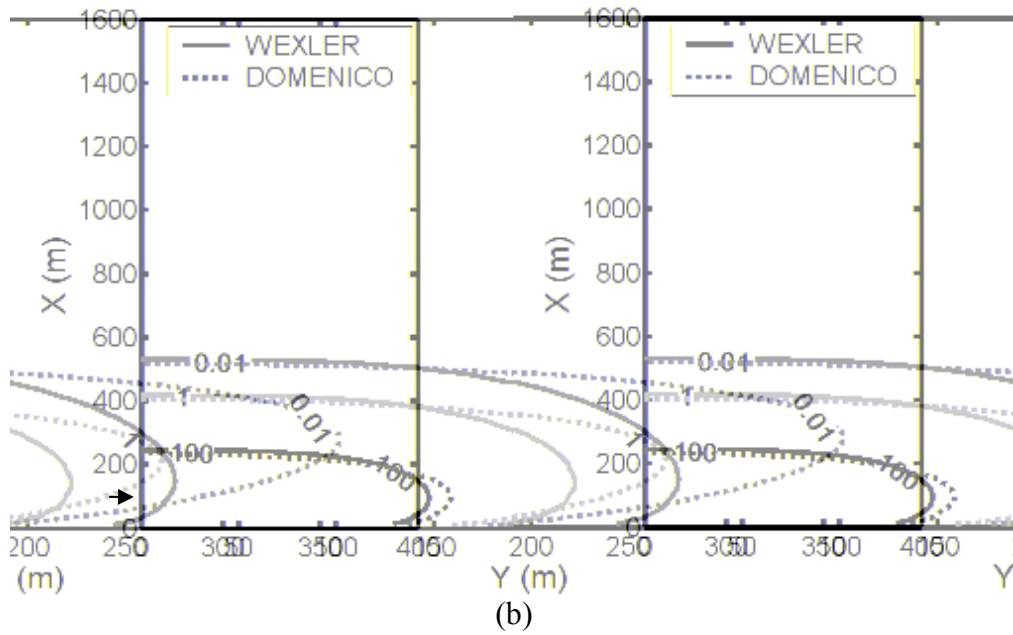
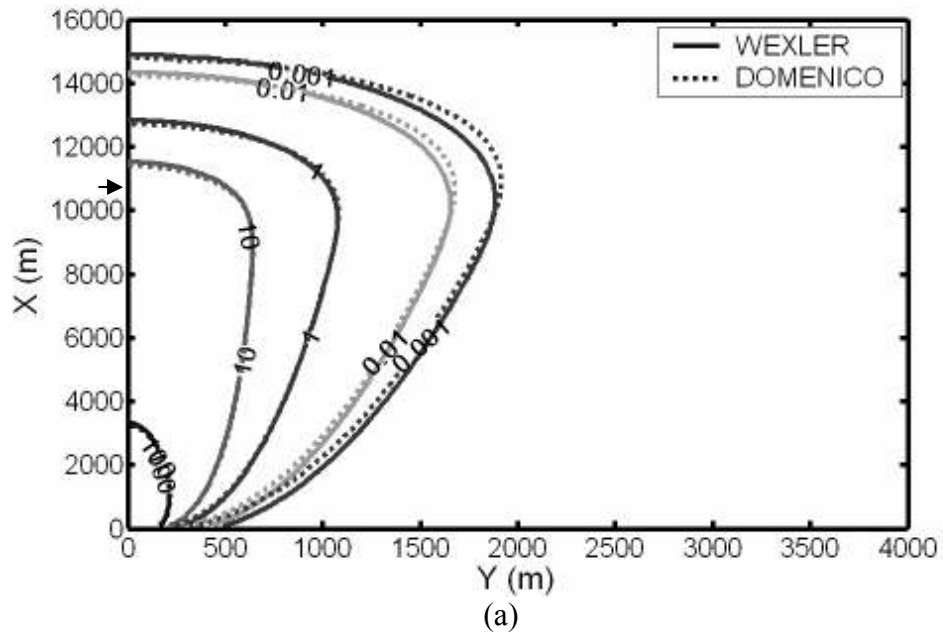


Fig. 10: Sensitivity results for variations in the transport velocity; solutions include concentration contours beyond the advective front for (a) $v \cdot 10$ (b) $v/10$

From the results of these sensitivity simulations, it can be safely concluded that, the two most important factors that affect the accuracy of the modified-Domenico solution are the value of the longitudinal dispersivity (α_x) and the position of the advective front ($v \cdot t$). The solution will have minimal errors when the value of α_x is low and when the advective front is farther away from the source. It must be noted that the conclusions obtained for the modified-Domenico solution apply to the Domenico solution as well, provided the value of $\frac{x}{\alpha_x}$ is sufficiently large [5].

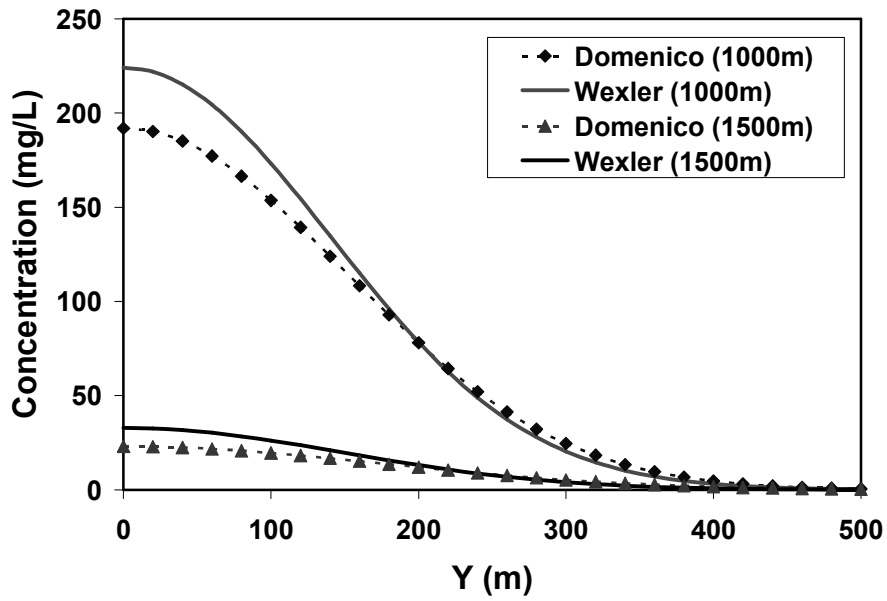
4.8 Discussions and Recommendations

Since the original Domenico solution lacked a theoretical basis, several misconceptions regarding its performance have evolved over the years. One of the common misconceptions is that the error will be a minimum along the plume centerline. For example, Guyonnet and Neville [22] compared the Domenico solution against the Sagar [39] solution and concluded that *“The results of the evaluation confirm that along the plume centerline, and for ground water flow regimes dominated by advection and mechanical dispersion rather than by molecular diffusion, discrepancies between the two solutions (namely the Domenico solution and the Sagar solution) can be considered negligible for all practical purposes. However the errors in the Domenico (1987) solution may increase significantly outside the plume centerline.”* However, the above simulation results indicate that this conclusion might not be true for all cases. To illustrate this, compare the y and z concentration transects predicted by the two solutions for our base case scenario. Figure 11a compares the concentration profiles along y

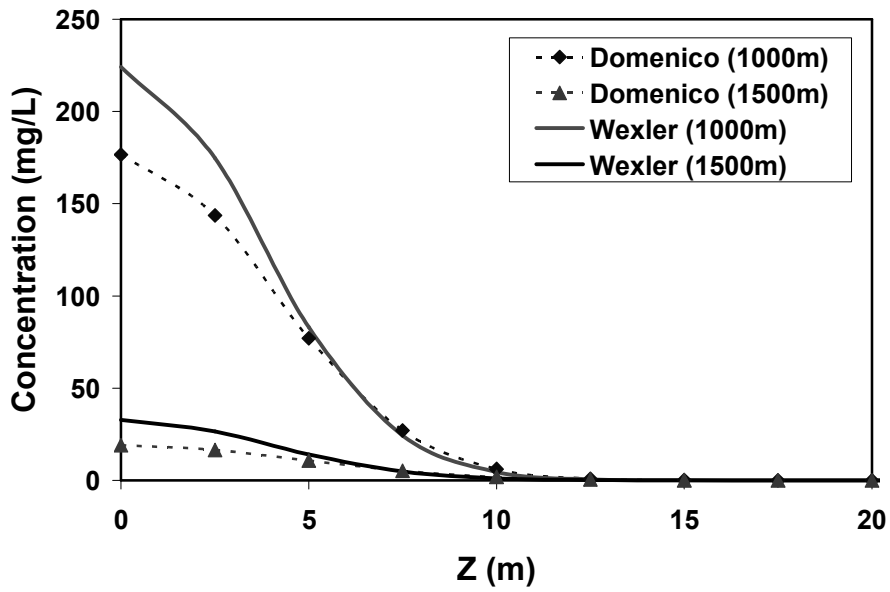
direction at $x = 1000$ m and 1500 m, and similar results for the z direction are shown in Figure 11b. It is evident from these figures that the error is not minimal along the centerline, but rather at a point which will always be away from the centerline. This error pattern can also be observed in all the two dimensional contours. Further, it can be observed from Figures 11a and 11b that the absolute error is, in fact, maximum along the plume centerline.

Another important issue that should be addressed here is the nature in which the error associated with the Domenico approximation is propagated spatially. The sensitivity results presented in this chapter show that the plumes predicted by the modified-Domenico solution are always wider than the actual plumes. This phenomenon can easily be observed in all the figures presented in this study. This can be attributed to the fact that the Domenico approximation over predicts the conceptual residence times of all particles along the centerline (hence allows more time to disperse in the transverse directions). This over prediction would lead to a decrease in the centerline concentrations; therefore, solutions that employ the Domenico approximation will always under predict the overall extent of the plume in the longitudinal direction.

An important transport parameter not addressed so far is the retardation factor (R). Retardation affects the advection velocity and possibly the decay constant (depending on the phase where the decay occurs). Since the presence of a decay term does not introduce any significant additional error to the Domenico solution, its effect can be ignored. However, retardation changes the location of the advective front by changing the advection velocity and hence would influence the overall accuracy of the Domenico solution.



(a)



(b)

Fig. 11: Concentration profiles predicted by the Domenico solution compared with the Wexler solution at $x = 1000\text{m}$ and 1500m (a) along Y axis (b) along Z axis

Based on the theoretical results presented in this study, it can be concluded that the key assumption used to derive the Domenico solution is the time reinterpretation step where the time τ in the transverse dispersion terms is replaced with x/v . The derivations presented in section 4.5 prove that this substitution process is valid only when the longitudinal dispersivity is zero. For all non-zero longitudinal dispersivity values the solution will have a finite error. The spatial distribution of this error is highly sensitive to the value of α_x and the position of the advective front ($v*t$), and is relatively less sensitive to other transport parameters. Based on the results of this study, it can be concluded that the error in the Domenico solution will be low when solving transport problems that have low longitudinal dispersivity values, high advection velocities, and large simulation times.

Despite its limitations, the Domenico approximation offers a simple alternative for extending one dimensional analytical solutions to three dimensions. This approach is useful for developing approximate solutions for unsolved, three dimensional, multi-species reactive transport problems that have explicit one dimensional solutions. However, such solutions should be used carefully after understanding the limitations identified in this study.

CHAPTER V

SUMMARY AND CONCLUSIONS

In this research a set of analytical solutions to multi-species reactive transport equations coupled through sorption and sequential first-order reactions is presented. In the first part of this work, the mathematical derivations of the general solution which incorporates a generic Bateman type exponentially decaying Dirichlet and Cauchy source boundary conditions and a spatially varying initial condition are presented. The solution strategy involves uncoupling the system of governing equations through the use of transformations and then solving the uncoupled system of equations individually to obtain independent solutions for each species in the transformed domain. A combination of Laplace and linear transforms were used to uncouple the system of equations. Finally the independent solutions in the transformed domain are analytically re-transformed back to the original time domain to obtain the final solutions. Some of the key challenges in performing these analytical operations include formulation of the generic linear transform matrices and evaluation of the inverse Laplace expressions.

The solutions to both the Dirichlet and Cauchy boundaries are presented in a common format to enable simultaneous implementation of both these solutions. Furthermore, the solutions are presented in a closed form format that avoids numerical integrations processes. Solutions for two cases of sorption; one involving decay in liquid

phase only and the other involving decay in both liquid and solid phases are also presented.

In the second part of this research, the computational techniques for implementing the solution are discussed and the general solution is validated against other published solutions. Some of the key challenges faced when implementing the solutions are round-off errors and underflow/overflow errors. A combination of techniques involving algebraic manipulation and logarithmic transforms are suggested to tackle these errors. A FORTRAN computer code “SEQUENTIAL” that solved the general solution was developed and is provided in Appendix I. The new code was used to simulate four example problems and the results generated by the code matched the results of previously published analytical and semi-analytical solutions.

In addition, several special-case solutions for simpler transport problems involving zero initial condition, identical retardation factors, zero advection, zero dispersion and steady-state condition are also presented. Where ever possible these special case solutions were successfully verified against previously published analytical solutions. Strategies for extending the one dimensional sequential solutions to a generic diverging reaction network and to multiple dimensions (using the approximate Domenico solution) are also presented. A detailed investigation into the errors involved in this approximation is also discussed. The solutions proposed in this research work can be used to develop efficient screening tools for assessing ground water quality issues at sites contaminated with radioactive materials or chlorinated solvents.

One of the key limitations of the general solution is that it can model sequential/diverging reaction networks only. More generic reaction networks involving

converging and/or reversible reaction networks cannot be modeled using the general solution. The main challenge in incorporating a more generic reaction network is obtaining the analytical linear transform matrices. Simplifying the problem to steady state may perhaps help in reducing the complexity in obtaining the matrices.

It must be noted that Laplace transform techniques were used in this study to uncouple the system of equations. The use of Fourier transforms for this purpose may possibly help in easing some of the other limitations on the general solution including the restriction on the transport parameters (See Table 4) and the limitation to modeling first-order kinetics. Future research efforts in this area should focus on implementing more robust mathematical induction techniques and using Fourier transforms to obtain solutions to more general multi-species transport problems.

REFERENCES

- [1] Angelakis AN, Kadir TN, Rolston DE. Analytical solutions for equations describing coupled transport of 2 solutes and a gaseous product in soil. *Water Resources Research* 1993;29:945-956.
- [2] Angelakis AN, Kadir TN, Rolston DE. Solutions for transport of 2 sorbed solutes with differing dispersion coefficients in soil. *Soil Science Society of America Journal* 1987;51:1428-1434.
- [3] Aziz CE, Newell CJ, Gonzales JR, Hass P, T.P C, Sun Y. BIOCHLOR: Natural attenuation decision support system V.1.0. User's manual. US EPA Report, EPA/600/R-00/008. Cincinnati, Ohio. 2000;
- [4] Bauer P, Attinger S, Kinzelbach W. Transport of a decay chain in homogenous porous media: analytical solutions. *Journal of Contaminant Hydrology* 2001;49:217-239.
- [5] Bear J. *Hydraulic of Groundwater*. McGraw-Hill, New York 1979;268-269.

- [6] Bruch JC, Street RL. Two-dimensional dispersion. Journal of the Sanitary Engineering Division Proceeding of the American Society of Civil Engineers 1967;93:17-39.
- [7] Burkholder HC, Rosinger ELJ. A model for the transport of radionuclides and their decay products through geologic media. Nuclear Technology 1980;49:150-158.
- [8] Carslaw HS, Jaeger JC. Conduction of heat in solids. Oxford, Great Britan: Oxford university Press; 1980.
- [9] Cho CM. Convective transport of ammonium with nitrification in soil. Canadian Journal of Soil Science 1971;51:339-350.
- [10] Cleary R, Unga. MJ. Analytical models for groundwater pollution and hydrology, Water Resources Program, Princeton University, 1978.
- [11] Clement TP. Generalized solution to multispecies transport equations coupled with a first-order reaction network. Water Resources Research 2001;37:157-163.
- [12] Clement TP, Sun Y, Hooker BS, Petersen JN. Modeling multispecies reactive transport in ground water. Ground Water Monitoring and Remediation 1998;18:79-92.

- [13] Clement TP, Truex MJ, Lee P. A case study for demonstrating the application of U.S. EPA's monitored natural attenuation screening protocol at a hazardous waste site. *Journal of Contaminant Hydrology* 2002;59:133-162.
- [14] Cody WJ. Rational Chebyshev approximations for the error function. *Mathematics of Computation* 1969;23:631-638.
- [15] Crank J. *The mathematics of diffusion*. New York: Oxford university press; 1975.
- [16] Domenico PA. An analytical model for multidimensional transport of a decaying contaminant species. *Journal of Hydrology* 1987;91:49-58.
- [17] Domenico PA, Robbins GA. A new method of contaminant plume analysis. *Ground Water* 1985;23:476-485.
- [18] Eykholt GR, Li L. Fate and transport of species in a linear reaction network with different retardation coefficients. *Journal of Contaminant Hydrology* 2000;46:163-185.
- [19] Gautschi W. Efficient computation of the complex error function. *SIAM Journal on Numerical Analysis* 1970;7:187-198.
- [20] Gautschi W. *Error function and fresnel integrals*. Washington D.C.: U.S. Government Printing Office; 1964.

- [21] Gureghian AB, Jansen G. One-dimensional analytical solutions for the migration of a 3-member radionuclide decay chain in a multilayered geologic medium. *Water Resources Research* 1985;21:733-742.
- [22] Guyonnet D, Neville C. Dimensionless analysis of two analytical solutions for 3-D solute transport in groundwater. *Journal of Contaminant Hydrology* 2004;75:141-153.
- [23] Harada M, Chambre PL, Foglia M, Higashi K, Iwamoto F, Leung D, Pigford TH, Ting D. Migrations of radionuclides through sorbing media, analytical solutions - 1. Report no. LBL-10500 (UC-70, Lawrence Berkely Laboratory, University of California Berkely 1980;
- [24] Higashi K, Pigford TH. Analytical Models for Migration of Radionuclides in Geologic Sorbing Media. *Journal of Nuclear Science and Technology* 1980;17:700-709.
- [25] Hunt B. Dispersive sources in uniform ground-water flow. *Journal of the Hydraulic Division* 1978;104:75-85.
- [26] Jones NL, Clement TP, Hansen CM. A three-dimensional analytical tool for modeling reactive transport. *Ground Water* 2006;44:613-617.

- [27] Khandelwal A, Rabideau AJ. Transport of sequentially decaying reaction products influenced by linear nonequilibrium sorption. *Water Resources Research* 1999;35:1939-1945.
- [28] Kreyszig E. *Advanced engineering mathematics*. New York, USA: John Wiley & Sons; 2004.
- [29] Lester DH, HJansen G, Burkholder HC. Migration of radionuclide chains through an adsorbing medium. *Adsorption and Ion Exchange, AICHE Symposium Series 152*, American Institute of Chemical Engineers, New York 1975;71:
- [30] Lunn M, Lunn RJ, Mackay R. Determining analytic solutions of multiple species contaminant transport, with sorption and decay. *Journal of Hydrology* 1996;180:195-210.
- [31] Martyn-Hayden J, Robbins GA. Plume distortion and apparent attenuation due to concentration averaging in monitoring wells. *Ground Water* 1997;35:339-346.
- [32] McLaren AD. Nitrification in soil - systems approaching a steady state. *Soil Science Society of America Proceedings* 1969;33:551-&.
- [33] McLaren AD. Temporal and vectorial reactions of nitrogen in soil - a review. *Canadian Journal of Soil Science* 1970;50:97-&.

- [34] Misra C, Nielsen DR, Biggar JW. Nitrogen transformations in soil during leaching .1. Theoretical considerations. Soil Science Society of America Journal 1974;38:289-293.
- [35] Montas HJ. An analytical solution of the three-component transport equation with application to third-order transport. Water Resources Research 2003;39:
- [36] Newell CJ, McLeod RK, Gonzales JR. BIOSCREEN: Natural attenuation decision support system user's manual EPA/600/R-96/087, Robert S. Kerr Environmental Research Center, 1996.
- [37] Quezada CR. Generalized Solution to Multi-Species Transport Equations Coupled with a First-Order reaction Network with Distinct Retardation factors, Auburn University, Auburn, AL, USA, 2004.
- [38] Quezada CR, Clement TP, Lee KK. Generalized Solution to Multi-dimensional Multi-species Transport Equations Coupled With a First-order Reaction Network Involving Distinct Retardation Factors. Advances in Water Resources 2004;27:508-521.
- [39] Sagar B. Dispersion in three dimensions: Approximate analytical solutions. Journal of the Hydraulic Division, Proceeding of the American Society of Civil Engineers. 1982;108:47-62.

- [40] Selby SM. Standard mathematical tables. Cleveland, Ohio: The Chemical Rubber Co.; 1971.
- [41] Srinivasan V, Clement TP, Lee KK. Domenico solution - Is it valid? Ground Water 2007;45:136-146.
- [42] Sun Y, Petersen JN, Clement TP. Analytical solutions for multiple species reactive transport in multiple dimensions. Journal of Contaminant Hydrology 1999;35:429-440.
- [43] Sun Y, Petersen JN, Clement TP, Skeen RS. Development of analytical solutions for multispecies transport with serial and parallel reactions. Water Resources Research 1999;35:185-190.
- [44] Tim US, Mostaghimi S. Modeling transport of a degradable chemical and its metabolites in the unsaturated zone. Ground Water 1989;27:672-681.
- [45] van Genuchten MT. Convective-dispersive transport of solutes involved in sequential 1st-order decay reactions. Computers and Geosciences 1985;11:129-147.
- [46] Wark K, Warner CF. Air Pollution, Its Origin and Control. New York: Addison Wesley Longman Inc; 1981.

[47] West M, Kueper BH. Natural attenuation of solute plumes in bedded fractured rock. USEPA/NGWA Fractured Rock Conference, Portland, Maine, National Ground Water Association 2004;388–401.

[48] Wexler EJ. Analytical solutions for one-, two-, and three-dimensional solute transport in ground-water systems with uniform flow, USGS—TWRI Book 3, 1992.

APPENDIX A

Factorization of the Laplace Terms for the Dirichlet Boundary Condition

The first term ‘ $\ell^{-1} \langle p_i^1(x, s) \rangle$ ’, can be evaluated as follows. From equations (41)

and (42) we get:

$$c_i^1(x, t) = \ell^{-1} \langle p_i^1(x, s) \rangle = \ell^{-1} \left\langle \sum_{i_1=1}^i \frac{\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_o(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})}}{e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]} \prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})} \right\rangle$$

; $\forall i = 1, 2, \dots, n$
(A.1)

Equation(A.1) can be rewritten as:

$$c_i^1(x,t) = \ell^{-1} \left\langle \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_1=1}^i \frac{\sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} B_{i_1}^{i_3} \left\{ 1 - e^{-t_0(s+\lambda_{i_3})} \right\} e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]}}{(s + a_{i_3,0}) \prod_{i_4=i_1, (i_4 \neq i_2)}^i -R_{i_2, i_4} (s + a_{i_2, i_4})} \right] \right\rangle$$

$; \forall i = 1, 2, \dots, n$

where ; $k_{i_1, i_2} = k_{i_1} - k_{i_2}$, $R_{i_1, i_2} = R_{i_1} - R_{i_2}$ and $a_{i_1, i_2} = \begin{cases} k_{i_1, i_2} & ; \text{when } i_2 > 0 \\ R_{i_1, i_2} & \\ \lambda_{i_1} & ; \text{when } i_2 = 0 \end{cases}$

(A.2)

It must be noted that for equation(A.2) to be valid the condition ‘ $R_{i_2, i_4} \neq 0$ ’ must be satisfied. This means that no two species in the transport problem can have identical retardation factors. However, in practice, we do have situations where the retardation factors of some of the species are equal [9]. To overcome this limitation, we reformulate equation(A.2) to accommodate a generic case when the transport problem has any number of sets of species having any number of species with identical retardation factors. Incorporating this special case scenario, equation(A.2) is reformulated as:

$$c_i^1(x, t) = \ell^{-1} \left\langle \sum_{i_1=1}^i \left[\frac{\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) B_{i_1}^{i_3} \left\{ 1 - e^{-t_o(s+\lambda_{i_3})} \right\} e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]}}{\sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} (s + a_{i_3,0}) \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) \prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (s + a_{i_2, i_4})} \right] \right\rangle \quad (\text{A.3})$$

; $\forall i = 1, 2, \dots, n$

Note that in equation(A.3) the condition ‘ $k_{i_2, i_4} \neq 0$ ’ must be satisfied for the solution to be determinate. Factorization

101

of equation(A.3) gives:

$$c_i^1(x,t) = \ell^{-1} \left[\sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \frac{B_{i_1}^{i_3} \left\{ 1 - e^{-t_o(s+\lambda_{i_3})} \right\} e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]}}{(s + a_{i_3,0}) \prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4}=R_{i_2})}^i -k_{i_2, i_4}} \right. \right. \\ \left. \left. \sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i \frac{1}{-R_{i_2, i_4} (s + a_{i_2, i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2, i_5} (a_{i_2, i_5} - a_{i_2, i_4})} \right] \right] \quad ; \forall i = 1, 2, \dots, n \quad (\text{A.4})$$

102

It must be noted that the solution formulation as given by equation(A.4) is valid only when the condition ‘ $a_{i_2, i_5} \neq a_{i_2, i_4}$ ’, is satisfied. Equation(A.4) can be further factorized and simplified as:

$$c_i^1(x, t) = \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \{ G_1^1 + h(G_1^1) G_2^1 \} \right]$$

; $\forall i = 1, 2, \dots, n$

where ;

$$G_1^1 = \frac{\sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i B_{i_1}^{i_3} e^{\frac{xv}{2D_x}} \ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_0(s+\lambda_{i_3})} \right\} e^{\left[\frac{-x}{2D_x} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right]}}{(s + a_{i_3,0})(s + a_{i_2,i_4})} \right\rangle}{\left(\prod_{i_5=i_1, (i_5 \neq i_2, R_{i_5} = R_{i_2})}^i -k_{i_2,i_5} \right) (-R_{i_2,i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2,i_5} (a_{i_2,i_5} - a_{i_2,i_4})}$$

$$G_2^1 = \frac{B_{i_1}^{i_3} e^{\frac{xv}{2D_x}} \ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_0(s+\lambda_{i_3})} \right\} e^{\left[\frac{-x}{2D_x} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right]}}{(s + a_{i_3,0})} \right\rangle}{\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2,i_4}}$$

(A.5)

$$h(M) = \begin{cases} 1 & \text{if } M \text{ loop is not executed} \\ 0 & \text{if } M \text{ loop is executed} \end{cases}$$

The term ‘ G_1^1 ’ in equation(A.5) can be further factorized and simplified as:

$$G_1^1 = \sum_{i_4=i_1, (i_4 \neq i_2, R_{i_5}=R_{i_2})}^i B_{i_1}^{i_3} e^{\frac{xv}{2D_x}} \left[\begin{array}{c} \ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_0(s+\lambda_{i_3})} \right\} e^{\left[\frac{-x}{2D_x} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right]}}{(s + a_{i_3,0})} \right\rangle \\ -\ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_0(s+\lambda_{i_3})} \right\} e^{\left[\frac{-x}{2D_x} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right]}}{(s + a_{i_2,i_4})} \right\rangle \end{array} \right] \quad (A.6)$$

$$\frac{\left(a_{i_2,i_4} - \lambda_{i_3} \right) \left(\prod_{i_5=i_1, (i_5 \neq i_2, R_{i_5}=R_{i_2})}^i -k_{i_2,i_5} \right) (-R_{i_2,i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2,i_5} (a_{i_2,i_5} - a_{i_2,i_4})}{}$$

Again it must be noted that equation(A.6) is valid only when the condition ‘ $a_{i_2,i_4} \neq a_{i_3,0}$ ’ where ‘ $a_{i_3,0} = \lambda_{i_3}$ ’ is satisfied. Using the results from Appendix B, inverse Laplace expressions for the terms ‘ G_1^1 ’ and ‘ G_2^1 ’ can be evaluated and the solution for ‘ $c_i^1(x, t)$ ’ is obtained as:

$$c_i^1(x, t) = \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \{ G_1^1 + h(G_1^1) G_2^1 \} \right]$$

; $\forall i = 1, 2, \dots, n$

$$G_1^1 = \frac{\sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i \left(a_{i_2, i_4} - \lambda_{i_3} \right) \left(\prod_{i_5=i_1, (i_5 \neq i_2, R_{i_5} = R_{i_2})}^i -k_{i_2, i_5} \right) \left(-R_{i_2, i_4} \right) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2, i_5} \left(a_{i_2, i_5} - a_{i_2, i_4} \right)}{B_{i_1}^{i_3} \left\langle F_{i_2, i_3, 0} [x, t] - u(t - t_o) e^{(-\lambda_{i_3} t_o)} F_{i_2, i_3, 0} [x, (t - t_o)] \right\rangle}$$

$$G_2^1 = \frac{B_{i_1}^{i_3} \left\langle F_{i_2, i_3, 0} [x, t] - u(t - t_o) e^{(-\lambda_{i_3} t_o)} F_{i_2, i_3, 0} [x, (t - t_o)] \right\rangle}{\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4}}$$

(A.7)

where; the term ' F_{i_1, i_2, i_3} ' is given by equation (B.8) or (B.14) in Appendix B. The second term ' $\ell^{-1} \langle p_i^2(x, s) \rangle$ ' is

evaluated as follows. From equations (41) and (42) we get:

$$\begin{aligned}
c_i^2(x, t) &= \ell^{-1} \langle p_i^2(x, s) \rangle \\
&= \ell^{-1} \left\langle \sum_{i_1=1}^i \frac{R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left\{ e^{\left[\frac{x}{2D_x} \left(v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right) \right]} - e^{-\mu_{i_1} x} \right\}}{\sum_{i_2=i_1}^i \left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_{i_2} - sR_{i_2} \right) \prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})} \right\rangle \quad (\text{A.8}) \\
&\quad ; \forall i = 1, 2, \dots, n
\end{aligned}$$

106

Equation(A.8) can be written as:

$$\begin{aligned}
c_i^2(x, t) &= \ell^{-1} \left\langle \sum_{i_1=1}^i \frac{R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left\{ e^{\left[\frac{x}{2D_x} \left(v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right) \right]} - e^{-\mu_{i_1} x} \right\}}{\sum_{i_2=i_1}^i \left[-(s + a_{i_1, -i_2}) R_{i_2} \left(\prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3} \right) \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i -R_{i_2, i_3} (s + a_{i_2, i_3}) \right]} \right\rangle \quad (\text{A.9}) \\
&\quad ; \forall i = 1, 2, \dots, n
\end{aligned}$$

Note: the term ‘ a_{i_1, i_2} ’ is modified as:

$$a_{i_1, i_2} = \begin{cases} \frac{k_{i_1, i_2}}{R_{i_1, i_2}} ; \text{ when } i_2 > 0 \\ \lambda_{i_1} ; \text{ when } i_2 = 0 \\ \frac{-\mu_{i_1}^2 D_x - \mu_{i_1} v + k_{i_2}}{R_{i_2}} ; \text{ when } i_2 < 0 \end{cases} \quad (\text{A.10})$$

Note that the condition ‘ $k_{i_2, i_3} \neq 0$ ’ must be satisfied in equation(A.9). Factorization of equation(A.9) yields:

$$c_i^2(x, t) = \sum_{i_1=1}^i \left[R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \left\{ G_1^2 + h(G_1^2) G_2^2 \right\} \right]$$

; $\forall i = 1, 2, \dots, n$

where ;

$$G_1^2 = \frac{e^{\frac{xv}{2D_x}} \ell^{-1} \left\langle \frac{\left\{ e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x} (k_{i_2} + sR_{i_2}) \right\} \right]} \right\} - e^{-\mu_{i_1} x}}{(s + a_{i_1, -i_2})(s + a_{i_2, i_3})} \right\rangle}{\sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (a_{i_2, i_4} - a_{i_2, i_3})}$$

$$G_2^2 = \frac{-e^{\frac{xv}{2D_x}} \ell^{-1} \left\langle \frac{\left\{ e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x} (k_{i_2} + sR_{i_2}) \right\} \right]} \right\} - e^{-\mu_{i_1} x}}{(s + a_{i_1, -i_2})} \right\rangle}{R_{i_2} \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3}}$$

(A.11)

Note that the condition ‘ $a_{i_2, i_4} \neq a_{i_2, i_3}$ ’ must be satisfied in equation(A.11). The term ‘ G_1^2 ’ in equation(A.11) can be further factorized and simplified as:

$$G_1^2 = \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i \frac{e^{\frac{xv}{2D_x} \ell^{-1}} \left(\frac{\left\{ e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\}} \right]}{s + a_{i_1, -i_2}} \right\} - e^{-\mu_{i_1} x}}{\left(s + a_{i_1, -i_2} \right)} - \frac{\left\{ e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\}} \right]}{s + a_{i_2, i_3}} \right\} - e^{-\mu_{i_1} x}}{\left(s + a_{i_2, i_3} \right)} \right)}{\left(a_{i_2, i_3} - a_{i_1, -i_2} \right) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} \left(a_{i_2, i_4} - a_{i_2, i_3} \right)} \right)} \quad (\text{A.12})$$

Note that the condition ‘ $a_{i_2, i_4} \neq a_{i_1, -i_2}$ ’ must be satisfied in equation(A.12). ‘ G_1^2 ’ can be further simplified as:

$$G_1^2 = \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i \frac{e^{\frac{xv}{2D_x}} \left[\ell^{-1} \left\langle \frac{e^{\left[\frac{x}{2D_x} \{v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})}\} \right]}}{(s + a_{i_1, -i_2})} \right\rangle - e^{-\mu_{i_1} x} \ell^{-1} \left\langle \frac{1}{(s + a_{i_1, -i_2})} \right\rangle \right.}{- \ell^{-1} \left\langle \frac{e^{\left[\frac{x}{2D_x} \{v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})}\} \right]}}{(s + a_{i_2, i_3})} \right\rangle + e^{-\mu_{i_1} x} \ell^{-1} \left\langle \frac{1}{(s + a_{i_2, i_3})} \right\rangle} \left(a_{i_2, i_3} - a_{i_1, -i_2} \right) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (a_{i_2, i_4} - a_{i_2, i_3}) \right) \quad (\text{A.13})$$

110

‘ G_2^2 ’ in equation(A.11) can be further simplified as:

$$G_2^2 = \frac{-e^{\frac{xv}{2D_x}} \left[\ell^{-1} \left\langle \frac{e^{\left[\frac{x}{2D_x} \{v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})}\} \right]}}{(s + a_{i_1, -i_2})} \right\rangle - e^{-\mu_{i_1} x} \ell^{-1} \left\langle \frac{1}{(s + a_{i_1, -i_2})} \right\rangle \right]}{R_{i_2} \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3}} \quad (\text{A.14})$$

From inverse Laplace transform tables we get: [8] (p494, eq3).

$$\ell^{-1} \left\langle \frac{1}{(s + a_{i_1, i_2})} \right\rangle = e^{-a_{i_1, i_2} t} \quad (\text{A.15})$$

Using equation(A.15) and Appendix B, inverse Laplace expressions for the terms ‘ G_1^2 ’ and ‘ G_2^2 ’ can be evaluated and the solution for ‘ $c_i^2(x, t)$ ’ is obtained as:

$$c_i^2(x, t) = \sum_{i_1=1}^i \left[R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \{ G_1^2 + h(G_1^2) G_2^2 \} \right] ; \forall i = 1, 2, \dots, n$$

≡ where ;

$$G_1^2 = \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i \frac{\left\langle F_{i_2, i_1, -i_2} [x, t] - e^{(-\mu_{i_1} x - a_{i_1, -i_2} t)} - F_{i_2, i_2, i_3} [x, t] + e^{(-\mu_{i_1} x - a_{i_2, i_3} t)} \right\rangle}{(a_{i_2, i_3} - a_{i_1, -i_2}) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (a_{i_2, i_4} - a_{i_2, i_3})}$$

$$G_2^2 = \frac{-\left\langle F_{i_2, i_1, -i_2} [x, t] - e^{(-\mu_{i_1} x - a_{i_1, -i_2} t)} \right\rangle}{R_{i_2} \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3}} \quad (\text{A.16})$$

where; the term ' F_{i_1, i_2, i_3} ' is given by equation (B.8) or (B.14) in Appendix B.

APPENDIX B

Evaluation of Inverse Laplace Expressions for the Dirichlet Boundary Condition

$$\chi_{i_1, i_2, i_3, i_4} = e^{\frac{xv}{2D_x}} \ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_0(s+\lambda_{i_4})} \right\} e^{\left[\frac{-x}{2D_x} \sqrt{v^2 + 4D_x(k_{i_1} + sR_{i_1})} \right]}}{(s + a_{i_2, i_3})} \right\rangle \quad (\text{B.1})$$

Equation(B.1) can be simplified as:

$$\chi_{i_1, i_2, i_4, i_3} = e^{\frac{xv}{2D_x}} (\beta_1 - \beta_2)$$

where ;

$$\beta_1 = \ell^{-1} \left\langle \frac{e^{\left[-x \sqrt{\frac{R_{i_1}}{D_x}} \sqrt{s - \left(\frac{-v^2}{4R_{i_1}D_x} - \frac{k_{i_1}}{R_{i_1}} \right)} \right]}}{\left\{ s - \left(\frac{-v^2}{4R_{i_1}D_x} - \frac{k_{i_1}}{R_{i_1}} \right) - \frac{v^2}{4R_{i_1}D_x} - \frac{k_{i_1}}{R_{i_1}} + a_{i_2, i_3} \right\}} \right\rangle \quad (\text{B.2})$$

$$\beta_2 = \ell^{-1} \left\langle \frac{e^{-t_0\lambda_{i_4}} e^{-t_0s} e^{\left[-x \sqrt{\frac{R_{i_1}}{D_x}} \sqrt{s - \left(\frac{-v^2}{4R_{i_1}D_x} - \frac{k_{i_1}}{R_{i_1}} \right)} \right]}}{\left\{ s - \left(\frac{-v^2}{4R_{i_1}D_x} - \frac{k_{i_1}}{R_{i_1}} \right) - \frac{v^2}{4R_{i_1}D_x} - \frac{k_{i_1}}{R_{i_1}} + a_{i_2, i_3} \right\}} \right\rangle$$

‘ β_1 ’ can be evaluated as follows: Invoking the ‘First Shifting Theorem’ (see

[28], p253) we can simplify ‘ β_1 ’ as:

$$\beta_1 = e^{\left[\frac{-v^2}{4R_1 D_x} \frac{k_{i_1}}{R_1} \right] t} \ell^{-1} \left\langle \frac{e^{\left[-x \sqrt{\frac{R_1}{D_x}} \sqrt{s} \right]}}{\left\{ s - \left(\frac{v^2}{4R_1 D_x} + \frac{k_{i_1}}{R_1} - a_{i_2, i_3} \right) \right\}} \right\rangle \quad (\text{B.3})$$

The Laplace inverse of equation(B.3) can be readily obtained from the tables (see [8], p495, eq19). Applying this inversion, ‘ β_1 ’ can be evaluated as:

$$\beta_1 = \frac{e^{-a_{i_2, i_3} t}}{2} \left[e^{-x \sqrt{\frac{R_1}{D_x}} \sqrt{\left(\frac{v^2}{4R_1 D_x} + \frac{k_{i_1}}{R_1} - a_{i_2, i_3} \right)}} \operatorname{erfc} \left\{ \frac{x \sqrt{\frac{R_1}{D_x}} - \sqrt{\left(\frac{v^2}{4R_1 D_x} + \frac{k_{i_1}}{R_1} - a_{i_2, i_3} \right)} t}{2 \sqrt{\frac{R_1}{D_x} t}} \right\} + e^{x \sqrt{\frac{R_1}{D_x}} \sqrt{\left(\frac{v^2}{4R_1 D_x} + \frac{k_{i_1}}{R_1} - a_{i_2, i_3} \right)}} \operatorname{erfc} \left\{ \frac{x \sqrt{\frac{R_1}{D_x}} + \sqrt{\left(\frac{v^2}{4R_1 D_x} + \frac{k_{i_1}}{R_1} - a_{i_2, i_3} \right)} t}{2 \sqrt{\frac{R_1}{D_x} t}} \right\} \right] \quad (\text{B.4})$$

‘ β_1 ’ can be further rearranged and simplified as:

$$\beta_1 = \frac{e^{-a_{i_2, i_3} t}}{2} \left[e^{\frac{-x}{2D_x} \sqrt{v^2 + 4R_1 D_x \left(\frac{k_{i_1}}{R_1} - a_{i_2, i_3} \right)}} \operatorname{erfc} \left\{ \frac{R_1 x - \sqrt{v^2 + 4R_1 D_x \left(\frac{k_{i_1}}{R_1} - a_{i_2, i_3} \right)} t}{2 \sqrt{R_1 D_x t}} \right\} + e^{\frac{x}{2D_x} \sqrt{v^2 + 4R_1 D_x \left(\frac{k_{i_1}}{R_1} - a_{i_2, i_3} \right)}} \operatorname{erfc} \left\{ \frac{R_1 x + \sqrt{v^2 + 4R_1 D_x \left(\frac{k_{i_1}}{R_1} - a_{i_2, i_3} \right)} t}{2 \sqrt{R_1 D_x t}} \right\} \right] \quad (\text{B.5})$$

‘ β_2 ’ is evaluated as follows: ‘ β_2 ’ can be simplified as;

$$\beta_2 = e^{-t_0 \lambda_{i_4}} \ell^{-1} \left\langle e^{-t_0 s} \ell(\beta_1) \right\rangle \quad (\text{B.6})$$

Now, we invoke the ‘Second Shifting Theorem’ (see [28], p265) and evaluate

‘ β_2 ’ as: [Note: We have already evaluated ‘ β_1 ’; equation(B.4)]

$$\beta_2 = e^{-\lambda_{i_4} t_0} u(t-t_0) e^{-a_{i_2, i_3} (t-t_0)} \left[\begin{array}{l} e^{\frac{-x}{2D_x} \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)}} \operatorname{erfc} \left\{ \frac{R_{i_1} x - \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} (t-t_0)}{2\sqrt{R_{i_1} D_x (t-t_0)}} \right\} \\ + e^{\frac{x}{2D_x} \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)}} \operatorname{erfc} \left\{ \frac{R_{i_1} x + \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} (t-t_0)}{2\sqrt{R_{i_1} D_x (t-t_0)}} \right\} \end{array} \right] \quad (\text{B.7})$$

The final solution can be compactly expressed as:

$$\chi_{i_1, i_2, i_3, i_4} [x, t] = F_{i_1, i_2, i_3} [x, t] - u(t-t_0) e^{-\lambda_{i_4} t_0} F_{i_1, i_2, i_3} [x, (t-t_0)]$$

where ;

$$F_{i_1, i_2, i_3} [x, t] = \frac{e^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}}}{2} \left[\begin{array}{l} e^{\frac{-x\omega_{i_1, i_2, i_3}}{2D_x}} \operatorname{erfc} \left\{ \frac{R_{i_1} x - \omega_{i_1, i_2, i_3} t}{2\sqrt{R_{i_1} D_x t}} \right\} \\ + e^{\frac{x\omega_{i_1, i_2, i_3}}{2D_x}} \operatorname{erfc} \left\{ \frac{R_{i_1} x + \omega_{i_1, i_2, i_3} t}{2\sqrt{R_{i_1} D_x t}} \right\} \end{array} \right] \quad (\text{B.8})$$

$$\omega_{i_1, i_2, i_3} = \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)}$$

The above solution is valid only for real values of ‘ ω_{i_1, i_2, i_3} ’. For problems involving complex values for ‘ ω_{i_1, i_2, i_3} ’ the ‘ F_{i_1, i_2, i_3} ’ terms are evaluated as follows:

$$\text{Let ; } \omega_{i_1, i_2, i_3} = \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} = i\omega_{i_1, i_2, i_3}^* \quad (\text{B.9})$$

$$\text{where ; } \omega_{i_1, i_2, i_3}^* = \sqrt{\left| v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right) \right|}$$

Now the ‘ F_{i_1, i_2, i_3} ’ terms can be written as:

$$F_{i_1, i_2, i_3} [x, t] = \frac{e^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}}}{2} \left[e^{\frac{-xi\omega_{i_1, i_2, i_3}^*}{2D_x}} \operatorname{erfc} \left\{ \frac{R_{i_1} x - i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\} + e^{\frac{xi\omega_{i_1, i_2, i_3}^*}{2D_x}} \operatorname{erfc} \left\{ \frac{R_{i_1} x + i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\} \right] \quad (\text{B.10})$$

From the symmetric relations we get:

$$\text{if ; } \operatorname{erfc}\{a + ib\} = A + iB \text{ ; then ; } \operatorname{erfc}\{a - ib\} = A - iB \quad (\text{B.11})$$

where ; $a, b, A, B \in R$

Also the exponent of a complex number can be expressed as:

$$e^{-i\theta} = \operatorname{Cos}(\theta) - i \operatorname{Sin}(\theta)$$

$$e^{i\theta} = \operatorname{Cos}(\theta) + i \operatorname{Sin}(\theta) \quad (\text{B.12})$$

Using equations(B.11) and (B.12) equation(B.10) can be simplified as:

$$F_{i_1, i_2, i_3} [x, t] = \frac{e^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}}}{2} \left[\begin{array}{l} \left\{ \text{Cos} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) - i \text{Sin} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) \right\} (A - iB) + \\ \left\{ \text{Cos} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) + i \text{Sin} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) \right\} (A + iB) \end{array} \right]$$

where ;

$$(A - iB) = \text{erfc} \left\{ \frac{R_{i_1} x - i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\} ; \text{ and } (A + iB) = \text{erfc} \left\{ \frac{R_{i_1} x + i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\} \quad (\text{B.13})$$

Further simplification of equation(B.13) yields:

$$F_{i_1, i_2, i_3} [x, t] = e^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}} \left[A \text{Cos} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) - B \text{Sin} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) \right]$$

$$\text{where ; } \omega_{i_1, i_2, i_3}^* = \sqrt{\left| v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right) \right|} \quad (\text{B.14})$$

$$\text{and } (A + iB) = \text{erfc} \left\{ \frac{R_{i_1} x + i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\}$$

Hence in problems where ‘ ω_{i_1, i_2, i_3}^* ’ is complex the ‘ F_{i_1, i_2, i_3} ’ term is given as by equation(B.14).

APPENDIX C

Derivation of the General Solution for the Cauchy Boundary Condition

Since the governing equations, the initial condition and the boundary condition at ‘ ∞ ’ are identical for the Dirichlet and the Cauchy boundaries, the procedures involved in uncoupling the system of equations will be identical and hence we can use the analysis in ‘Section 2.2’ to obtain the semi-determined general solution and the linear transform matrices ‘ $[A]$ ’ and ‘ $[A]^{-1}$ ’. The semi-determined general solution [identical to equation(33) is given by:

$$b_i(x, s) = \Psi_i^2 e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right]} - \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o e^{-\mu_{i_1} x} \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_{i_1} - sR_{i_1} \right) \prod_{i_2=i_1, (i_2 \neq i)}^n -\left(k_{i_1} + sR_{i_1} - k_{i_2} - sR_{i_2} \right)} \right] \quad (C.1)$$

; $\forall i = 1, 2, \dots, n$

The linear transform matrices ‘ $[A]$ ’ and ‘ $[A]^{-1}$ ’ are given by equations(15) and (16) respectively. The constant ‘ Ψ_i^2 ’ in equation(C.1) is evaluated by using the boundary condition given by equation(47) after transforming them into the ‘ b ’ domain; this is done as follows. Laplace transform of equation(47) gives:

$$-D_x \frac{\partial \{p(0,s)\}}{\partial x} + v \{p(0,s)\} = \{\xi\} \quad (C.2)$$

where ; $\xi_i = \sum_{i_1=1}^i \frac{B_{i_1}^{i_1} v \{1 - e^{-t_0(s+\lambda_{i_1})}\}}{(s + \lambda_{i_1})}$; $\forall i = 1, 2, \dots, n$

In order to transform the boundary condition from the ‘ p ’ domain to the ‘ b ’ domain, we apply the linear transform given by equation(12).

$$-D_x \frac{\partial \{b(0,s)\}}{\partial x} + v \{b(0,s)\} = [A]^{-1} \{\xi\} \quad (C.3)$$

The explicit expression for ‘ $[A]^{-1} \{\xi\}_i$ ’ in equation(C.3) is given as:

$$\langle [A]^{-1} \{\xi\} \rangle_i = \sum_{i_1=1}^i \left[\frac{v \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_{i_2} + sR_{i_2} - k_{i_2} - sR_{i_2})} \sum_{i_2=1}^{i_1} \frac{B_{i_2}^{i_2} \{1 - e^{-t_0(s+\lambda_{i_2})}\}}{(s + \lambda_{i_2})} \right] \quad (C.4)$$

; $\forall i = 1, 2, \dots, n$

From equations(C.3) and (C.4) we obtain the boundary condition in the ‘ b ’ domain as:

$$-D_x \frac{\partial b_i(0,s)}{\partial x} + v b_i(0,s) = \sum_{i_1=1}^i \left[\frac{v \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_{i_2} + sR_{i_2} - k_{i_2} - sR_{i_2})} \sum_{i_2=1}^{i_1} \frac{B_{i_2}^{i_2} \{1 - e^{-t_0(s+\lambda_{i_2})}\}}{(s + \lambda_{i_2})} \right] \quad (C.5)$$

; $\forall i = 1, 2, \dots, n$

Substituting the expression for ‘ $b_i(x,s)$ ’ from the semi-determined general solution given by equation(C.1) into the transformed boundary condition given by equation(C.5), we can evaluate the constant ‘ Ψ_i^2 ’, as follows:

$$\begin{aligned}
& -D_x \Psi_i^2 \frac{1}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \\
& + D_x \sum_{i_1=1}^i \left[\frac{-\mu_{i_1} R_{i_1} c_{i_1}^o \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_i - sR_i \right) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right] \\
& + v \Psi_i^2 - v \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_i - sR_i \right) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right] \\
& = \sum_{i_1=1}^i \left[\frac{v \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_0(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})} \right] \\
& \quad ; \forall i = 1, 2, \dots, n
\end{aligned} \tag{C.6}$$

From equation(C.6) ‘ Ψ_i^2 ’, is evaluated as:

$$\begin{aligned}
\Psi_i^2 = & \left\{ \sum_{i_1=1}^i \left[\frac{v \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_0(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})} \right] \right. \\
& \left. + \sum_{i_1=1}^i \left[\frac{(v + \mu_{i_1} D_x) R_{i_1} c_{i_1}^o \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_i - sR_i \right) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right] \right\} \\
& \left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x (k_i + sR_i)} \right\} \\
& \quad ; \forall i = 1, 2, \dots, n
\end{aligned} \tag{C.7}$$

Therefore, the solution in the ‘ b ’ domain is:

$$\begin{aligned}
b_i(x,s) = & \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_o(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})} \right] \frac{ve^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_i + sR_i)} \right\}} \\
& + \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1} \left\{ \frac{\left(1 + \frac{\mu_{i_1} D_x}{v} \right) ve^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_i + sR_i)} \right\}} - e^{-\mu_{i_1} x} \right\}}{\left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_i - sR_i \right) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right] \\
& ; \forall i = 1, 2, \dots, n
\end{aligned} \tag{C.8}$$

121

Inverse linear transform of equation(C.8) is done to obtain the solution in the Laplace domain (‘ p ’ domain) by using equation(12). The solution given by equation(C.8) can be split into two parts and represented as:

$$b_i(x, s) = b_i^1(x, s) + b_i^2(x, s)$$

where ;

$$b_i^1(x, s) = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_o(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})} \right] \frac{ve^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_i + sR_i)} \right\}}$$

$$b_i^2(x, s) = \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1} \left\{ \frac{\left(1 + \frac{\mu_{i_1} D_x}{v} \right) ve^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4(k_i + sR_i)}{D_x}} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_i + sR_i)} \right\}} - e^{-\mu_{i_1} x} \right\}}{\left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_i - sR_i \right) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right]$$

$$; \forall i = 1, 2, \dots, n$$

(C.9)

Using the distributive property of matrix addition, we can apply the inverse linear transform to each of the individual terms and then sum them to get the solution in the ‘ p ’ domain. This is expressed as:

$$\{p\} = [A]\{b\} = [A]\{b^1\} + [A]\{b^2\} \quad (\text{C.10})$$

The first term ‘ $[A]\{b^1\}$ ’, can be evaluated as:

$$\{p^1\} = [A]\{b^1\} \quad (\text{C.11})$$

The explicit expression for ‘ $p_i^1(x, s)$ ’, in equation(C.11) is given as:

$$p_i^1(x, s) = \sum_{i_1=1}^i \left[\frac{\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=1}^{i_1} \frac{B_{i_2}^{i_2} \left\{ 1 - e^{-t_0(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})}}{ve^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]}} \sum_{i_2=i_1}^i \frac{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\}}{\prod_{i_3=i_1, (i_3 \neq i_2)}^i - (k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})} \right] ; \forall i = 1, 2, \dots, n \quad (\text{C.12})$$

Using a similar approach the second term ‘ $[A]\{b^2\}$ ’, is evaluated and the explicit expression for ‘ $p_i^2(x, s)$ ’ is:

$$p_i^2(x, s) = \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right)}{\sum_{i_2=i_1}^i \left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_{i_2} - sR_{i_2} \right) \prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})} \right. \\
\left. \left\{ \frac{\left(1 + \frac{\mu_{i_1} D_x}{v} \right) v e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\}} - e^{-\mu_{i_1} x} \right\} \right] \quad (C.13)$$

; $\forall i = 1, 2, \dots, n$

Substituting equations(C.12) and (C.13) into equation(C.10) we get the solution in the Laplace domain as:

$$\begin{aligned}
p_i(x, s) &= p_i^1(x, s) + p_i^2(x, s) = \\
& \sum_{i_1=1}^i \left[\frac{\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_0(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})}}{v e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]}} \right. \\
& \left. \sum_{i_2=i_1}^i \left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \prod_{i_3=i_1, (i_3 \neq i_2)}^i - (k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3}) \right] \\
& + \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right)}{\sum_{i_2=i_1}^i \left\{ \frac{\left(1 + \frac{\mu_{i_1} D_x}{v} \right) v e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\}} - e^{-\mu_{i_1} x} \right\}} \right. \\
& \left. \sum_{i_2=i_1}^i \left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_{i_2} - sR_{i_2} \right) \prod_{i_3=i_1, (i_3 \neq i_2)}^i - (k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3}) \right] \tag{C.14} \\
& ; \forall i = 1, 2, \dots, n
\end{aligned}$$

The final solution is obtained by taking an inverse Laplace transform of the solution given by equation(C.14). Inverse Laplace transform is performed as follows:

$$\begin{aligned}
c_i(x, t) &= c_i^1(x, t) + c_i^2(x, t) \\
&= \ell^{-1} \left\langle p_i^1(x, s) + p_i^2(x, s) \right\rangle \\
&= \ell^{-1} \left\langle p_i^1(x, s) \right\rangle + \ell^{-1} \left\langle p_i^2(x, s) \right\rangle \\
& ; \forall i = 1, 2, \dots, n
\end{aligned} \tag{C.15}$$

In Appendix D, the terms ‘ $\ell^{-1} \left\langle p_i^1(x, s) \right\rangle$ ’, and ‘ $\ell^{-1} \left\langle p_i^2(x, s) \right\rangle$ ’, are evaluated.

Substituting equations (D.6) and (D.13) in equation(C.15) we obtain the final solution in the time domain as:

$$\begin{aligned}
c_i(x, t) = & \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \{ G_1^1 + h(G_1^1) G_2^1 \} \right] \\
& + \sum_{i_1=1}^i \left[R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \{ G_1^2 + h(G_1^2) G_2^2 \} \right] \quad (C.16) \\
& ; \forall i = 1, 2, \dots, n
\end{aligned}$$

where; the ‘ G ’ are defined in equations (D.6) and (D.13) in Appendix D.

APPENDIX D

Factorization of the Laplace Terms for the Cauchy Boundary Condition

The first term ‘ $\ell^{-1}\langle p_i^1(x,s)\rangle$ ’, can be evaluated as follows. From equations (C.14) and (C.15) we get:

$$\begin{aligned}
 c_i^1(x,t) &= \ell^{-1}\langle p_i^1(x,s)\rangle \\
 &= \ell^{-1} \left\langle \sum_{i_1=1}^i \left[\frac{\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_0(s+\lambda_{i_2})} \right\}}{(s+\lambda_{i_2})}}{v e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]}} \right.} \right. \\
 &\quad \left. \left. \sum_{i_2=i_1}^i \frac{1}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\}} \prod_{i_3=i_1, (i_3 \neq i_2)}^i - (k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})} \right] \right\rangle \\
 &\quad ; \forall i = 1, 2, \dots, n \\
 &\quad (D.1)
 \end{aligned}$$

Equation(D.1) can be rewritten as:

$$c_i^1(x,t) = \ell^{-1} \left\langle \left[\sum_{i_1=1}^i \frac{\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) B_{i_1}^{i_3} \left\{ 1 - e^{-t_o(s+\lambda_{i_3})} \right\} v e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} \right]}}{\sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} (s + \lambda_{i_3}) \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) \prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (s + a_{i_2, i_4})} \right] \right\rangle$$

; $\forall i = 1, 2, \dots, n$

where ; $k_{i_1, i_2} = k_{i_1} - k_{i_2}$, $R_{i_1, i_2} = R_{i_1} - R_{i_2}$ and $a_{i_1, i_2} = \frac{k_{i_1, i_2}}{R_{i_1, i_2}}$

(D.2)

Note that in equation(D.2) the condition ' $k_{i_2, i^*} \neq 0$ ' must be satisfied for the solution to be determinate. Factorization of equation(D.2) gives:

$$c_i^1(x,t) = \ell^{-1} \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_1=1}^i \left\{ \sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \left[\frac{B_{i_1}^{i_3} \left\{ 1 - e^{-t_0(s+\lambda_{i_3})} \right\} v e^{\left[\frac{x}{2D_x} \left(v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right) \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} (s + \lambda_{i_3}) \prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4}=R_{i_2})}^i -k_{i_2, i_4}} \right.} \right. \right. \\ \left. \left. \left. \frac{1}{\sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (s + a_{i_2, i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2, i_5} (a_{i_2, i_5} - a_{i_2, i_4})} \right]} \right\} \right] \right] \quad (D.3)$$

$; \forall i = 1, 2, \dots, n$

It must be noted that the solution formulation as given by equation(D.3) is valid only when the condition ‘ $a_{i_2, i_5} \neq a_{i_2, i_4}$ ’, is satisfied. Equation(D.3) can be further factorized and simplified as:

$$c_i^1(x, t) = \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \left\{ G_1^1 + h(G_1^1) G_2^1 \right\} \right]$$

; $\forall i = 1, 2, \dots, n$

where ;

$$G_1^1 = \frac{\sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i B_{i_1}^{i_3} e^{\frac{xv}{2D_x}} \ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_0(s+\lambda_{i_3})} \right\} v e^{\left[\frac{-x}{2D_x} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} (s + \lambda_{i_3})(s + a_{i_2, i_4})} \right\rangle}{\left(\prod_{i_5=i_1, (i_5 \neq i_2, R_{i_5} = R_{i_2})}^i -k_{i_2, i_5} \right) (-R_{i_2, i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2, i_5} (a_{i_2, i_5} - a_{i_2, i_4})}$$

$$G_2^1 = \frac{B_{i_1}^{i_3} e^{\frac{xv}{2D_x}} \ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_0(s+\lambda_{i_3})} \right\} v e^{\left[\frac{-x}{2D_x} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} (s + \lambda_{i_3})} \right\rangle}{\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4}} \quad (D.4)$$

$$h(M) = \begin{cases} 1 & \text{if } M \text{ loop is not executed} \\ 0 & \text{if } M \text{ loop is executed} \end{cases}$$

The term ‘ G_1^1 ’ in equation(D.4) can be further factorized and simplified as:

$$G_1^1 = \sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i B_{i_1}^{i_3} e^{\frac{xv}{2D_x}} \ell^{-1} \left(\frac{\left\{ 1 - e^{-t_o(s+\lambda_{i_3})} \right\} v e^{\left[\frac{-x}{2D_x} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} (s + \lambda_{i_3})} - \frac{\left\{ 1 - e^{-t_o(s+\lambda_{i_3})} \right\} e^{\left[\frac{-x}{2D_x} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})} \right\} (s + a_{i_2, i_4})} \right) \left(a_{i_2, i_4} - \lambda_{i_3} \right) \left(\prod_{i_5=i_1, (i_5 \neq i_2, R_{i_5} = R_{i_2})}^i -k_{i_2, i_5} \right) (-R_{i_2, i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2, i_5} (a_{i_2, i_5} - a_{i_2, i_4}) \quad (D.5)$$

Again it must be noted that equation(D.5) is valid only when the condition ‘ $a_{i_2, i_4} \neq a_{i_3, 0}$ ’, where ‘ $a_{i_3, 0} = \lambda_{i_3}$ ’ is satisfied. Using the results from Appendix E, inverse Laplace expressions for the terms ‘ G_1^1 ’ and ‘ G_2^1 ’ can be evaluated and the solution for ‘ $c_i^1(x, t)$ ’ is obtained as:

$$c_i^1(x, t) = \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \left\{ G_1^1 + h(G_1^1, G_2^1) \right\} \right]$$

; $\forall i = 1, 2, \dots, n$

$$G_1^1 = \sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i \frac{B_{i_1}^{i_3} \left\langle \begin{array}{l} F_{i_2, i_3, 0}[x, t] - u(t-t_o) e^{(-\lambda_{i_3} t_o)} F_{i_2, i_3, 0}[x, (t-t_o)] \\ -F_{i_2, i_2, i_4}[x, t] + u(t-t_o) e^{(-\lambda_{i_3} t_o)} F_{i_2, i_2, i_4}[x, (t-t_o)] \end{array} \right\rangle}{(a_{i_2, i_4} - \lambda_{i_3}) \left(\prod_{i_5=i_1, (i_5 \neq i_2, R_{i_5} = R_{i_2})}^i -k_{i_2, i_5} \right) (-R_{i_2, i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2, i_5} (a_{i_2, i_5} - a_{i_2, i_4})}$$

$$G_2^1 = \frac{B_{i_1}^{i_3} \left\langle F_{i_2, i_3, 0}[x, t] - u(t-t_o) e^{(-\lambda_{i_3} t_o)} F_{i_2, i_3, 0}[x, (t-t_o)] \right\rangle}{\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4}}$$

(D.6)

where; the term ' F_{i_1, i_2, i_3} ' is given by equation (E.20) or (E.26) Appendix E. The second term ' $\ell^{-1} \langle p_i^2(x, s) \rangle$ ', is

evaluated as follows. From equations (C.14) and (C.15) we get:

$$c_i^2(x, t) = \ell^{-1} \langle p_i^2(x, s) \rangle$$

$$= \ell^{-1} \left\langle \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right)}{\sum_{i_2=i_1}^i \frac{\left(1 + \frac{\mu_{i_1} D_x}{v} \right) v e^{\left[\frac{x}{2D_x} \{ v - \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\}}} - e^{-\mu_{i_1} x}} \right]}{\sum_{i_2=i_1}^i \left(\mu_{i_1}^2 D_x + \mu_{i_1} v - k_{i_2} - sR_{i_2} \right) \prod_{i_3=i_1, (i_3 \neq i_2)}^i - (k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})} \right] \right\rangle \quad (D.7)$$

$; \forall i = 1, 2, \dots, n$

Equation(D.7) can be written as:

$$c_i^2(x, t) = \ell^{-1} \left[\sum_{i_1=1}^i \frac{R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left\{ \frac{\left(1 + \frac{\mu_{i_1} D_x}{v} \right) v e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\}} - e^{-\mu_{i_1} x} \right\}}{\sum_{i_2=i_1}^i \frac{-\left(s + a_{i_1, -i_2} \right) R_{i_2} \left(\prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3} \right) \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i -R_{i_2, i_3} \left(s + a_{i_2, i_3} \right)}} \right] \quad (D.8)$$

; $\forall i = 1, 2, \dots, n$

Note that the condition ‘ $k_{i_2, i_3} \neq 0$ ’ must be satisfied in equation(D.8). Factorization of equation(D.8) yields:

$$c_i^2(x,t) = \sum_{i_1=1}^i \left[R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \left\{ G_1^2 + h(G_1^2) G_2^2 \right\} \right]$$

$; \forall i=1, 2, \dots, n$

where ;

$$G_1^2 = \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (a_{i_2, i_4} - a_{i_2, i_3})$$

$$e^{\frac{xv}{2D_x} \ell^{-1}} \left\{ \frac{\left(1 + \frac{\mu_{i_1} D_x}{v} \right) v e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\}} - e^{-\mu_{i_1} x}} \right\}$$

$$\frac{1}{(s + a_{i_1, -i_2})(s + a_{i_2, i_3})}$$

$$G_2^2 = R_{i_2} \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3}$$

$$-e^{\frac{xv}{2D_x} \ell^{-1}} \left\{ \frac{\left(1 + \frac{\mu_{i_1} D_x}{v} \right) v e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\}} - e^{-\mu_{i_1} x}} \right\}$$

$$\frac{1}{(s + a_{i_1, -i_2})}$$

(D.9)

Note that the condition ‘ $a_{i_2, i_5} \neq a_{i_2, i_4}$ ’ must be satisfied in equation(D.9). The term ‘ G_1^2 ’ in equation(D.9) can be further factorized and simplified as:

$$G_1^2 = \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3}=R_{i_2})}^i \left(a_{i_2, i_3} - a_{i_1, -i_2} \right) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4}=R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \left(\prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} \right) (a_{i_2, i_4} - a_{i_2, i_3})$$

$$e^{\frac{xv}{2D_x} \ell^{-1}} \left\{ \frac{\left(1 + \frac{\mu_{i_1} D_x}{v} \right) v e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\}} - e^{-\mu_{i_1} x} \right\} \frac{1}{(s + a_{i_1, -i_2})}$$

$$\left\{ \frac{\left(1 + \frac{\mu_{i_1} D_x}{v} \right) v e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x (k_{i_2} + sR_{i_2})} \right\}} - e^{-\mu_{i_1} x} \right\} \frac{1}{(s + a_{i_2, i_3})}$$
(D.10)

Note that the condition ‘ $a_{i_2, i_4} \neq a_{i_1, -i_2}$ ’ must be satisfied in equation(D.10). ‘ G_1^2 ’ can be further simplified as:

$$G_1^2 = \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i \left[e^{\frac{xv}{2D_x} \ell^{-1}} \left[\ell^{-1} \left\langle \frac{\left(1 + \frac{\mu_{i_1} D_x}{v}\right) v e^{\left[\frac{x}{2D_x} \left\{v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})}\right\}\right]}}{\left(s + a_{i_1, -i_2}\right) \left\{\frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})}\right\}} \right\rangle - e^{-\mu_{i_1} x} \ell^{-1} \left\langle \frac{1}{\left(s + a_{i_1, -i_2}\right)} \right\rangle \right. \right. \\
\left. \left. - \ell^{-1} \left\langle \frac{\left(1 + \frac{\mu_{i_1} D_x}{v}\right) v e^{\left[\frac{x}{2D_x} \left\{v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})}\right\}\right]}}{\left(s + a_{i_2, i_3}\right) \left\{\frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})}\right\}} \right\rangle + e^{-\mu_{i_1} x} \ell^{-1} \left\langle \frac{1}{\left(s + a_{i_2, i_3}\right)} \right\rangle \right] \right. \\
\left. \left(a_{i_2, i_3} - a_{i_1, -i_2} \right) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} \left(a_{i_2, i_4} - a_{i_2, i_3} \right) \right) \quad (D.11)$$

' G_2^2 ' in equation(D.9) can be further simplified as:

$$G_2^2 = \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3}=R_{i_2})}^i R_{i_2} \left[-e^{\frac{xv}{2D_x}} \ell^{-1} \left\langle \frac{\left(1 + \frac{\mu_i D_x}{v}\right) v e^{\left[\frac{x}{2D_x} \left\{v - \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})}\right\}\right]}}{(s + a_{i_1, -i_2}) \left\{\frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_2} + sR_{i_2})}\right\}} \right\rangle - e^{-\mu_i x} \ell^{-1} \left\langle \frac{1}{(s + a_{i_1, -i_2})} \right\rangle \right] \quad (D.12)$$

Using equation(A.15) and Appendix E, inverse Laplace expressions for the terms ‘ G_1^2 ’ and ‘ G_2^2 ’ can be evaluated and the solution for ‘ $c_i^2(x, t)$ ’ is obtained as:

$$c_i^2(x, t) = \sum_{i_1=1}^i \left[R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \left\{ G_1^2 + h(G_1^2) G_2^2 \right\} \right]$$

; $\forall i = 1, 2, \dots, n$

where ;

$$G_1^2 = \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i \frac{\left\langle \left(1 + \frac{\mu_{i_1} D_x}{\nu} \right) F_{i_2, i_1, -i_2} [x, t] - e^{(-\mu_{i_1} x - a_{i_1, -i_2} t)} - \left(1 + \frac{\mu_{i_1} D_x}{\nu} \right) F_{i_2, i_2, i_3} [x, t] + e^{(-\mu_{i_1} x - a_{i_2, i_3} t)} \right\rangle}{(a_{i_2, i_3} - a_{i_1, -i_2}) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (a_{i_2, i_4} - a_{i_2, i_3})}$$

$$G_2^2 = \frac{-\left\langle \left(1 + \frac{\mu_{i_1} D_x}{\nu} \right) F_{i_2, i_1, -i_2} [x, t] - e^{(-\mu_{i_1} x - a_{i_1, -i_2} t)} \right\rangle}{R_{i_2} \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3}} \tag{D.13}$$

where; the term ‘ F_{i_1, i_2, i_3} ’ is given by equation (E.20) or (E.26) Appendix E.

APPENDIX E

Evaluation of Inverse Laplace Expressions for the Cauchy Boundary Condition

$$\chi_{i_1, i_2, i_3, i_4} = e^{\frac{xv}{2D_x}} \ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_0(s + \lambda_{i_4})} \right\} v e^{\left[\frac{-x}{2D_x} \sqrt{v^2 + 4D_x(k_{i_1} + sR_{i_1})} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x(k_{i_1} + R_{i_1}s)} \right\} (s + a_{i_2, i_3})} \right\rangle \quad (\text{E.1})$$

Equation(E.1) can be simplified as:

$$\chi_{i_1, i_2, i_3, i_4} = e^{\frac{xv}{2D_x}} (\beta_1 - \beta_2)$$

where ;

$$\beta_1 = \ell^{-1} \left\langle \frac{ve^{\left[-x\sqrt{\frac{R_i}{D_x}} \sqrt{s - \left(\frac{-v^2}{4R_i D_x} - \frac{k_i}{R_i} \right)} \right]}}{\sqrt{R_i D_x} \left\{ \frac{v}{2\sqrt{R_i D_x}} + \sqrt{s - \left(\frac{-v^2}{4R_i D_x} - \frac{k_i}{R_i} \right)} \right\} \left\{ s - \left(\frac{-v^2}{4R_i D_x} - \frac{k_i}{R_i} \right) - \frac{v^2}{4R_i D_x} - \frac{k_i}{R_i} + a_{i_2, i_3} \right\}} \right\rangle$$

$$\beta_2 = \ell^{-1} \left\langle \frac{e^{-t_0 \lambda_{i_4}} e^{-t_0 s} ve^{\left[-x\sqrt{\frac{R_i}{D_x}} \sqrt{s - \left(\frac{-v^2}{4R_i D_x} - \frac{k_i}{R_i} \right)} \right]}}{\sqrt{R_i D_x} \left\{ \frac{v}{2\sqrt{R_i D_x}} + \sqrt{s - \left(\frac{-v^2}{4R_i D_x} - \frac{k_i}{R_i} \right)} \right\} \left\{ s - \left(\frac{-v^2}{4R_i D_x} - \frac{k_i}{R_i} \right) - \frac{v^2}{4R_i D_x} - \frac{k_i}{R_i} + a_{i_2, i_3} \right\}} \right\rangle \quad (E.2)$$

‘ β_1 ’ can be evaluated as follows: Invoking the ‘First Shifting Theorem’ (see [28], p253) we can simplify ‘ β_1 ’ as:

$$\beta_1 = \frac{e^{\left[\frac{-v^2}{4R_{i_1}D_x} \frac{k_{i_1}}{R_{i_1}} \right] t}}{\sqrt{R_{i_1}D_x}} \ell^{-1} \left\langle \frac{ve^{\left[-x\sqrt{\frac{R_{i_1}}{D_x}}\sqrt{s} \right]}}{\left\{ \sqrt{s} + \frac{v}{2\sqrt{R_{i_1}D_x}} \right\} \left\{ s - \left(\frac{v^2}{4R_{i_1}D_x} + \frac{k_{i_1}}{R_{i_1}} - a_{i_2,i_3} \right) \right\}} \right\rangle \quad (\text{E.3})$$

The Laplace inverse of equation(E.3) can be readily obtained from the tables (see [8], p496, eq31). Applying this inversion, ‘ β_1 ’ can be evaluated as:

$$\beta_1 = \frac{ve^{-a_{i_2,i_3}t}}{2\sqrt{R_{i_1}D_x}} \left[\frac{e^{-x\sqrt{\frac{R_{i_1}}{D_x}}\sqrt{\left(\frac{v^2}{4R_{i_1}D_x} + \frac{k_{i_1}}{R_{i_1}} - a_{i_2,i_3}\right)}}}{\left\{\frac{v}{2\sqrt{R_{i_1}D_x}} + \sqrt{\frac{v^2}{4R_{i_1}D_x} + \frac{k_{i_1}}{R_{i_1}} - a_{i_2,i_3}}\right\}} \operatorname{erfc}\left\{\frac{x}{2}\sqrt{\frac{R_{i_1}}{D_x}t} - \sqrt{\left(\frac{v^2}{4R_{i_1}D_x} + \frac{k_{i_1}}{R_{i_1}} - a_{i_2,i_3}\right)t}\right\}} + \right. \\
\left. \frac{e^{x\sqrt{\frac{R_{i_1}}{D_x}}\sqrt{\left(\frac{v^2}{4R_{i_1}D_x} + \frac{k_{i_1}}{R_{i_1}} - a_{i_2,i_3}\right)}}}{\left\{\frac{v}{2\sqrt{R_{i_1}D_x}} - \sqrt{\frac{v^2}{4R_{i_1}D_x} + \frac{k_{i_1}}{R_{i_1}} - a_{i_2,i_3}}\right\}} \operatorname{erfc}\left\{\frac{x}{2}\sqrt{\frac{R_{i_1}}{D_x}t} + \sqrt{\left(\frac{v^2}{4R_{i_1}D_x} + \frac{k_{i_1}}{R_{i_1}} - a_{i_2,i_3}\right)t}\right\}} \right] \\
- \frac{ve^{\left[\frac{-v^2}{4R_{i_1}D_x} - \frac{k_{i_1}}{R_{i_1}}\right]t} \frac{v}{2\sqrt{R_{i_1}D_x}}}{\sqrt{R_{i_1}D_x} \left\{\frac{v^2}{4R_{i_1}D_x} - \left(\frac{v^2}{4R_{i_1}D_x} + \frac{k_{i_1}}{R_{i_1}} - a_{i_2,i_3}\right)\right\}} e^{\left[\frac{v}{2\sqrt{R_{i_1}D_x}}x\sqrt{\frac{R_{i_1}}{D_x}} + \frac{v^2t}{4R_{i_1}D_x}\right]} \operatorname{erfc}\left\{\frac{x}{2}\sqrt{\frac{R_{i_1}}{D_x}} + \frac{v\sqrt{t}}{2\sqrt{R_{i_1}D_x}}\right\} \quad (\text{E.4})$$

Note: the solution given by equation(E.5) is valid only when ' $\frac{k_{i_1}}{R_{i_1}} \neq a_{i_2,i_3}$ '. ' β_1 ' can be further rearranged and

simplified as:

$$\beta_1 = e^{-a_{i_2, i_3} t} \left[\frac{ve^{\frac{-x}{2D_x} \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)}}}{v + \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)}} \operatorname{erfc} \left\{ \frac{R_{i_1} x - \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} t}{2\sqrt{R_{i_1} D_x t}} \right\} + \frac{ve^{\frac{x}{2D_x} \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)}}}{v - \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)}} \operatorname{erfc} \left\{ \frac{R_{i_1} x + \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} t}{2\sqrt{R_{i_1} D_x t}} \right\} \right] - e^{\left[\frac{xv}{2D_x} \frac{k_{i_1} t}{R_{i_1}} \right]} \left\{ \frac{-v^2}{2R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} \right\} \operatorname{erfc} \left\{ \frac{R_{i_1} x + vt}{2\sqrt{R_{i_1} D_x t}} \right\} \quad (\text{E.5})$$

‘ β_2 ’ is evaluated as follows: ‘ β_2 ’ can be simplified as;

$$\beta_2 = e^{-\lambda_{i_4} t_0} \ell^{-1} \left\langle e^{-t_0 s} \ell(\beta_1) \right\rangle \quad (\text{E.6})$$

Now, we invoke the ‘Second Shifting Theorem’ (see [28], p265) and evaluate ‘ β_2 ’ as: [Note: We have already evaluated ‘ β_1 ’; equation(E.5)]

$$\beta_2 = e^{-\lambda_{i_4} t_0} u(t - t_0)$$

$$e^{-a_{i_2, i_3}(t-t_0)} \left[\begin{array}{l} \frac{ve^{\frac{-x}{2D_x} \sqrt{v^2 + 4R_{i_1} D_x} \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)}}{v + \sqrt{v^2 + 4R_{i_1} D_x} \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} \operatorname{erfc} \left\{ \frac{R_{i_1} x - \sqrt{v^2 + 4R_{i_1} D_x} \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right) (t - t_0)}{2\sqrt{R_{i_1} D_x} (t - t_0)} \right\} + \\ \frac{ve^{\frac{x}{2D_x} \sqrt{v^2 + 4R_{i_1} D_x} \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)}}{v - \sqrt{v^2 + 4R_{i_1} D_x} \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} \operatorname{erfc} \left\{ \frac{R_{i_1} x + \sqrt{v^2 + 4R_{i_1} D_x} \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right) (t - t_0)}{2\sqrt{R_{i_1} D_x} (t - t_0)} \right\} \end{array} \right] - e^{\left[\frac{xv}{2D_x} \frac{k_{i_1}(t-t_0)}{R_{i_1}} \right]} \left\{ \frac{-v^2}{2R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} \operatorname{erfc} \left\{ \frac{R_{i_1} x + v(t - t_0)}{2\sqrt{R_{i_1} D_x} (t - t_0)} \right\} \right. \quad (E.7)$$

Now, for the case when $\frac{k_{i_1}}{R_{i_1}} = a_{i_2, i_3}$, we evaluate ' β_1 ' as follows: From

equation(E.3) we get:

$$\beta_1 = \frac{e^{\left[\frac{-v^2}{4R_{i_1}D_x} \frac{k_{i_1}}{R_{i_1}} \right] t}}{\sqrt{R_{i_1}D_x}} \ell^{-1} \left\langle \frac{ve^{\left[-x\sqrt{\frac{R_{i_1}}{D_x}}\sqrt{s} \right]}}{\left\{ \sqrt{s} + \frac{v}{2\sqrt{R_{i_1}D_x}} \right\} \left\{ s - \frac{v^2}{4R_{i_1}D_x} \right\}} \right\rangle \quad (\text{E.8})$$

' β_1 ' can be factorized as:

$$\beta_1 = \frac{\sqrt{R_{i_1}D_x} e^{\left[\frac{-v^2}{4R_{i_1}D_x} \frac{k_{i_1}}{R_{i_1}} \right] t}}{v} \ell^{-1} \left\langle e^{\left[-x\sqrt{\frac{R_{i_1}}{D_x}}\sqrt{s} \right]} \left\{ \frac{1}{\left(\sqrt{s} - \frac{v}{2\sqrt{R_{i_1}D_x}} \right)} - \frac{\sqrt{s} + \frac{3v}{2\sqrt{R_{i_1}D_x}}}{\left(\sqrt{s} + \frac{v}{2\sqrt{R_{i_1}D_x}} \right)^2} \right\} \right\rangle \quad (\text{E.9})$$

' β_1 ' can be further simplified as:

$$\beta_1 = e^{\left[\frac{-v^2}{4R_{i_1}D_x} \frac{k_{i_1}}{R_{i_1}} \right] t} \ell^{-1} \left\langle \frac{e^{\left[-x\sqrt{\frac{R_{i_1}}{D_x}}\sqrt{s} \right]}}{\left(s - \frac{v^2}{4R_{i_1}D_x} \right)} - \frac{e^{\left[-x\sqrt{\frac{R_{i_1}}{D_x}}\sqrt{s} \right]}}{\left(\sqrt{s} + \frac{v}{2\sqrt{R_{i_1}D_x}} \right)^2} \right\rangle \quad (\text{E.10})$$

Equation(E.10) can be conveniently rewritten as:

$$\beta_1 = e^{\left[\frac{-v^2}{4R_i D_x} - \frac{k_i}{R_i} \right] t} (\gamma_1 - \gamma_2)$$

where ;

$$\gamma_1 = \ell^{-1} \left\langle \frac{e^{\left[-x \sqrt{\frac{R_i}{D_x}} \sqrt{s} \right]}}{\left(s - \frac{v^2}{4R_i D_x} \right)} \right\rangle \text{ and } \gamma_2 = \ell^{-1} \left\langle \frac{e^{\left[-x \sqrt{\frac{R_i}{D_x}} \sqrt{s} \right]}}{\left(\sqrt{s} + \frac{v}{2\sqrt{R_i D_x}} \right)^2} \right\rangle$$

Laplace inverse of ‘ γ_1 ’ can be obtained directly by using the tables (see [8], p495, eq19). Applying this inversion, ‘ γ_1 ’ can be evaluated as:

$$\gamma_1 = \frac{e^{\frac{v^2 t}{4R_i D_x}}}{2} \left[e^{-x \sqrt{\frac{R_i}{D_x}} \frac{v}{2\sqrt{R_i D_x}}} \operatorname{erfc} \left\{ \frac{x \sqrt{\frac{R_i}{D_x}}}{2\sqrt{t}} - \frac{v\sqrt{t}}{2\sqrt{R_i D_x}} \right\} + e^{x \sqrt{\frac{R_i}{D_x}} \frac{v}{2\sqrt{R_i D_x}}} \operatorname{erfc} \left\{ \frac{x \sqrt{\frac{R_i}{D_x}}}{2\sqrt{t}} + \frac{v\sqrt{t}}{2\sqrt{R_i D_x}} \right\} \right]$$

‘ γ_1 ’ can be further simplified as:

$$\gamma_1 = \frac{e^{\frac{v^2 t}{4R_i D_x}}}{2} \left[e^{\frac{-xv}{2D_x}} \operatorname{erfc} \left\{ \frac{R_i x - vt}{2\sqrt{D_x R_i t}} \right\} + e^{\frac{xv}{2D_x}} \operatorname{erfc} \left\{ \frac{R_i x + vt}{2\sqrt{D_x R_i t}} \right\} \right]$$

Laplace inverse of ‘ γ_2 ’ can also be obtained directly by using the tables (see [8], p495, eq17). Applying this inversion, ‘ γ_2 ’ can be evaluated as:

$$\begin{aligned}
\gamma_2 = & -2 \frac{v}{2\sqrt{R_i D_x}} \sqrt{\frac{t}{\pi}} e^{\frac{-x^2 R_i}{D_x 4t}} \\
& + \left(1 + \frac{v}{2\sqrt{R_i D_x}} x \sqrt{\frac{R_i}{D_x}} + 2 \frac{v^2 t}{4R_i D_x} \right) e^{\left[\frac{v}{2\sqrt{R_i D_x}} x \sqrt{\frac{R_i}{D_x}} + \frac{v^2 t}{4R_i D_x} \right]} \operatorname{erfc} \left\{ \frac{x \sqrt{\frac{R_i}{D_x}}}{2\sqrt{t}} + \frac{v\sqrt{t}}{2\sqrt{R_i D_x}} \right\}
\end{aligned}
\tag{E.14}$$

‘ γ_2 ’ can be further simplified as:

$$\gamma_2 = -v \sqrt{\frac{t}{R_i D_x \pi}} e^{\frac{-R_i x^2}{4D_x t}} + \left(1 + \frac{xv}{2D_x} + \frac{v^2 t}{2R_i D_x} \right) e^{\left[\frac{xv}{2D_x} + \frac{v^2 t}{4R_i D_x} \right]} \operatorname{erfc} \left\{ \frac{R_i x + vt}{2\sqrt{D_x R_i t}} \right\}
\tag{E.15}$$

From equations(E.11), (E.13) and (E.15) we can evaluate ‘ β_1 ’ as:

$$\begin{aligned}
\beta_1 = & e^{\frac{-k_i t}{R_i}} \left[\frac{1}{2} e^{\frac{-xv}{2D_x}} \operatorname{erfc} \left\{ \frac{R_i x - vt}{2\sqrt{D_x R_i t}} \right\} + \frac{1}{2} e^{\frac{xv}{2D_x}} \operatorname{erfc} \left\{ \frac{R_i x - vt}{2\sqrt{D_x R_i t}} \right\} \right. \\
& \left. + v \sqrt{\frac{t}{R_i D_x \pi}} e^{\left[\frac{-R_i x^2}{4D_x t} - \frac{v^2 t}{4R_i D_x} \right]} - \left(1 + \frac{xv}{2D_x} + \frac{v^2 t}{2R_i D_x} \right) e^{\frac{xv}{2D_x}} \operatorname{erfc} \left\{ \frac{R_i x + vt}{2\sqrt{D_x R_i t}} \right\} \right]
\end{aligned}
\tag{E.16}$$

‘ β_1 ’ can be further simplified as:

$$\beta_1 = e^{\frac{-k_i t}{R_i}} e^{\frac{-xv}{2D_x}} \left[\frac{1}{2} \operatorname{erfc} \left\{ \frac{R_i x - vt}{2\sqrt{D_x R_i t}} \right\} + \sqrt{\frac{v^2 t}{R_i D_x \pi}} e^{\frac{-(R_i x - vt)^2}{4D_x R_i t}} \right. \\
\left. - \frac{1}{2} \left(1 + \frac{xv}{D_x} + \frac{v^2 t}{R_i D_x} \right) e^{\frac{xv}{D_x}} \operatorname{erfc} \left\{ \frac{R_i x + vt}{2\sqrt{D_x R_i t}} \right\} \right]
\tag{E.17}$$

In order to evaluate ‘ β_2 ’ for the case when ‘ $\frac{k_i}{R_i} = a_{i_2, i_3}$ ’, we use the same

method as before. From equation(E.2) ‘ β_2 ’ can be written as:

$$\beta_2 = e^{-\lambda_4 t_0} \ell^{-1} \left\langle e^{-t_0 s} \ell(\beta_1) \right\rangle \quad (\text{E.18})$$

Now, we invoke the ‘Second Shifting Theorem’ (see [28], p265) and evaluate ‘ β_2 ’ as: [Note: We have already evaluated ‘ β_1 ’; equation(E.16)]

$$\beta_2 = e^{-\lambda_4 t_0} u(t-t_0) e^{\frac{-k_i(t-t_0)}{R_i} - \frac{-xv}{2D_x}} \left[\frac{1}{2} \operatorname{erfc} \left\{ \frac{R_i x - v(t-t_0)}{2\sqrt{D_x R_i (t-t_0)}} \right\} + \sqrt{\frac{v^2(t-t_0)}{R_i D_x \pi}} e^{\frac{-(R_i x - v(t-t_0))^2}{4D_x R_i (t-t_0)}} \right. \\ \left. - \frac{1}{2} \left(1 + \frac{xv}{D_x} + \frac{v^2(t-t_0)}{R_i D_x} \right) e^{\frac{xv}{D_x}} \operatorname{erfc} \left\{ \frac{R_i x + v(t-t_0)}{2\sqrt{D_x R_i (t-t_0)}} \right\} \right] \quad (\text{E.19})$$

The final solution can be compactly expressed as:

$$\chi_{i_1, i_2, i_3, i_4} [x, t] = F_{i_1, i_2, i_3} [x, t] - u(t - t_o) e^{(-\lambda_{i_3} t_o)} F_{i_1, i_2, i_3} [x, (t - t_o)]$$

where ;

$$F_{i_1, i_2, i_3} [x, t] = ve^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}} \left[\frac{e^{-\frac{x\omega_{i_1, i_2, i_3}}{2D_x}}}{(v + \omega_{i_1, i_2, i_3})} \operatorname{erfc} \left\{ \frac{R_{i_1} x - \omega_{i_1, i_2, i_3} t}{2\sqrt{R_{i_1} D_x t}} \right\} + \frac{e^{\frac{x\omega_{i_1, i_2, i_3}}{2D_x}}}{(v - \omega_{i_1, i_2, i_3})} \operatorname{erfc} \left\{ \frac{R_{i_1} x + \omega_{i_1, i_2, i_3} t}{2\sqrt{R_{i_1} D_x t}} \right\} \right] + \frac{2v^2}{(\omega_{i_1, i_2, i_3}^2 - v^2)} e^{\left[\frac{xv}{D_x} - \frac{k_{i_1} t}{R_{i_1}} \right]} \operatorname{erfc} \left\{ \frac{R_{i_1} x + vt}{2\sqrt{R_{i_1} D_x t}} \right\} ; \text{ when } \frac{k_{i_1}}{R_{i_1}} \neq a_{i_2, i_3}, \text{ and}$$

$$= e^{-\frac{k_{i_1} t}{R_{i_1}}} \left[\frac{1}{2} \operatorname{erfc} \left(\frac{R_{i_1} x - vt}{2\sqrt{R_{i_1} D_x t}} \right) + \sqrt{\frac{v^2 t}{\pi R_{i_1} D_x}} e^{-\frac{(R_{i_1} x - vt)^2}{4R_{i_1} D_x t}} - \frac{1}{2} \left(1 + \frac{xv}{D_x} + \frac{v^2 t}{R_{i_1} D_x} \right) e^{\frac{xv}{D_x}} \operatorname{erfc} \left(\frac{R_{i_1} x + vt}{2\sqrt{R_{i_1} D_x t}} \right) \right] ; \text{ when } \frac{k_{i_1}}{R_{i_1}} = a_{i_2, i_3}$$

where ; $\omega_{i_1, i_2, i_3} = \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)}$

(E.20)

The above solution is valid only for real values of ‘ ω_{i_1, i_2, i_3} ’. For problems involving complex values for ‘ ω_{i_1, i_2, i_3} ’ in the case when ‘ $\frac{k_{i_1}}{R_{i_1}} \neq a_{i_2, i_3}$ ’, the ‘ F_{i_1, i_2, i_3} ’ terms are evaluated as follows: (Note: when ‘ $\frac{k_{i_1}}{R_{i_1}} = a_{i_2, i_3}$ ’, the ‘ F_{i_1, i_2, i_3} ’ terms are unchanged)

$$\text{Let ; } \omega_{i_1, i_2, i_3} = \sqrt{v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} = i\omega_{i_1, i_2, i_3}^* \quad (\text{E.21})$$

$$\text{where ; } \omega_{i_1, i_2, i_3}^* = \sqrt{\left| v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right) \right|}$$

Now the ‘ F_{i_1, i_2, i_3} ’ terms can be written as:

$$F_{i_1, i_2, i_3} [x, t] = ve^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}} \left[\frac{e^{\frac{-xi\omega_{i_1, i_2, i_3}^*}{2D_x}}}{(v + i\omega_{i_1, i_2, i_3}^*)} \operatorname{erfc} \left\{ \frac{R_{i_1} x - i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\} + \frac{e^{\frac{-xi\omega_{i_1, i_2, i_3}^*}{2D_x}}}{(v - i\omega_{i_1, i_2, i_3}^*)} \operatorname{erfc} \left\{ \frac{R_{i_1} x + i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\} \right] \quad (\text{E.22})$$

$$+ \frac{v^2}{2R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} e^{\left[\frac{xv}{D_x} \frac{k_{i_1}}{R_{i_1}} \right]} \operatorname{erfc} \left\{ \frac{R_{i_1} x + vt}{2\sqrt{R_{i_1} D_x t}} \right\}$$

Using the symmetric relations:

$$\text{if ; } \operatorname{erfc} \{a + ib\} = A + iB \text{ ; then ; } \operatorname{erfc} \{a - ib\} = A - iB \quad (\text{E.23})$$

where ; $a, b, A, B \in R$

Also the exponent of a complex number can be expressed as:

$$e^{-i\theta} = \operatorname{Cos}(\theta) - i \operatorname{Sin}(\theta)$$

$$e^{i\theta} = \operatorname{Cos}(\theta) + i \operatorname{Sin}(\theta) \quad (\text{E.24})$$

Using equations(E.23) and (E.24) equation(E.22) can be simplified as:

$$\begin{aligned}
F_{i_1, i_2, i_3} [x, t] = & \frac{v}{\left(v^2 + \omega_{i_1, i_2, i_3}^{*2}\right)} e^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}} \left[\begin{aligned} & \left(v - i\omega_{i_1, i_2, i_3}^*\right) \left\{ \text{Cos} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) - i \text{Sin} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) \right\} (A - iB) + \\ & \left(v + i\omega_{i_1, i_2, i_3}^*\right) \left\{ \text{Cos} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) + i \text{Sin} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) \right\} (A + iB) \end{aligned} \right] \\
& + \frac{v^2}{2R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} e^{\left[\frac{xv}{D_x} - \frac{k_{i_1} t}{R_{i_1}} \right]} \text{erfc} \left\{ \frac{R_{i_1} x + vt}{2\sqrt{R_{i_1} D_x t}} \right\} \tag{E.25}
\end{aligned}$$

where ;

$$(A - iB) = \text{erfc} \left\{ \frac{R_{i_1} x - i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\} ; \text{ and } (A + iB) = \text{erfc} \left\{ \frac{R_{i_1} x + i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\}$$

Further simplification of equation(E.25) yields:

$$\begin{aligned}
 F_{i_1, i_2, i_3} [x, t] = & \frac{2v}{\left(v^2 + \omega_{i_1, i_2, i_3}^*{}^2\right)} e^{-a_{i_2, i_3} t} e^{\frac{xv}{2D_x}} \left[\begin{aligned} & \left(Av - B\omega_{i_1, i_2, i_3}^*\right) \text{Cos} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) \\ & - \left(A\omega_{i_1, i_2, i_3}^* + Bv\right) \text{Sin} \left(\frac{x\omega_{i_1, i_2, i_3}^*}{2D_x} \right) \end{aligned} \right] \\
 & + \frac{v^2}{2R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right)} e^{\left[\frac{xv}{D_x} - \frac{k_{i_1} t}{R_{i_1}} \right]} \text{erfc} \left\{ \frac{R_{i_1} x + vt}{2\sqrt{R_{i_1} D_x t}} \right\}
 \end{aligned}$$

where ; $\omega_{i_1, i_2, i_3}^* = \sqrt{\left| v^2 + 4R_{i_1} D_x \left(\frac{k_{i_1}}{R_{i_1}} - a_{i_2, i_3} \right) \right|}$

and $(A + iB) = \text{erfc} \left\{ \frac{R_{i_1} x + i\omega_{i_1, i_2, i_3}^* t}{2\sqrt{R_{i_1} D_x t}} \right\}$ (E.26)

Hence in problems where ‘ ω_{i_1, i_2, i_3} ’ is complex; for the case when ‘ $\frac{k_{i_1}}{R_{i_1}} \neq a_{i_2, i_3}$ ’,

the ‘ F_{i_1, i_2, i_3} ’ term is given as by equation(E.26).

APPENDIX F

Derivation of the Steady State Solutions

Dirichlet Boundary

The system of governing transport equations can be written in a matrix format as [11, 38]:

$$v \frac{d\{c\}}{dx} - D_x \frac{d^2\{c\}}{dx^2} = [K]\{c\} \quad (\text{F.1})$$

where; ‘[]’ denotes a square matrix and ‘{ }’ denotes a column vector. The corresponding boundary conditions can be written as:

$$\left\{ \frac{dc(\infty)}{dx} \right\} = 0 \quad (\text{F.2})$$

$$\{c(0)\} = \{\omega\} \quad (\text{F.3})$$

where ; $\omega_i = \sum_{i=1}^i B_i^i$; $\forall i = 1, 2, \dots, n$

Now in order to uncouple the system of ordinary differential equations (ODEs) given by equation(F.1), we apply a linear transform procedure described by Clement [11]. i.e. we perform the following matrix operation.

$$\{c\} = [A]\{b\} \quad (\text{F.4})$$

where; ‘ $\{b\}$ ’ is the concentration in the transformed domain. Applying this

transformation equation(F.4) gets modified as:

$$\frac{d^2 [A]\{b\}}{dx^2} - \left(\frac{v}{D_x}\right) \frac{d[A]\{b\}}{dx} + \frac{1}{D_x} ([K])[A]\{b\} = 0 \quad (\text{F.5})$$

Pre-multiplying equation(F.5) with ‘ $[A]^{-1}$ ’, we get:

$$\frac{d^2 \{b\}}{dx^2} - \left(\frac{v}{D_x}\right) \frac{d\{b\}}{dx} + \frac{1}{D_x} [\tilde{K}]\{b\} = 0 \quad (\text{F.6})$$

where ; $[\tilde{K}] = [A]^{-1} [K] [A]$

By using the similarity transformation procedure described in Clement [11], we

can make ‘ $[\tilde{K}]$ ’, a diagonal matrix and thus uncouple the system of equations. The

corresponding ‘ $[A]$ ’, matrix is given as:

$$[A] = \begin{bmatrix} \prod_{i=2}^n \frac{-(k_1 - k_i)}{y_i k_{i-1}}, 0, 0, \dots, (n-1) \text{ times} \\ \prod_{i=3}^n \frac{-(k_1 - k_i)}{y_i k_{i-1}}, \prod_{i=3}^n \frac{-(k_2 - k_i)}{y_i k_{i-1}}, 0, 0, \dots, (n-2) \text{ times} \\ . \\ . \\ \prod_{i=n}^n \frac{-(k_{11} - k_i)}{y_i k_{i-1}}, \prod_{i=n}^n \frac{-(k_2 - k_i)}{y_i k_{i-1}}, \dots, \prod_{i=n}^n \frac{-(k_{n-1} - k_i)}{y_i k_{i-1}}, 0 \\ 1, 1, \dots, n \text{ times} \end{bmatrix} \quad (\text{F.7})$$

The ‘ $[A]^{-1}$ ’, matrix is:

$$[A]^{-1} = \left[\begin{array}{c} \frac{\prod_{i=2}^n y_i k_{i-1}}{\prod_{i=1, (i \neq 1)}^n -(k_1 - k_i)}, 0, 0, \dots, (n-1) \text{ times} \\ \frac{\prod_{i=2}^n y_i k_{i-1}}{\prod_{i=1, (i \neq 2)}^n -(k_2 - k_i)}, \frac{\prod_{i=3}^n y_i k_{i-1}}{\prod_{i=2, (i \neq 2)}^n -(k_2 - k_i)}, 0, 0, \dots, (n-2) \text{ times} \\ \cdot \\ \cdot \\ \frac{\prod_{i=2}^n y_i k_{i-1}}{\prod_{i=1, (i \neq n)}^n -(k_n - k_i)}, \frac{\prod_{i=3}^n y_i k_{i-1}}{\prod_{i=2, (i \neq n)}^n -(k_n - k_i)}, \dots, \frac{\prod_{i=n}^n y_i k_{i-1}}{\prod_{i=n-1, (i \neq n)}^n -(k_n - k_i)}, 1 \end{array} \right] \quad (\text{F.8})$$

The corresponding $\left[\tilde{K} \right]$ matrix is:

$$\left[\tilde{K} \right] = \left[\begin{array}{c} -k_1, 0, 0, \dots, (n-1) \text{ times} \\ 0, -k_2, 0, 0, \dots, (n-2) \text{ times} \\ \cdot \\ 0, 0, \dots, (i-1) \text{ times}, -k_i, 0, 0, \dots, (n-i) \text{ times} \\ \cdot \\ 0, 0, \dots, (n-1) \text{ times}, -k_n \end{array} \right] \quad (\text{F.9})$$

Equation(F.5) describes a set of ‘ n ’ independent second-order homogeneous ODEs the boundary conditions of which are obtained by linear transforming equations(F.2) and(F.3). This yields:

$$\left\{ \frac{db(\infty)}{dx} \right\} = 0 \quad (\text{F.10})$$

$$b_i(0) = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i - k_{i_2})} \sum_{i_2=1}^{i_1} B_{i_1}^{i_2} \right] \quad (\text{F.11})$$

; $\forall i = 1, 2, \dots, n$

Since equation(F.5) is uncoupled, it can now be written as a set of ‘ n ’ independent equations as:

$$\frac{d^2 b_i(x)}{dx^2} - \left(\frac{v}{D_x} \right) \frac{db_i(x)}{dx} + \frac{1}{D_x} (-k_i) b_i(x) = 0 \quad (\text{F.12})$$

; $\forall i = 1, 2, \dots, n$

The general solution to equation(F.12) is given as:

$$b_i(x) = \Psi_i^1 e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} + \sqrt{\frac{v^2}{D_x^2} + \frac{4k_i}{D_x}} \right\} \right]} + \Psi_i^2 e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4k_i}{D_x}} \right\} \right]} \quad (\text{F.13})$$

; $\forall i = 1, 2, \dots, n$

where; Ψ_i^1 and Ψ_i^2 are constants. In order to apply the boundary condition given by equation(F.10) we differentiate the general solution with respect to ‘ x ’.

Differentiation of equation(F.13) with respect to x yields:

$$\begin{aligned} \frac{db_i(x)}{dx} = & \Psi_i^1 \left[\frac{1}{2} \left\{ \frac{v}{D_x} + \sqrt{\frac{v^2}{D_x^2} + \frac{4k_i}{D_x}} \right\} \right] e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} + \sqrt{\frac{v^2}{D_x^2} + \frac{4k_i}{D_x}} \right\} \right]} \\ & + \Psi_i^2 \left[\frac{1}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4k_i}{D_x}} \right\} \right] e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4k_i}{D_x}} \right\} \right]} \end{aligned} \quad (\text{F.14})$$

; $\forall i = 1, 2, \dots, n$

To satisfy the boundary condition given by equation(F.10), i.e. when ‘ x ’ tends to ‘ ∞ ’; the exponential function in the first term tends to ‘ ∞ ’, hence Ψ_i^1 must vanish.

Equation(F.13) now reduces to:

$$b_i(x) = \Psi_i^2 e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4k_i}{D_x}} \right\} \right]} ; \forall i = 1, 2, \dots, n \quad (\text{F.15})$$

Applying the second boundary condition given by equation(F.11), we get:

$$\Psi_i^2 = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_{i_2} - k_{i_1})} \sum_{i_2=1}^{i_1} B_{i_1}^{i_2} \right] ; \forall i = 1, 2, \dots, n \quad (\text{F.16})$$

Note that the condition ‘ $k_{i_1, i_2} \neq 0$ ’ where $1 \leq i_1, i_2 \leq n$ and $i_1 \neq i_2$ must be satisfied for equation(F.16) to be valid. Therefore, the solution in the ‘ b ’ domain is:

$$b_i(x) = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_{i_2} - k_{i_1})} \sum_{i_2=1}^{i_1} B_{i_1}^{i_2} \right] e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4k_i}{D_x}} \right\} \right]} ; \forall i = 1, 2, \dots, n \quad (\text{F.17})$$

Inverse linear transform of equation(F.17) is done to obtain the solution in the ‘ c ’ domain by using equation(F.4). The final steady state solution for the Dirichlet boundary is:

$$c_i(x) = \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left(\sum_{i_2=1}^{i_1} B_{i_1}^{i_2} \right) \sum_{i_2=i_1}^i \frac{e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x k_{i_2}} \right\} \right]}}{\prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} - k_{i_3})} \right] ; \forall i = 1, 2, \dots, n \quad (\text{F.18})$$

Cauchy Boundary

For the case of the Cauchy boundary, the boundary condition at the source is described as follows:

$$-D_x \frac{dc_i(0)}{dx} + vc_i(0) = \sum_{i_1=1}^i B_{i_1}^i v ; \forall i = 1, 2, \dots, n \quad (\text{F.19})$$

Since the governing equations and the boundary condition at ‘ ∞ ’ are identical for the Dirichlet and the Cauchy boundaries, the procedures involved in uncoupling the system of equations will be identical and hence we can use the previous section to obtain the semi-determined general solution and the linear transform matrices ‘ $[A]$ ’ and ‘ $[A]^{-1}$ ’. The semi-determined general solution [identical to equation(F.15)] is given by:

$$b_i(x) = \Psi_i^2 e^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4k_i}{D_x}} \right\} \right]} ; \forall i = 1, 2, \dots, n \quad (\text{F.20})$$

The linear transform matrices ‘ $[A]$ ’ and ‘ $[A]^{-1}$ ’, are given by equations(F.7) and (F.8) respectively. The constant ‘ Ψ_i^2 ’ in equation(F.20) is evaluated by using the boundary condition given by equation(F.19) after transforming them into the ‘ b ’ domain by applying the linear transform given by equation(F.4).

$$-D_x \frac{db_i(0)}{dx} + vb_i(0) = \sum_{i_1=1}^i \left[\frac{v \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_{i_2} - k_{i_2})} \sum_{i_2=1}^{i_1} B_{i_2}^{i_1} \right] ; \forall i = 1, 2, \dots, n \quad (\text{F.21})$$

Note that the condition ‘ $k_{i_1, i_2} \neq 0$ ’ where $1 \leq i_1, i_2 \leq n$ and $i_1 \neq i_2$ must be satisfied for equation(F.21) to be valid. Substituting the expression for ‘ $b_i(x, s)$ ’ from the semi-determined general solution given by equation(F.20) into the transformed boundary condition given by equation(F.21), we can evaluate the constant ‘ Ψ_i^2 ’ as follows:

$$-D_x \Psi_i^2 \left[\frac{1}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4k_i}{D_x}} \right\} \right] + v \Psi_i^2 = \sum_{i_1=1}^i \left[\frac{v \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1} \sum_{i_2=1}^{i_1} B_{i_1}^{i_2}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i - k_{i_2})} \right] \quad (F.22)$$

; $\forall i = 1, 2, \dots, n$

From equation(F.22) ‘ Ψ_i^2 ’ is evaluated as:

$$\Psi_i^2 = \frac{\sum_{i_1=1}^i \left[\frac{v \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1} \sum_{i_2=1}^{i_1} B_{i_1}^{i_2}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i - k_{i_2})} \right]}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x k_i} \right\}} ; \forall i = 1, 2, \dots, n \quad (F.23)$$

Therefore, the solution in the ‘ b ’ domain is:

$$b_i(x) = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1} \sum_{i_2=1}^{i_1} B_{i_1}^{i_2}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i - k_{i_2})} \right] \frac{ve^{\left[\frac{x}{2} \left\{ \frac{v}{D_x} - \sqrt{\frac{v^2}{D_x^2} + \frac{4k_i}{D_x}} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x k_i} \right\}} \quad (F.24)$$

; $\forall i = 1, 2, \dots, n$

Inverse Linear transform of equation(F.24) is done to obtain the solution in the ‘ c ’ domain by using equation(F.4). The final steady state solution for the Dirichlet boundary is:

$$c_i(x) = \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left(\sum_{i_2=1}^{i_1} B_{i_1}^{i_2} \right) \sum_{i_2=i_1}^i \frac{v e^{\left[\frac{x}{2D_x} \left\{ v - \sqrt{v^2 + 4D_x k_{i_2}} \right\} \right]}}{\left\{ \frac{v}{2} + \frac{1}{2} \sqrt{v^2 + 4D_x k_{i_2}} \right\}} \prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} - k_{i_3}) \right]$$

; $\forall i = 1, 2, \dots, n$
(F.25)

APPENDIX G

Derivation of the Solution for the zero Dispersion Case

The system of governing transport equations can be written in a matrix format as [11, 38]:

$$[R] \frac{\partial \{c\}}{\partial t} + v \frac{\partial \{c\}}{\partial x} = [K] \{c\} \quad (\text{G.1})$$

where; ‘ $[\]$ ’ denotes a square matrix and ‘ $\{ \}$ ’ denotes a column vector. The corresponding initial and boundary conditions are:

$$\{c(x, 0)\} = \{c^0 e^{-\mu x}\}, \quad 0 < x < \infty \quad (\text{G.2})$$

$$\{c(0, t)\} = \{\omega\}, \quad t > 0 \quad (\text{G.3})$$

$$\text{where ; } \omega_i = \sum_{i=1}^i B_i^i e^{-\lambda_i t} \{u(t) - u(t - t_o)\}, \quad t > 0 ; \quad \forall i = 1, 2, \dots, n$$

The solution procedure used here is adopted from Quezada et al. [38]. Applying Laplace transform to equation(G.1), we get:

$$[R] s \{p\} - [R] \{c(x, 0)\} + v \frac{\partial \{p\}}{\partial x} = [K] \{p\} \quad (\text{G.4})$$

where; ‘ s ’ is the Laplace variable and ‘ p ’ is the Laplace transformed concentration.

Substituting equation(G.2) in (G.4) and rearranging we get:

$$-v \frac{\partial \{p\}}{\partial x} + ([K] - [R]s) \{p\} = -[R] \{c^o e^{-\mu x}\} \quad (G.5)$$

Now in order to uncouple the system of ordinary differential equations (ODEs) given by equation(G.5), we apply a linear transform procedure described by Clement [11]. i.e. we perform the following matrix operation.

$$\{p\} = [A] \{b\} \quad (G.6)$$

where; ‘ $\{b\}$ ’ is the concentration in the doubly transformed domain. Applying this transformation equation(G.5) gets modified as:

$$-v \frac{\partial [A] \{b\}}{\partial x} + ([K] - [R]s) [A] \{b\} = -[R] \{c^o e^{-\mu x}\} \quad (G.7)$$

Pre-multiplying equation(G.7) with ‘ $[A]^{-1}$ ’, we get:

$$-v \frac{\partial \{b\}}{\partial x} + [\tilde{K}] \{b\} = -\{\tilde{C}\} \quad (G.8)$$

where ; $[\tilde{K}] = [A]^{-1} ([K] - [R]s) [A]$ and $\{\tilde{C}\} = [A]^{-1} [R] \{c^o e^{-\mu x}\}$

By using the procedure described in Clement [11], we can make ‘ $[\tilde{K}]$ ’, a diagonal matrix and thus uncouple the system of equations. The corresponding ‘ $[A]$ ’ matrix, ‘ $[A]^{-1}$ ’, matrix, ‘ $[\tilde{K}]$ ’, matrix and ‘ $\{\tilde{C}\}$ ’, vector is given in ‘Section 2.2’. Equation(G.8) describes a set of ‘ n ’ independent first-order non-homogeneous ODEs the boundary conditions of which are obtained by a combination of Laplace and Linear transforms of equation(G.3). Laplace transform of equation(G.3) yields:

$$\{p(0, s)\} = \{\xi\}$$

$$\text{where ; } \xi_i = \sum_{i_1=1}^i \frac{B_i^{i_1} \left\{1 - e^{-t_0(s+\lambda_{i_1})}\right\}}{(s+\lambda_{i_1})} ; \forall i=1, 2, \dots, n \quad (\text{G.9})$$

In order to transform the boundary conditions from ‘ p ’ domain to the ‘ b ’ domain, we apply the Linear transform given by equation(G.6). This yields:

$$\{b(0, s)\} = [A]^{-1} \{\xi\} \quad (\text{G.10})$$

The explicit expression for ‘ $b_i(0, s)$ ’ in equation(G.10) is given as:

$$b_i(0, s) = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{1 - e^{-t_0(s+\lambda_{i_2})}\right\}}{(s+\lambda_{i_2})} \right] \quad (\text{G.11})$$

$$; \forall i=1, 2, \dots, n$$

Since equation(G.8) is uncoupled, it can now be written as a set of ‘ n ’ independent equations as:

$$-v \frac{\partial b_i(x, s)}{\partial x} + (-k_i - sR_i) b_i(x, s) = - \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o e^{-\mu_{i_1} x} \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right] \quad (\text{G.12})$$

$$; \forall i=1, 2, \dots, n$$

Equation(G.12) can be re-arranged and expressed as:

$$\frac{\partial b_i(x, s)}{\partial x} + \frac{(k_i + sR_i)}{v} b_i(x, s) = \frac{1}{v} \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o e^{-\mu_{i_1} x} \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right] \quad (\text{G.13})$$

$$; \forall i=1, 2, \dots, n$$

The general solution to equation(G.13) is given as:

$$b_i(x, s) = - \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o e^{-\mu_{i_1} x} \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\left(\mu_{i_1} \nu - k_{i_1} - sR_{i_1} \right) \prod_{i_2=i_1, (i_2 \neq i)}^n -\left(k_{i_1} + sR_{i_1} - k_{i_2} - sR_{i_2} \right)} \right] + \Psi_i e^{\frac{-(k_i + sR_i)x}{\nu}} \quad (G.14)$$

; $\forall i = 1, 2, \dots, n$

where; Ψ_i is a constant. Applying the boundary condition given by equation(G.11) we get:

$$\Psi_i = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -\left(k_{i_1} + sR_{i_1} - k_{i_2} - sR_{i_2} \right)} \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_o(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})} \right]$$

$$+ \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\left(\mu_{i_1} \nu - k_{i_1} - sR_{i_1} \right) \prod_{i_2=i_1, (i_2 \neq i)}^n -\left(k_{i_1} + sR_{i_1} - k_{i_2} - sR_{i_2} \right)} \right] \quad (G.15)$$

; $\forall i = 1, 2, \dots, n$

Therefore, the solution in the ‘ b ’ domain is:

$$b_i(x, s) = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -\left(k_{i_1} + sR_{i_1} - k_{i_2} - sR_{i_2} \right)} \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_o(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})} \right] e^{\frac{-(k_i + sR_i)x}{\nu}}$$

$$+ \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1} \left\{ e^{\frac{-(k_i + sR_i)x}{\nu}} - e^{-\mu_{i_1} x} \right\}}{\left(\mu_{i_1} \nu - k_{i_1} - sR_{i_1} \right) \prod_{i_2=i_1, (i_2 \neq i)}^n -\left(k_{i_1} + sR_{i_1} - k_{i_2} - sR_{i_2} \right)} \right] \quad (G.16)$$

; $\forall i = 1, 2, \dots, n$

Inverse Linear transform of equation(G.16) is done to obtain the solution in the Laplace domain (‘ p ’ domain) by using equation(G.6). The solution given by equation(G.16) can be split into two parts and represented as:

$$b_i(x, s) = b_i^1(x, s) + b_i^2(x, s)$$

where ;

$$b_i^1(x, s) = \sum_{i_1=1}^i \left[\frac{\prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1}}{\prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_o(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})} \right] e^{\frac{-(k_i+sR_i)x}{v}}$$

$$b_i^2(x, s) = \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \prod_{i_2=i_1+1}^n y_{i_2} k_{i_2-1} \left\{ e^{\frac{-(k_i+sR_i)x}{v}} - e^{-\mu_{i_1}x} \right\}}{(\mu_{i_1}v - k_i - sR_i) \prod_{i_2=i_1, (i_2 \neq i)}^n -(k_i + sR_i - k_{i_2} - sR_{i_2})} \right]$$

; $\forall i = 1, 2, \dots, n$

(G.17)

Using the distributive property, we can apply the inverse linear transform to each of the individual terms and then sum them to get the solution in the ‘ p ’ domain. This is expressed as:

$$\{p\} = [A]\{b\} = [A]\{b^1\} + [A]\{b^2\} \quad (G.18)$$

The first term ‘ $[A]\{b^1\}$ ’, can be evaluated as:

$$\{p^1\} = [A]\{b^1\} \quad (G.19)$$

The explicit expression for ‘ $p_i^1(x, s)$ ’ in equation(G.19) is given as:

$$p_i^1(x,s) = \sum_{i_1=1}^i \left[\frac{\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_0(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})}}{\sum_{i_2=i_1}^i \frac{e^{-\frac{(k_{i_2} + sR_{i_2})x}{v}}}{\prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})}} \right] \quad (G.20)$$

; $\forall i = 1, 2, \dots, n$

Using a similar approach the second term ‘ $[A]\{b^2\}$ ’ is evaluated and the explicit expression for ‘ $p_i^2(x,s)$ ’ is:

$$p_i^2(x,s) = \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left\{ e^{-\frac{(k_{i_2} + sR_{i_2})x}{v}} - e^{-\mu_{i_1} x} \right\}}{\sum_{i_2=i_1}^i \frac{1}{(\mu_{i_1} v - k_{i_2} - sR_{i_2}) \prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})}} \right] \quad (G.21)$$

; $\forall i = 1, 2, \dots, n$

Substituting equations(G.20) and (G.21) into equation(G.18) we get the solution in the Laplace domain as:

$$\begin{aligned}
p_i(x, s) &= p_i^1(x, s) + p_i^2(x, s) = \\
&\sum_{i_1=1}^i \left[\frac{\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_o(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})}}{e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}}} \right. \\
&\quad \left. \sum_{i_2=i_1}^i \frac{\prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})}{R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right)} \right. \\
&\quad \left. \sum_{i_2=i_1}^i \frac{\left\{ e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}} - e^{-\mu_{i_1}x} \right\}}{\left(\mu_{i_1} v - k_{i_2} - sR_{i_2} \right) \prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})} \right] \quad (G.22) \\
&\quad ; \forall i = 1, 2, \dots, n
\end{aligned}$$

The final solution is obtained by taking an inverse Laplace transform of the solution given by equation(G.22). Inverse Laplace transform is performed as follows:

$$\begin{aligned}
c_i(x, t) &= c_i^1(x, t) + c_i^2(x, t) \\
&= \ell^{-1} \left\langle p_i^1(x, s) + p_i^2(x, s) \right\rangle \\
&= \ell^{-1} \left\langle p_i^1(x, s) \right\rangle + \ell^{-1} \left\langle p_i^2(x, s) \right\rangle \\
&\quad ; \forall i = 1, 2, \dots, n
\end{aligned} \quad (G.23)$$

The first term ‘ $\ell^{-1} \left\langle p_i^1(x, s) \right\rangle$ ’, can be evaluated as follows. From equations (G.22) and (G.23) we get:

$$c_i^1(x,t) = \ell^{-1} \langle p_i^1(x,s) \rangle = \ell^{-1} \left(\sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=1}^{i_1} \frac{B_{i_1}^{i_2} \left\{ 1 - e^{-t_o(s+\lambda_{i_2})} \right\}}{(s + \lambda_{i_2})} \right. \right. \\ \left. \left. \frac{e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}}}{\prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})} \right] \right) \\ ; \forall i = 1, 2, \dots, n \\ \text{(G.24)}$$

Equation(G.24) can be rewritten as:

$$c_i^1(x,t) = \ell^{-1} \left\langle \sum_{i_1=1}^i \left[\frac{\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) B_{i_1}^{i_3} \left\{ 1 - e^{-t_o(s+\lambda_{i_3})} \right\} e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}}}{(s + a_{i_3,0}) \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) \prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (s + a_{i_2, i_4})} \right] \right\rangle$$

; $\forall i = 1, 2, \dots, n$ (G.25)

where ; $k_{i_1, i_2} = k_{i_1} - k_{i_2}$, $R_{i_1, i_2} = R_{i_1} - R_{i_2}$ and $a_{i_1, i_2} = \begin{cases} \frac{k_{i_1, i_2}}{R_{i_1, i_2}} & ; \text{ when } i_2 > 0 \\ \lambda_{i_1} & ; \text{ when } i_2 = 0 \end{cases}$

Note that in equation(G.25) the condition ' $k_{i_2, i_4} \neq 0$ ' must be satisfied for the solution to be determinate. Factorization of equation(G.25) gives:

$$c_i^1(x,t) = \ell^{-1} \left[\sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left\{ \frac{B_{i_1}^{i_3} \left\{ 1 - e^{-t_o(s+\lambda_{i_3})} \right\} e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}}}{(s + a_{i_3,0}) \prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4}=R_{i_2})}^i -k_{i_2, i_4}} \right. \right. \right. \right. \\ \left. \left. \left. \frac{1}{\sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (s + a_{i_2, i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2, i_5} (a_{i_2, i_5} - a_{i_2, i_4})} \right\} \right] \right] \quad ; \forall i = 1, 2, \dots, n \quad (\text{G.26})$$

171

It must be noted that the solution formulation as given by equation(G.26) is valid only when the condition ‘ $a_{i_2, i_5} \neq a_{i_2, i_4}$ ’ is satisfied. Equation(G.26) can be further factorized and simplified as:

$$c_i^1(x, t) = \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \{ G_1^1 + h(G_1^1) G_2^1 \} \right]$$

; $\forall i = 1, 2, \dots, n$

where ;

$$G_1^1 = \frac{\sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i B_{i_1}^{i_3} \ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_0(s+\lambda_{i_3})} \right\} e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}}}{(s + a_{i_3,0})(s + a_{i_2,i_4})} \right\rangle}{\left(\prod_{i_5=i_1, (i_5 \neq i_2, R_{i_5} = R_{i_2})}^i -k_{i_2,i_5} \right) (-R_{i_2,i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2,i_5} (a_{i_2,i_5} - a_{i_2,i_4})}$$

$$G_2^1 = \frac{B_{i_1}^{i_3} \ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_0(s+\lambda_{i_3})} \right\} e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}}}{(s + a_{i_3,0})} \right\rangle}{\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2,i_4}}$$

(G.27)

$$h(M) = \begin{cases} 1 & \text{if } M \text{ loop is not executed} \\ 0 & \text{if } M \text{ loop is executed} \end{cases}$$

The term ‘ G_1^1 ’ in equation(G.27) can be further factorized and simplified as:

$$G_1^1 = \sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i \frac{B_{i_1}^{i_3} \left[\ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_0(s+\lambda_{i_3})} \right\} e^{-\frac{(k_{i_2}+sR_{i_2})x}{v}}}{(s+a_{i_3,0})} \right\rangle - \ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_0(s+\lambda_{i_3})} \right\} e^{-\frac{(k_{i_2}+sR_{i_2})x}{v}}}{(s+a_{i_2,i_4})} \right\rangle \right]}{(a_{i_2,i_4} - \lambda_{i_3}) \left(\prod_{i_5=i_1, (i_5 \neq i_2, R_{i_5} = R_{i_2})}^i -k_{i_2,i_5} \right) (-R_{i_2,i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2,i_5} (a_{i_2,i_5} - a_{i_2,i_4})} \quad (G.28)$$

Again it must be noted that equation(G.28) is valid only when the condition ‘ $a_{i_2,i_4} \neq a_{i_3,0}$ ’ where; ‘ $a_{i_3,0} = \lambda_{i_3}$ ’ is satisfied. Using the Appendix H, inverse Laplace expressions for the terms ‘ G_1^1 ’ and ‘ G_2^1 ’ can be evaluated and the solution for ‘ $c_i^1(x,t)$ ’ is obtained as:

$$c_i^1(x, t) = \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \{ G_1^1 + h(G_1^1) G_2^1 \} \right]$$

; $\forall i = 1, 2, \dots, n$

where ;

$$G_1^1 = \sum_{i_4=i_1, (i_4 \neq i_2, R_{i_4} \neq R_{i_2})}^i \frac{B_{i_1}^{i_3} \left\langle \begin{array}{l} F_{i_2, i_3, 0} [x, t] - e^{-t_0 \lambda_{i_3}} F_{i_1, i_2, i_3} [x, (t - t_0)] \\ -F_{i_2, i_2, i_4} [x, t] + e^{-t_0 \lambda_{i_3}} F_{i_2, i_2, i_4} [x, (t - t_0)] \end{array} \right\rangle}{(a_{i_2, i_4} - \lambda_{i_3}) \left(\prod_{i_5=i_1, (i_5 \neq i_2, R_{i_5} = R_{i_2})}^i -k_{i_2, i_5} \right) (-R_{i_2, i_4}) \prod_{i_5=i_1, (i_5 \neq i_2, i_5 \neq i_4, R_{i_5} \neq R_{i_2})}^i -R_{i_2, i_5} (a_{i_2, i_5} - a_{i_2, i_4})}$$

$$G_2^1 = \frac{B_{i_1}^{i_3} \left\langle F_{i_2, i_3, 0} [x, t] - e^{-t_0 \lambda_{i_3}} F_{i_1, i_2, i_3} [x, (t - t_0)] \right\rangle}{\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4}} \quad (G.29)$$

where; the term ' F_{i_1, i_2, i_3} ' is given by equation(H.5) in Appendix H. The second term ' $\ell^{-1} \langle p_i^2(x, s) \rangle$ ', is evaluated as

follows. From equations (G.22) and (G.23) we get:

$$c_i^2(x, t) = \ell^{-1} \langle p_i^2(x, s) \rangle = \ell^{-1} \left\langle \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left\{ e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}} - e^{-\mu_{i_1}x} \right\}}{\sum_{i_2=i_1}^i (\mu_{i_1} v - k_{i_2} - sR_{i_2}) \prod_{i_3=i_1, (i_3 \neq i_2)}^i -(k_{i_2} + sR_{i_2} - k_{i_3} - sR_{i_3})} \right] \right\rangle \quad (G.30)$$

; $\forall i = 1, 2, \dots, n$

Equation(G.30) can be written as:

$$c_i^2(x, t) = \ell^{-1} \left\langle \sum_{i_1=1}^i \left[\frac{R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \left\{ e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}} - e^{-\mu_{i_1}x} \right\}}{\sum_{i_2=i_1}^i -(s + a_{i_1, -i_2}) R_{i_2} \left(\prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3} \right) \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i -R_{i_2, i_3} (s + a_{i_2, i_3})} \right] \right\rangle \quad (G.31)$$

; $\forall i = 1, 2, \dots, n$

Note: the term ' a_{i_1, i_2} ' is modified as:

$$a_{i_1, i_2} = \begin{cases} \frac{k_{i_1, i_2}}{R_{i_1, i_2}} ; \text{ when } i_2 > 0 \\ \lambda_{i_1} ; \text{ when } i_2 = 0 \\ \frac{-\mu_{i_1} \nu + k_{i_2}}{R_{i_2}} ; \text{ when } i_2 < 0 \end{cases} \quad (\text{G.32})$$

Note that the condition ‘ $k_{i_2, i_3} \neq 0$ ’ must be satisfied in equation(G.31). Factorization of equation(G.31) yields:

$$c_i^2(x, t) = \sum_{i_1=1}^i \left[R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \{ G_1^2 + h(G_1^2) G_2^2 \} \right]$$

; $\forall i = 1, 2, \dots, n$

where ;

177

$$G_1^2 = \frac{\ell^{-1} \left\langle \frac{\left\{ e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}} - e^{-\mu_{i_1}x} \right\}}{(s + a_{i_1, -i_2})(s + a_{i_2, i_3})} \right\rangle}{\sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (a_{i_2, i_4} - a_{i_2, i_3})}$$

$$G_2^2 = \frac{-\ell^{-1} \left\langle \frac{\left\{ e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}} - e^{-\mu_{i_1}x} \right\}}{(s + a_{i_1, -i_2})} \right\rangle}{R_{i_2} \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3}}$$

(G.33)

Note that the condition ‘ $a_{i_2,i_4} \neq a_{i_2,i_3}$ ’ must be satisfied in equation(G.33). The term ‘ G_1^2 ’ in equation(G.33) can be further factorized and simplified as:

$$G_1^2 = \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i \ell^{-1} \left(\frac{\left\{ e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}} - e^{-\mu_{i_1}x} \right\}}{(s + a_{i_1, -i_2})} - \frac{\left\{ e^{\frac{-(k_{i_2} + sR_{i_2})x}{v}} - e^{-\mu_{i_1}x} \right\}}{(s + a_{i_2, i_3})} \right) \frac{(a_{i_2, i_3} - a_{i_1, -i_2}) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \left(\prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} \right) (a_{i_2, i_4} - a_{i_2, i_3})}{(G.34)}$$

Note that the condition ‘ $a_{i_2, i_4} \neq a_{i_1, -i_2}$ ’ must be satisfied in equation(G.34). ‘ G_1^2 ’ can be further simplified as:

$$G_1^2 = \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i \frac{\left[\begin{array}{l} \ell^{-1} \left\langle \frac{e^{-\frac{(k_{i_2} + sR_{i_2})x}{v}}}{(s + a_{i_1, -i_2})} \right\rangle - e^{-\mu_{i_1} x} \ell^{-1} \left\langle \frac{1}{(s + a_{i_1, -i_2})} \right\rangle \\ -\ell^{-1} \left\langle \frac{e^{-\frac{(k_{i_2} + sR_{i_2})x}{v}}}{(s + a_{i_2, i_3})} \right\rangle + e^{-\mu_{i_1} x} \ell^{-1} \left\langle \frac{1}{(s + a_{i_2, i_3})} \right\rangle \end{array} \right]}{(a_{i_2, i_3} - a_{i_1, -i_2}) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (a_{i_2, i_4} - a_{i_2, i_3})} \quad (G.35)$$

179

' G_2^2 ' in equation(G.33) can be further simplified as:

$$G_2^2 = \frac{\left[\ell^{-1} \left\langle \frac{e^{-\frac{(k_{i_2} + sR_{i_2})x}{v}}}{(s + a_{i_1, -i_2})} \right\rangle - e^{-\mu_{i_1} x} \ell^{-1} \left\langle \frac{1}{(s + a_{i_1, -i_2})} \right\rangle \right]}{R_{i_2} \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3}} \quad (G.36)$$

From inverse Laplace transform tables we get: (see [8], p494, eq3).

$$\ell^{-1} \left\langle \frac{1}{(s + a_{i_1, i_2})} \right\rangle = e^{-a_{i_1, i_2} t} \quad (\text{G.37})$$

Using equation(G.37) and Appendix H, inverse Laplace expressions for the terms ‘ G_1^2 ’ and ‘ G_2^2 ’ can be evaluated and the solution for ‘ $c_i^2(x, t)$ ’ is obtained as:

$$c_i^2(x, t) = \sum_{i_1=1}^i \left[R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \{ G_1^2 + h(G_1^2) G_2^2 \} \right] ; \forall i = 1, 2, \dots, n$$

where ;

$$G_1^2 = \sum_{i_3=i_1, (i_3 \neq i_2, R_{i_3} \neq R_{i_2})}^i \frac{\left\langle F_{i_2, i_1, -i_2} [x, t] - e^{(-\mu_{i_1} x - a_{i_1, -i_2} t)} - F_{i_2, i_2, i_3} [x, t] + e^{(-\mu_{i_1} x - a_{i_2, i_3} t)} \right\rangle}{(a_{i_2, i_3} - a_{i_1, -i_2}) R_{i_2} \left(\prod_{i_4=i_1, (i_4 \neq i_2, R_{i_4} = R_{i_2})}^i -k_{i_2, i_4} \right) R_{i_2, i_3} \prod_{i_4=i_1, (i_4 \neq i_2, i_4 \neq i_3, R_{i_4} \neq R_{i_2})}^i -R_{i_2, i_4} (a_{i_2, i_4} - a_{i_2, i_3})}$$

$$G_2^2 = \frac{-\left\langle F_{i_2, i_1, -i_2} [x, t] - e^{(-\mu_{i_1} x - a_{i_1, -i_2} t)} \right\rangle}{R_{i_2} \prod_{i_3=i_1, (i_3 \neq i_2, R_{i_3} = R_{i_2})}^i -k_{i_2, i_3}} \quad (\text{G.38})$$

where; the term ‘ F_{i_1, i_2, i_3} ’ is given by equation(H.5) in Appendix H. The final solution is obtained by substituting equations(G.29) and (G.38) into equation(G.23).

$$\begin{aligned}
 c_i(x, t) = & \sum_{i_1=1}^i \left[\left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \sum_{i_3=1}^{i_1} \{ G_1^1 + h(G_1^1) G_2^1 \} \right] \\
 & + \sum_{i_1=1}^i \left[R_{i_1} c_{i_1}^o \left(\prod_{i_2=i_1+1}^i y_{i_2} k_{i_2-1} \right) \sum_{i_2=i_1}^i \{ G_1^2 + h(G_1^2) G_2^2 \} \right] \quad (G.39) \\
 & ; \forall i = 1, 2, \dots, n
 \end{aligned}$$

APPENDIX H

Evaluation of Inverse Laplace Expressions for the Zero Dispersion Case

$$\mathcal{X}_{i_1, i_2, i_3, i_4} [x, t] = \ell^{-1} \left\langle \frac{\left\{ 1 - e^{-t_o(s + \lambda_{i_4})} \right\} e^{-\frac{(k_{i_1} + sR_{i_1})x}{v}}}{(s + a_{i_2, i_3})} \right\rangle \quad (\text{H.1})$$

Equation(H.1) can be simplified as:

$$\mathcal{X}_{i_1, i_2, i_3, i_4} [x, t] = e^{-\frac{xk_{i_1}}{v}} \ell^{-1} \left\langle \frac{e^{-\left(\frac{R_{i_1}x}{v}\right)s}}{(s + a_{i_2, i_3})} - \frac{e^{-t_o\lambda_{i_4}} e^{-\left(\frac{R_{i_1}x}{v} + t_o\right)s}}{(s + a_{i_2, i_3})} \right\rangle \quad (\text{H.2})$$

Invoking the ‘Second Shifting Theorem’ (see [28], p267) we can simplify

‘ $\mathcal{X}_{i_1, i_2, i_3, i_4}$ ’, as:

$$\chi_{i_1, i_2, i_3, i_4} [x, t] = e^{\frac{-xk_{i_1}}{v}} \left\langle u \left(t - \frac{R_{i_1} x}{v} \right) f \left(t - \frac{R_{i_1} x}{v} \right) - e^{-t_o \lambda_{i_4}} u \left(t - t_o - \frac{R_{i_1} x}{v} \right) f \left(t - t_o - \frac{R_{i_1} x}{v} \right) \right\rangle \quad (\text{H.3})$$

$$\text{where ; } f(t) = \ell^{-1} \left\langle \frac{1}{(s + a_{i_2, i_3})} \right\rangle$$

‘ $f(t)$ ’ can be evaluated using equation(G.37). Using equation(G.37) in equation(H.3) we get:

$$\chi_{i_1, i_2, i_3, i_4} [x, t] = e^{\frac{-xk_{i_1}}{v}} \left\langle u \left(t - \frac{R_{i_1} x}{v} \right) e^{-a_{i_2, i_3} \left(t - \frac{R_{i_1} x}{v} \right)} - e^{-t_o \lambda_{i_4}} u \left(t - t_o - \frac{R_{i_1} x}{v} \right) e^{-a_{i_2, i_3} \left(t - t_o - \frac{R_{i_1} x}{v} \right)} \right\rangle \quad (\text{H.4})$$

Equation(H.4) can be compactly expressed as:

$$\chi_{i_1, i_2, i_3, i_4} [x, t] = e^{\frac{-xk_{i_1}}{v}} \left\langle F_{i_1, i_2, i_3} [x, t] - e^{-t_o \lambda_{i_4}} F_{i_1, i_2, i_3} [x, (t - t_o)] \right\rangle \quad (\text{H.5})$$

where ;

$$F_{i_1, i_2, i_3} [x, t] = u \left(t - \frac{R_{i_1} x}{v} \right) e^{-a_{i_2, i_3} \left(t - \frac{R_{i_1} x}{v} \right)}$$

APPENDIX I

FORTRAN Code SEQUENTIAL

```
C      ANALYTICAL SOLUTION FOR MULTI-SPECIES SOLUTE TRANSPORT
C      CODED BY: VENKATRAMAN SRINIVASAN
C              AUBURN UNIVERSITY
C              JULY 2006
C
C      PROGRAM SEQUENTIAL
C              IMPLICIT NONE
C
C      PROBLEM PARAMETERS
C
C      INTEGER N,BC,ADS
C      DOUBLE PRECISION V,AX,AY,AZ,LX,LY,LZ,DELX,DELY,DELZ,T,TA,DELT,
$              YS,ZS,X
C      DOUBLE PRECISION, ALLOCATABLE :: R(:),KM(:),Y(:),C(:),
$              LAM(:),B(:,,:),CO(:),MU(:)
C
C      PROGRAM PARAMETERS
C
C      CHARACTER FILENAME*80,TITLE*80,COMMENT*80
C      INTEGER I,I1,I2,I3,I4,I5,PARA,NEGPARA
C      DOUBLE PRECISION,ALLOCATABLE :: KIJ(:,,:),KAIJKL(:,,:,:),RIJ(:,,:)
$              ,AIJ(:,,:),FIJK(:,,:,:)
C      DOUBLE PRECISION TIME,TEMP,TEMP1,TEMP2,TEMP3,TEMP4
C
C      READ THE INPUT FILE NAME
C
C      WRITE(*,*) "ENTER THE NAME OF THE INPUT FILE"
C      READ(*,*) FILENAME
C
C      OPEN(5,FILE=FILENAME,FORM='FORMATTED',STATUS='UNKNOWN')
C      OPEN(10,FILE="OUTPUT.DAT",FORM='FORMATTED',STATUS='UNKNOWN')
C
C      READ(5,10) TITLE
C      WRITE(10,10) TITLE
C      READ(5,*) N,COMMENT !NUMBER OF SPECIES
C      ALLOCATE (R(N),KM(N),Y(N),LAM(N),B(N,N),CO(N),MU(N))
C
C      READING THE PROBLEM GEOMETRY
C
C      READ (5,*) LX,DELX,LY,DELY,LZ,DELZ,COMMENT !PROBLEM GEOMETRY
C      READ (5,*) YS,ZS,COMMENT !SOURCE GEOMETRY
C      READ (5,*) T,TA,DELT,COMMENT !SIMULATION TIME, PULSE TIME, TIME STEP
C
C      READING THE PROBLEM PARAMETERS
C
C      READ(5,*) V,AX,AY,AZ,COMMENT !ADVECTION DISPERSION PARAMETERS
C      READ(5,*) (R(I),I = 1, N), COMMENT !RETARDATION FACTORS
C      READ(5,*) (KM(I),I = 1, N), COMMENT !REACTION CONSTANTS
C      READ(5,*) (Y(I),I = 2, N), COMMENT !YIELD COEFFICIENTS
C      READ(5,*) BC, COMMENT !BOUNDARY CONDITION
C              IF BC.EQ.0 THEN CONCENTRATION BOUNDARY
C              IF BC.NE.0 THEN FLUX BOUNDARY
C      READ(5,*) (LAM(I),I = 1, N), COMMENT !SOURCE DECAY
C      DO I = 1, N
C          READ(5,*) (B(I,I1),I1 = 1, I), COMMENT !BATEMAN BOUNDARY VALUES
```

```

END DO
READ(5,*) ADS !REACTION CONDITION
C      IF ADS.EQ.0, REACTION IN LIQUID PHASE
C      IF ADS.NE.0, REACTION IN BOTH SOLID AND LIQUID PHASES
READ(5,*) (CO(I), I = 1, N), COMMENT !INITIAL CONDITION CONCENTRATION
READ(5,*) (MU(I), I = 1, N), COMMENT !INITIAL CONDITION EXPONENTIAL DECAY
CLOSE (5)
C
IF (ADS .NE. 0) THEN
  DO I = 1, N
    KM(I) = R(I)*KM(I)
  END DO
END IF
C
C      ALLOCATING THE PROGRAM PARAMETERS
C
C      ALLOCATE (KAIJKL(N,N,0:N,N),FIJK(N,N,-N:N),
$      KIJ(N,N),RIJ(N,N),AIJ(N,-N:N))
C
CALL CKIJ(KIJ,KM,N)
CALL CRIJ(RIJ,R,N)
CALL CAIJ(AIJ,KIJ,RIJ,LAM,R,KM,MU,AX,V,N)
C
C      ALLOCATING THE CONCENTRATION VECTOR MATRIX
C
C      ALLOCATE (C(N))
C
C      CHECKING CONDITIONS OF PARAMETERS FOR DETERMINATE SOLUTION
C
C      PARA = 0
C      IF ANY OF THE CONDITIONS ARE NOT SATISFIED THEN PARA = 1
C      IF PARA = 1, THEN PROGRAM STOPS
C
C      CONDITION 1: KIJ(I2,I4) .NE. 0
C
DO I2 = 1, N
  DO I4 = 1, N
    IF ((I2 .NE. I4) .AND. (R(I2) .EQ. R(I4)) .AND.
$      (KM(I2) .EQ. KM(I4))) THEN
      PARA = 1
      WRITE(10,*) "SOLUTION INDETERMINATE DUE TO PARAMETER
$VIOLATION"
      WRITE(10,*) "CONDITION KIJ(",I2,",",I4,") = 0 ;
$WHEN RIJ(",I2,",",I4,") = 0) IS VIOLATED"
      END IF
    END DO
  END DO
C
C      CONDITION 2: AIJ(I2,I5) .NE. AIJ(I2,I4)
C
DO I2 = 1, N
  DO I4 = 1, N
    IF ((I2 .NE. I4) .AND. (R(I2) .NE. R(I4))) THEN
      DO I5 = 1, N
        IF ((I5 .NE. I2) .AND. (I5 .NE. I4) .AND.
$          (R(I5) .NE. R(I2)) .AND.
$          (AIJ(I2,I5) .EQ. AIJ(I2,I4))) THEN
          PARA = 1
          WRITE(10,*) "SOLUTION INDETERMINATE DUE TO
$PARAMETER VIOLATION"
          WRITE(10,*) "CONDITION AIJ(",I2,",",I5,") =
$AIJ(",I2,",",I4,") ;WHEN RIJ(",I2,",",I4,") AND RIJ(",I2,",",I5,")
$ NE 0 IS VIOLATED"
          END IF
        END DO
      END IF
    END DO
  END DO
C

```

```

C      CONDITION 3&4: AIJ(I2,I4) .NE. LAM(I3) & AIJ(I2,I4) .NE. AIJ(I1,-I2)
C
      DO I1 = 1, N
        DO I2 = I1, N
          DO I3 = 1, I1
            DO I4 = I1, N
              IF ((I4 .NE. I2) .AND. (R(I4) .NE. R(I2))
$              .AND. (AIJ(I2,I4) .EQ. LAM(I3))) THEN
                PARA = 1
                WRITE(10,*) "SOLUTION INDETERMINATE DUE TO
$PARAMETER VIOLATION"
                WRITE(10,*) "CONDITION AIJ(",I2,",",I4,") =
$LAM(",I3,") ; RIJ(",I4,",",I2,") NE 0 IS VIOLATED"
                ELSE IF ((I4 .NE. I2) .AND. (R(I4) .NE. R(I2))
$                .AND. (AIJ(I2,I4) .EQ. AIJ(I1,-I2))) THEN
                  PARA = 1
                  WRITE(10,*) "SOLUTION INDETERMINATE DUE TO
$PARAMETER VIOLATION"
                  WRITE(10,*) "CONDITION AIJ(",I2,",",I4,") =
$AIJ(",I1,",",-I2,") ; RIJ(",I4,",",I2,") NE 0 IS VIOLATED"
                  END IF
                END DO
              END DO
            END DO
          END DO
        END DO
      END DO
C
C      STARTING THE TIME LOOP
C
      IF (PARA .EQ. 0) THEN !CHECKING THE PARAMETER CONDITIONS
        TIME = 0.0
        DO WHILE (TIME .LT. T)
          TIME = TIME + DELT
          WRITE(10,*) "TIME = ", TIME
          WRITE (10,20) (I, I = 1, N)
C
C      STARTING THE DISTANCE LOOP
C
          X = 0.0
          DO WHILE (X .LT. LX)
            X = X + DELX
C
C      STARTING THE LOOP FOR CALCULATING THE CONCENTRATION OF ALL SPECIES
C
            DO I = 1, N
              TEMP = 0.0
              DO I1 = 1, I
                TEMP1 = 1.0
                DO I2 = I1+1, I
                  TEMP1 = TEMP1*Y(I2)*KM(I2-1)
                END DO !I2 LOOP (1)
                TEMP2 = 0.0
                DO I2 = I1, I
C
C      EXPONENTIALLY DECAYING INITIAL CONDITION
C
                  PARA = 0
                  IF (CO(I1) .NE. 0.0) THEN
C
C
                    WRITE (*,*) "INITIAL LOOP"
                    DO I3 = I1, I
                      IF ((I3 .NE. I2) .AND.
$                      (R(I2) .NE. R(I3))) THEN
                        PARA = 1
                        TEMP3 = 0.0
                        NEGPARA = 1.0
                        DO I4 = I1, I
                          IF ((I4 .NE. I2) .AND.
$                          (I4 .NE. I3) .AND.
$                          (R(I2) .NE. R(I4))) THEN

```

```

          IF ((-KIJ(I2,I4)+RIJ(I2,I4)
$          *AIJ(I2,I3)) .LT. 0.0) THEN
            NEGPARA = NEGPARA*(-1)
          END IF
$          TEMP3 = TEMP3+LOG(ABS(
            (-KIJ(I2,I4)+RIJ(I2,I4)*AIJ(I2,I3)))
          END IF
        END DO !I4 LOOP (1)
        DO I4 = I1, I
          IF ((I4 .NE. I2) .AND. (R(I2) .EQ. R(I4))) THEN
            IF (-KIJ(I2,I4) .LT. 0.0) THEN
              NEGPARA = NEGPARA*(-1)
            END IF
            TEMP3 = TEMP3+LOG(ABS(-KIJ(I2,I4)))
          END IF
        END DO !I4 LOOP (2)
$        IF ((KIJ(I2,I3)-RIJ(I2,I3)*
          AIJ(I1,-I2))*R(I2)) .LT. 0.0) THEN
          NEGPARA = NEGPARA*(-1)
        END IF
$        TEMP3 = TEMP3+LOG(ABS((KIJ(I2,I3)
          -RIJ(I2,I3)*AIJ(I1,-I2))*R(I2)))
$        CALL CFIJK(FIJK,AIJ,LAM,TIME,X,V,
$        AX,R,KM,N,I2,I1,-I2,BC)
$        CALL CFIJK(FIJK,AIJ,LAM,TIME,X,V,
$        AX,R,KM,N,I2,I2,I3,BC)
$        TEMP4 = DEXP(-MU(I1)*X)*(DEXP(-AIJ(I1,-I2)*TIME)-
$        DEXP(-AIJ(I2,I3)*TIME))
          IF (ABS(TEMP4) .GT. 1E+10) TEMP4 = 0.0
C        CONDITION FOR DIRICHLET BOUNDARY
          IF (BC .EQ. 0) THEN
            TEMP2 = TEMP2 + R(I1)*CO(I1)*
$            (FIJK(I2,I1,-I2)-FIJK(I2,I2,I3)
$            -TEMP4)/(DEXP(TEMP3)*NEGPARA)
C        CONDITION FOR NEWMAN BOUNDARY
          ELSE
            TEMP2 = TEMP2 + R(I1)*CO(I1)*
$            ((1.0+MU(I1)*AX)*
$            (FIJK(I2,I1,-I2)-FIJK(I2,I2,I3))
$            -TEMP4)/(DEXP(TEMP3)*NEGPARA)
          END IF
        END IF
      END DO !I3 LOOP
      IF (PARA .EQ. 0) THEN
        TEMP3 = 0.0
        NEGPARA = 1
        DO I3 = I1, I
          IF ((I3 .NE. I2) .AND. (R(I2) .EQ. R(I3))) THEN
            IF ((-KIJ(I2,I3)) .LT. 0.0) THEN
              NEGPARA = NEGPARA*(-1)
            END IF
            TEMP3 = TEMP3+LOG(ABS((-KIJ(I2,I3))))
          END IF
        END DO !I3 LOOP (2)
        TEMP3 = TEMP3+LOG(R(I2))
        CALL CFIJK(FIJK,AIJ,LAM,TIME,X,V,
$        AX,R,KM,N,I2,I1,-I2,BC)
        TEMP4 = DEXP(-MU(I1)*X-AIJ(I1,-I2)*TIME)
        IF (ABS(TEMP4) .GT. 1E+10) TEMP4 = 0.0
C        CONDITION FOR DIRICHLET BOUNDARY
          IF (BC .EQ. 0) THEN
            TEMP2 = TEMP2+R(I1)*CO(I1)*(-FIJK(I2,I1,-I2)+
$            TEMP4)/(DEXP(TEMP3)*NEGPARA)
C        CONDITION FOR NEWMAN BOUNDARY
          ELSE
            TEMP2 = TEMP2+R(I1)*CO(I1)*
$            (- (1.0+MU(I1)*AX)*FIJK(I2,I1,-I2)
$            +TEMP4)/(DEXP(TEMP3)*NEGPARA)
          END IF

```



```

        END IF
        END IF
C
C      BOUNDARY CONDITION
C
        DO I3 = 1, I1
          IF (B(I1,I3) .NE. 0.0) THEN
C            WRITE (*,*) "BOUNDARY LOOP"
            PARA = 0
            DO I4 = I1, I
              IF ((I4 .NE. I2) .AND. (R(I2) .NE. R(I4))) THEN
                PARA = 1
                TEMP3 = 0.0
                NEGPARA = 1
                DO I5 = I1, I
                  IF ((I5 .NE. I2) .AND. (I5 .NE. I4) .AND.
$                   (R(I2) .NE. R(I5))) THEN
$                     IF ((-KIJ(I2,I5)+RIJ(I2,I5)
$                       *AIJ(I2,I4)) .LT.0.0) THEN
                         NEGPARA = NEGPARA*(-1)
                         END IF
$                       TEMP3 = TEMP3+LOG(ABS((-KIJ(I2,I5)+
$                         RIJ(I2,I5)*AIJ(I2,I4))))
                         END IF
                    END DO !I5 LOOP (1)
                    DO I5 = I1, I
                      IF ((I5 .NE. I2) .AND. (R(I2) .EQ. R(I5))) THEN
                        IF ((-KIJ(I2,I5)) .LT. 0.0) THEN
                          NEGPARA = NEGPARA*(-1)
                        END IF
                        TEMP3 = TEMP3+LOG(ABS((-KIJ(I2,I5))))
                      END IF
                    END DO !I5 LOOP (2)
                    IF ((-KIJ(I2,I4)+RIJ(I2,I4)*LAM(I3)) .LT.0.0) THEN
                      NEGPARA = NEGPARA*(-1)
                    END IF
                    TEMP3 = TEMP3+LOG(ABS((-KIJ(I2,I4)
$                   +RIJ(I2,I4)*LAM(I3))))
                    CALL CKAIJKL(KAIJKL,AIJ,LAM,TIME
$                   ,X,V,AX,R,KM,TA,N,I2,I3,0,I3,BC)
                    CALL CKAIJKL(KAIJKL,AIJ,LAM,TIME
$                   ,X,V,AX,R,KM,TA,N,I2,I2,I4,I3,BC)
                    TEMP2 = TEMP2 + B(I1,I3)*(KAIJKL(I2,I3,0,I3)-
$                   KAIJKL(I2,I2,I4,I3))/
$                   (DEXP(TEMP3)*NEGPARA)
                    END IF
                END DO !I4 LOOP (1)
                IF (PARA .EQ. 0) THEN
                  TEMP3 = 0.0
                  NEGPARA = 1
                  DO I4 = I1, I
                    IF ((I4 .NE. I2) .AND. (R(I2) .EQ. R(I4))) THEN
                      IF ((-KIJ(I2,I4)) .LT.0.0) THEN
                        NEGPARA = NEGPARA*(-1)
                      END IF
                      TEMP3 = TEMP3+LOG(ABS((-KIJ(I2,I4))))
                    END IF
                  END DO !I4 LOOP (2)
                  CALL CKAIJKL(KAIJKL,AIJ,LAM,TIME
$                   ,X,V,AX,R,KM,TA,N,I2,I3,0,I3,BC)
                  TEMP2 = TEMP2 + B(I1,I3)*KAIJKL(I2,I3,0,I3)/
$                   (DEXP(TEMP3)*NEGPARA)
                END IF
              END IF
            END DO !I3 LOOP
          END DO !I2 LOOP (2)
          TEMP = TEMP + TEMP1*TEMP2
        END DO !I1 LOOP
        C(I) = TEMP

```

```

        END DO
C
C      WRITING THE OUTPUT
C
        WRITE (10,30) X, (C(I), I = 1, N)
C
C      END OF DISTANCE LOOP
C
        END DO
C
C      END OF TIME LOOP
C
        END DO
        END IF
10      FORMAT (A80)
20      FORMAT ('          DISTANCE', <N>("          SPECIES", I3.3))
30      FORMAT (<(N+1)>E15.5)
        CLOSE(10)
        END PROGRAM
C
C      SUBROUTINE FOR CALCULATING THE KIJ COEFFICIENTS
C
        SUBROUTINE CKIJ(KIJ,KM,N)
        IMPLICIT NONE
        INTEGER N,I1,I2
        DOUBLE PRECISION KIJ(N,N),KM(N)
        DO I1 = 1, N
            DO I2 = 1, N
                KIJ(I1,I2) = KM(I1) - KM(I2)
            END DO
        END DO
        RETURN
        END
C
C      SUBROUTINE FOR CALCULATING THE RIJ COEFFICIENTS
C
        SUBROUTINE CRIJ(RIJ,R,N)
        IMPLICIT NONE
        INTEGER N,I1,I2
        DOUBLE PRECISION RIJ(N,N),R(N)
        DO I1 = 1, N
            DO I2 = 1, N
                RIJ(I1,I2) = R(I1) - R(I2)
            END DO
        END DO
        RETURN
        END
C
C      SUBROUTINE FOR CALCULATING THE AIJ COEFFICIENTS
C
        SUBROUTINE CAIJ(AIJ,KIJ,RIJ,LAM,R,KM,MU,AX,V,N)
        IMPLICIT NONE
        INTEGER N,I1,I2
        DOUBLE PRECISION AIJ(N,-N:N),KIJ(N,N),RIJ(N,N),LAM(N),R(N),KM(N),
        $          MU(N),AX,V
C
C      INITIALISING THE VALUE OF AIJ
C
        DO I1 = 1, N
            DO I2 = 0, N
                AIJ(I1,I2) = 0.0
            END DO
        END DO
        DO I1 = 1, N
            DO I2 = -N, N
                IF (I1 .NE. I2) THEN
                    IF (I2 .LT. 0) THEN
                        AIJ(I1,I2) = (-MU(I1)**2.0*AX*V-MU(I1)*V+KM(-I2))/R(-I2)
                    ELSE IF (I2 .EQ. 0) THEN

```

```

        AIJ(I1,I2) = LAM(I1)
        ELSE IF (R(I1) .NE. R(I2)) THEN
            AIJ(I1,I2) = KIJ(I1,I2) / RIJ(I1,I2)
        END IF
    END IF
END DO
RETURN
END

C
C
C
SUBROUTINE FOR CALCULATING KAIJJKL COEFFICIENTS
C
C
SUBROUTINE CKAIJJKL(KAIJJKL,AIJ,LAM,TIME,X,V,AX,R,KM,TA,N,I1,I2,
$ I3,I4,BC)
    IMPLICIT NONE
    INTEGER N,I1,I2,I3,I4,BC
    DOUBLE PRECISION KAIJJKL(N,N,0:N,N),AIJ(N,-N:N),LAM(N),TIME,X,V,
$ AX,R(N),KM(N),TA,FIJK(N,N,-N:N),TEMP
    CALL CFIJK(FIJK,AIJ,LAM,TIME,X,V,AX,R,KM,N,I1,I2,I3,BC)
    KAIJJKL(I1,I2,I3,I4) = FIJK(I1,I2,I3)
    IF (TIME .GT. TA) THEN
        TEMP = TIME-TA
        CALL CFIJK(FIJK,AIJ,LAM,TEMP,X,V,AX,R,KM,N,I1,I2,I3,BC)
        KAIJJKL(I1,I2,I3,I4) = KAIJJKL(I1,I2,I3,I4) -
$ DEXP(-LAM(I4)*TA)*FIJK(I1,I2,I3)
    END IF
    RETURN
    END

C
C
C
SUBROUTINE FOR CALCULATING THE FIJK COEFFICIENTS
C
C
SUBROUTINE CFIJK(FIJK,AIJ,LAM,TIME,X,V,AX,R,KM,N,I1,I2,I3,BC)
    IMPLICIT NONE
    INTEGER N,I1,I2,I3,BC,PARA
    DOUBLE PRECISION FIJK(N,N,-N:N),AIJ(N,-N:N),LAM(N),TIME,X,V,AX
$ ,R(N),KM(N),WIJK,PI,CALEXF,TEMP1,TEMP2,
$ TEMP3,TEMP4,TEMP5,TEMP6,TEMP7
    PI = 22.0/7.0

C
C
C
CALCULATING THE WIJ TERMS
C
CALL CWIJK(WIJK,V,R,AX,KM,AIJ,N,I1,I2,I3,PARA)

C
C
C
CASE FOR CONCENTRATION BOUNDARY CONDITION
C
IF (BC .EQ. 0) THEN

C
C
C
CASE FOR NORMAL ERROR FUNCTION
C
    IF (PARA .EQ. 0) THEN
        TEMP1 = -AIJ(I2,I3)*TIME + (V-WIJK)*X/(2.0*AX*V)
        TEMP2 = (R(I1)*X-WIJK*TIME)/(4.0*R(I1)*AX*V*TIME)**0.5
        FIJK(I1,I2,I3) = 0.5*CALEXF(TEMP1,TEMP2)

C
C
C
CASE WHEN X << AX
C
    IF (X .LT. (100.0*AX)) THEN
        TEMP3 = -AIJ(I2,I3)*TIME + (V+WIJK)*X/(2.0*AX*V)
        TEMP4 = (R(I1)*X+WIJK*TIME)/(4.0*R(I1)*AX*V*TIME)**0.5
        FIJK(I1,I2,I3)=FIJK(I1,I2,I3)+0.5*CALEXF(TEMP3,TEMP4)
    END IF

C
C
C
CASE FOR COMPLEX ERROR FUNCTION
C
    ELSE
        TEMP1 = R(I1)*X/(4.0*R(I1)*AX*V*TIME)**0.5
        TEMP2 = WIJK*TIME/(4.0*R(I1)*AX*V*TIME)**0.5
        TEMP3 = -AIJ(I2,I3)*TIME+V*X/(2.0*AX*V)
        CALL CEXF(TEMP1,TEMP2,TEMP3,TEMP4,TEMP5)
    END IF

```

```

      FIJK(I1,I2,I3) = DCOS(WIJK*X/(2.0*V*AX))*TEMP4-
$      DSIN(WIJK*X/(2.0*V*AX))*TEMP5
      END IF
C
C      CASE FOR FLUX BOUNDARY CONDITION
C
      ELSE IF (BC .EQ. 1) THEN
C
      CASE FOR (KM(I1)/R(I1)) .NE. AIJ(I2,I3)
C
      IF ((KM(I1)/R(I1)) .NE. AIJ(I2,I3)) THEN
C
      CASE FOR NORMAL ERROR FUNCTION
C
      IF (PARA .EQ. 0) THEN
      TEMP1 = -AIJ(I2,I3)*TIME + (V-WIJK)*X/(2.0*AX*V)
      TEMP2 = (R(I1)*X-WIJK*TIME)/(4.0*R(I1)*AX*V*TIME)**0.5
      TEMP3 = -AIJ(I2,I3)*TIME + (V+WIJK)*X/(2.0*AX*V)
      TEMP4 = (R(I1)*X+WIJK*TIME)/(4.0*R(I1)*AX*V*TIME)**0.5
      TEMP5 = V*X/(V*AX)-KM(I1)*TIME/R(I1)
      TEMP6 = (R(I1)*X+V*TIME)/(4.0*R(I1)*V*AX*TIME)**0.5
      FIJK(I1,I2,I3) = V*(CALEXF(TEMP1,TEMP2)/(V+WIJK)
$      +CALEXF(TEMP3,TEMP4)/(V-WIJK))
$      +2.0*V**2.0/(4.0*R(I1)*V*AX*
$      (KM(I1)/R(I1)-AIJ(I2,I3)))*
$      CALEXF(TEMP5,TEMP6)
C
      CASE FOR COMPLEX ERROR FUNCTION
C
      ELSE
      TEMP1 = R(I1)*X/(4.0*AX*V*R(I1)*TIME)**0.5
      TEMP2 = WIJK*TIME/(4.0*AX*V*R(I1)*TIME)**0.5
      TEMP3 = -AIJ(I2,I3)*TIME+V*X/(2.0*AX*V)
      TEMP4 = V*X/(V*AX)-KM(I1)*TIME/R(I1)
      TEMP5 = (R(I1)*X+V*TIME)/(4.0*R(I1)*AX*V*TIME)**0.5
      CALL CEXF(TEMP1,TEMP2,TEMP3,TEMP6,TEMP7)
      FIJK(I1,I2,I3) = 2.0*V/(V**2.0+WIJK**2.0)*((TEMP6*V-TEMP7*
$      WIJK)*DCOS(X*WIJK/(2.0*V*AX))-(TEMP6*WIJK
$      +TEMP7*V)*DSIN(X*WIJK/(2.0*V*AX)))+V**2.0
$      /(2.0*R(I1)*V*AX*(KM(I1)/R(I1)-AIJ(I2,I3)
$      ))*CALEXF(TEMP4,TEMP5)
      END IF
C
      CASE FOR KM(I1) .EQ. AIJ(I2,I3)
C
      ELSE
C
      ORIGINAL FORMULATION
C
      TEMP1 = -KM(I1)*TIME/R(I1)
      TEMP2 = (R(I1)*X-V*TIME)/(4.0*R(I1)*V*AX*TIME)**0.5
      TEMP3 = -KM(I1)*TIME/R(I1)-(R(I1)*X-V*TIME)**2.0/
$      (4.0*R(I1)*V*AX*TIME)
      TEMP4 = -KM(I1)*TIME/R(I1)+V*X/(V*AX)
      TEMP5 = (R(I1)*X+V*TIME)/(4.0*R(I1)*V*AX*TIME)**0.5
      FIJK(I1,I2,I3) = 0.5*CALEXF(TEMP1,TEMP2)+
$      (V**2.0*TIME/(PI*R(I1)*AX*V))**0.5*DEXP(TEMP3)
$      -0.5*(1.0+X*V/(AX*V)+V**2.0*TIME/(R(I1)*AX*V))
$      *CALEXF(TEMP4,TEMP5)
C
      NEW FORMULATION
      REFORMULATING TO AVOID THE CONDITION WHEN THE ARGUMENT IN THE
      SECOND TERM (EXPONENTIAL FUNCTION) BECOMES ZERO AND THE ARGUMENT
      IN THE THIRD TERM (ERFC FUNCTION) BECOMES GREATER THAN 25
C
      WRITE (*,*) "EQUAL"
      TEMP1 = -KM(I1)*TIME/R(I1)
      TEMP2 = (R(I1)*X-V*TIME)/(4.0*R(I1)*V*AX*TIME)**0.5
      TEMP3 = -(R(I1)*X-V*TIME)**2.0/(4.0*R(I1)*V*AX*TIME)-X/AX

```

```

TEMP4 = -KM(I1)*TIME/R(I1)+V*X/(V*AX)
TEMP5 = (R(I1)*X+V*TIME)/(4.0*R(I1)*V*AX*TIME)**0.5
TEMP6 = 0.0
IF (TEMP5 .LT. 20.0) THEN
    FIJK(I1,I2,I3) = 0.5*CALEXF(TEMP1,TEMP2)+DEXP(TEMP4)*
$           (V**2.0*TIME/(PI*R(I1)*AX*V))**0.5*DEXP
$           (TEMP3)-0.5*(1.0+X*V/(AX*V)+V**2.0*TIME/
$           (R(I1)*AX*V))*CALEXF(TEMP6,TEMP5))
    ELSE
        FIJK(I1,I2,I3) = 0.5*CALEXF(TEMP1,TEMP2)
    END IF
END IF
END IF
IF (ABS(FIJK(I1,I2,I3)) .GE. 1E+8) THEN
C   WRITE (*,*) "FLAG, FIJK"
    FIJK(I1,I2,I3) = 0.0
END IF
RETURN
END

C
C   SUBROUTINE FOR CALCULATING THE WIJK COEFFICIENTS
C
SUBROUTINE CWIJK(WIJK,V,R,AX,KM,AIJ,N,I1,I2,I3,PARA)
IMPLICIT NONE
INTEGER N,I1,I2,I3,PARA
DOUBLE PRECISION WIJK,V,R(N),AX,KM(N),AIJ(N,-N:N),TEMP
PARA = 0
TEMP = V**2.0+4.0*R(I1)*V*AX*(KM(I1)/R(I1)-AIJ(I2,I3))
IF (TEMP .LT. 0.0) THEN
C   PARA = 1
    WRITE (*,*) "NEGATIVE"
END IF
WIJK=(ABS(TEMP))**0.5
RETURN
END

C
C   FUNCTION TO CALCULATE EXP(TEMP1)*ERFC(TEMP2)
C
FUNCTION CALEXF(TEMP1,TEMP2)
IMPLICIT NONE

C
C   NEW ERFC ROUTINE
C
DOUBLE PRECISION TEMP1,TEMP2,TEMP,CALEXF

C
CALL CALERF(TEMP2,TEMP,1)
CALEXF = 0.0
IF ((TEMP .NE. 0.0) .AND. ((TEMP1+LOG(TEMP)) .LT.500.0)) THEN
    CALEXF = DEXP(TEMP1+LOG(TEMP))
END IF

C
C   VANGENUCHTEN ERFC ROUTINE
C
DOUBLE PRECISION TEMP1,TEMP2,CALEXF,EXF
IF (TEMP2 .GT. 3.5) THEN
    CALEXF = 0.0
ELSE IF (TEMP2 .LT. -3.5) THEN
    CALEXF = 2.0*DEXP(TEMP1)
ELSE
    CALEXF = EXF(TEMP1,TEMP2)
END IF

RETURN
END

C
C   FUNCTION TO CALCULATE EXP(A) ERFC(B)
C
FUNCTION EXF(A,B)

```

```

C      PURPOSE: TO CALCULATE EXP(A) ERFC(B)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      EXF=0.0
      IF((DABS(A).GT.170.).AND.(B.LE.0.)) RETURN
      IF(B.NE.0.0) GO TO 1
      EXF=DEXP(A)
      RETURN
1 C=A-B*B
      IF((DABS(C).GT.170.).AND.(B.GT.0.)) RETURN
      IF(C.LT.-170.) GO TO 4
      X=DABS(B)
      IF(X.GT.3.0) GO TO 2
      T=1./(1.+3275911*X)
      Y=T*(.2548296-T*(.2844967-T*(1.421414-T*(1.453152-1.061405*T))))
      GO TO 3
2 Y=.5641896/(X+.5/(X+1./(X+1.5/(X+2./(X+2.5/(X+1.))))))
3 EXF=Y*DEXP(C)
4 IF(B.LT.0.0) EXF=2.*DEXP(A)-EXF
      RETURN
      END

C
C      SUBROUTINE FOR CALCULATING COMPLEX ERROR FUNCTION
C
C      SUBROUTINE CEXF(A,B,Z,U,V)
C
C      COMPLEX ERFC-FUNCTION: U+IV=EXP(Z)ERFC(A+IB)
C
      IMPLICIT REAL*8 (A-H,O-Z)
      DIMENSION W(10),H(10)
      DATA W/.4622437,.2866755,.1090172,.02481052,.3243773D-2,
1.2283386D-3,.7802556D-5,.1086069D-6,.4399341D-9,.2229394D-12/,
2H/.2453407,.7374737,1.234076,1.738538,2.254974,
32.788806,3.347855,3.944764,4.603682,5.387481/

C
C      -----
C      X=DABS(A)
C      Y=DABS(B)
C      U=0.0
C      V=0.0
C      E=0.0
C      XYZ=Y*Y+Z-X*X
C      IF(DABS(XYZ).GT.170.)RETURN
C      COS=DCOS(2.*X*Y)
C      SIN=DSIN(2.*X*Y)
C      IF((X+Y).LT.6.) GO TO 2
C      T=0.0
C      DO 1 K=1,10
C      T=T+W(K)*((Y-H(K))**2+X*X)+(X/((Y+H(K))**2+X*X))
1 V=V+W(K)*((Y-H(K))/(Y-H(K))**2+X*X)+(Y+H(K))/(Y+H(K))**2+X*X)
      U=.3183099*DEXP(XYZ)*(T*COS-V*SIN)
      V=.3183099*DEXP(XYZ)*(-T*SIN-V*COS)
      IF(X.LE.0.)U=DEXP(DMIN1(Z,1.7D2))
      GO TO 8
2 IF(X.GT.2.5) GO TO 3
      T=1./(1.+3275911*X)
      U=T*(.2548296-T*(.2844967-T*(1.421414-T*(1.453152-1.061405*T))))
      GO TO 4
3 U=.5641896/(X+.5/(X+1./(X+1.5/(X+2./(X+2.5/(X+3./(X+1.))))))
4 IF(Y.LE.0.) GO TO 7
      IF(X.LE.0.) V=-.3183099*Y
      IF(X.LE.0.) GO TO 5
      U=U-.1591549*(1.D0-COS)/X
      V=V-.1591549*SIN/X
5 NT=12.+2.*Y
      DO 6 I=1,NT
      P=I
      ARG=P*Y
      F1=X*(DEXP(ARG)+DEXP(-ARG))

```

```

      F2=0.5*P*(DEXP(ARG)-DEXP(-ARG))
      EX=.6366198*DEXP(-0.25*P*P)/(4.*X*X+P*P)
      U=U-EX*(2.*X-F1*COS+F2*SIN)
6    V=V-EX*(F1*SIN+F2*COS)
      V=V*DEXP(Z-X*X)
7    U=U*DEXP(Z-X*X)
8    IF(B.LT.0.) V=-V
      IF(A.LT.0.) U=2.*EXF(Z,E)-U
      RETURN
      END
C
C      SUBROUTINE FOR CALCULATING ERFC
C
C      SUBROUTINE CALERF(ARG,RESULT,JINT)
C-----
C
C This packet evaluates erf(x), erfc(x), and exp(x*x)*erfc(x)
C for a real argument x. It contains three FUNCTION type
C subprograms: ERF, ERFC, and ERFCX (or DERF, DERFC, and DERFCX),
C and one SUBROUTINE type subprogram, CALERF. The calling
C statements for the primary entries are:
C
C      Function          Parameters for CALERF
C      call              ARG          Result          JINT
C
C      ERF(ARG)          ANY REAL ARGUMENT          ERF(ARG)          0
C      ERFC(ARG)         ABS(ARG) .LT. XBIG          ERFC(ARG)          1
C      ERFCX(ARG)        XNEG .LT. ARG .LT. XMAX      ERFCX(ARG)         2
C
C
C Author: W. J. Cody
C          Mathematics and Computer Science Division
C          Argonne National Laboratory
C          Argonne, IL 60439
C
C Latest modification: March 19, 1990
C-----
C      INTEGER I,JINT
CS     REAL
C      DOUBLE PRECISION
C      1      A,ARG,B,C,D,DEL,FOUR,HALF,P,ONE,Q,RESULT,SIXTEN,SQRPI,
C      2      TWO,THRESH,X,XBIG,XDEN,XHUGE,XINF,XMAX,XNEG,XNUM,XSMALL,
C      3      Y,YSQ,ZERO
C      DIMENSION A(5),B(4),C(9),D(8),P(6),Q(5)
C-----
C      Mathematical constants
C-----
CS     DATA FOUR,ONE,HALF,TWO,ZERO/4.0E0,1.0E0,0.5E0,2.0E0,0.0E0/,
CS     1      SQRPI/5.6418958354775628695E-1/,THRESH/0.46875E0/,
CS     2      SIXTEN/16.0E0/
C      DATA FOUR,ONE,HALF,TWO,ZERO/4.0D0,1.0D0,0.5D0,2.0D0,0.0D0/,
CS     1      SQRPI/5.6418958354775628695D-1/,THRESH/0.46875D0/,
CS     2      SIXTEN/16.0D0/
C-----
C      Machine-dependent constants
C-----
CS     DATA XINF,XNEG,XSMALL/3.40E+38,-9.382E0,5.96E-8/,
CS     1      XBIG,XHUGE,XMAX/9.194E0,2.90E3,4.79E37/
C      DATA XINF,XNEG,XSMALL/1.79D308,-26.628D0,1.11D-16/,
CS     1      XBIG,XHUGE,XMAX/26.543D0,6.71D7,2.53D307/
C-----
C      Coefficients for approximation to erf in first interval
C-----
CS     DATA A/3.16112374387056560E00,1.13864154151050156E02,
CS     1      3.77485237685302021E02,3.20937758913846947E03,
CS     2      1.85777706184603153E-1/
CS     DATA B/2.36012909523441209E01,2.44024637934444173E02,
CS     1      1.28261652607737228E03,2.84423683343917062E03/

```

```

DATA A/3.16112374387056560D00,1.13864154151050156D02,
1 3.77485237685302021D02,3.20937758913846947D03,
2 1.85777706184603153D-1/
DATA B/2.36012909523441209D01,2.44024637934444173D02,
1 1.28261652607737228D03,2.84423683343917062D03/
-----
C
C Coefficients for approximation to erfc in second interval
C
CS DATA C/5.64188496988670089E-1,8.88314979438837594E0,
CS 1 6.61191906371416295E01,2.98635138197400131E02,
CS 2 8.81952221241769090E02,1.71204761263407058E03,
CS 3 2.05107837782607147E03,1.23033935479799725E03,
CS 4 2.15311535474403846E-8/
CS DATA D/1.57449261107098347E01,1.17693950891312499E02,
CS 1 5.37181101862009858E02,1.62138957456669019E03,
CS 2 3.29079923573345963E03,4.36261909014324716E03,
CS 3 3.43936767414372164E03,1.23033935480374942E03/
DATA C/5.64188496988670089D-1,8.88314979438837594D0,
1 6.61191906371416295D01,2.98635138197400131D02,
2 8.81952221241769090D02,1.71204761263407058D03,
3 2.05107837782607147D03,1.23033935479799725D03,
4 2.15311535474403846D-8/
DATA D/1.57449261107098347D01,1.17693950891312499D02,
1 5.37181101862009858D02,1.62138957456669019D03,
2 3.29079923573345963D03,4.36261909014324716D03,
3 3.43936767414372164D03,1.23033935480374942D03/
-----
C
C Coefficients for approximation to erfc in third interval
C
CS DATA P/3.05326634961232344E-1,3.60344899949804439E-1,
CS 1 1.25781726111229246E-1,1.60837851487422766E-2,
CS 2 6.58749161529837803E-4,1.63153871373020978E-2/
CS DATA Q/2.56852019228982242E00,1.87295284992346047E00,
CS 1 5.27905102951428412E-1,6.05183413124413191E-2,
CS 2 2.33520497626869185E-3/
DATA P/3.05326634961232344D-1,3.60344899949804439D-1,
1 1.25781726111229246D-1,1.60837851487422766D-2,
2 6.58749161529837803D-4,1.63153871373020978D-2/
DATA Q/2.56852019228982242D00,1.87295284992346047D00,
1 5.27905102951428412D-1,6.05183413124413191D-2,
2 2.33520497626869185D-3/
-----
C
X = ARG
Y = ABS(X)
IF (Y .LE. THRESH) THEN
-----
C
C Evaluate erf for |X| <= 0.46875
C
YSQ = ZERO
IF (Y .GT. XSMALL) YSQ = Y * Y
XNUM = A(5)*YSQ
XDEN = YSQ
DO 20 I = 1, 3
XNUM = (XNUM + A(I)) * YSQ
XDEN = (XDEN + B(I)) * YSQ
20 CONTINUE
RESULT = X * (XNUM + A(4)) / (XDEN + B(4))
IF (JINT .NE. 0) RESULT = ONE - RESULT
IF (JINT .EQ. 2) RESULT = EXP(YSQ) * RESULT
GO TO 800
-----
C
C Evaluate erfc for 0.46875 <= |X| <= 4.0
C
ELSE IF (Y .LE. FOUR) THEN
XNUM = C(9)*Y
XDEN = Y
DO 120 I = 1, 7
XNUM = (XNUM + C(I)) * Y
XDEN = (XDEN + D(I)) * Y

```



```

120     CONTINUE
      RESULT = (XNUM + C(8)) / (XDEN + D(8))
      IF (JINT .NE. 2) THEN
         YSQ = AINT(Y*SIXTEN)/SIXTEN
         DEL = (Y-YSQ)*(Y+YSQ)
         RESULT = EXP(-YSQ*YSQ) * EXP(-DEL) * RESULT
      END IF
C-----
C Evaluate erfc for |X| > 4.0
C-----
      ELSE
         RESULT = ZERO
         IF (Y .GE. XBIG) THEN
            IF ((JINT .NE. 2) .OR. (Y .GE. XMAX)) GO TO 300
            IF (Y .GE. XHUGE) THEN
               RESULT = SQRPI / Y
               GO TO 300
            END IF
         END IF
         YSQ = ONE / (Y * Y)
         XNUM = P(6)*YSQ
         XDEN = YSQ
         DO 240 I = 1, 4
            XNUM = (XNUM + P(I)) * YSQ
            XDEN = (XDEN + Q(I)) * YSQ
240     CONTINUE
         RESULT = YSQ *(XNUM + P(5)) / (XDEN + Q(5))
         RESULT = (SQRPI - RESULT) / Y
         IF (JINT .NE. 2) THEN
            YSQ = AINT(Y*SIXTEN)/SIXTEN
            DEL = (Y-YSQ)*(Y+YSQ)
            RESULT = EXP(-YSQ*YSQ) * EXP(-DEL) * RESULT
         END IF
      END IF
C-----
C Fix up for negative argument, erf, etc.
C-----
300 IF (JINT .EQ. 0) THEN
      RESULT = (HALF - RESULT) + HALF
      IF (X .LT. ZERO) RESULT = -RESULT
   ELSE IF (JINT .EQ. 1) THEN
      IF (X .LT. ZERO) RESULT = TWO - RESULT
   ELSE
      IF (X .LT. ZERO) THEN
         IF (X .LT. XNEG) THEN
            RESULT = XINF
         ELSE
            YSQ = AINT(X*SIXTEN)/SIXTEN
            DEL = (X-YSQ)*(X+YSQ)
            Y = EXP(YSQ*YSQ) * EXP(DEL)
            RESULT = (Y+Y) - RESULT
         END IF
      END IF
   END IF
800 RETURN
C----- Last card of CALERF -----
      END

```

Sample INPUT file

```
3 SPECIES SEQUENTIAL CHO EXAMPLE PROBLEM
3 !NUMBER OF SPECIES
200.0 10.0 1.0 1.0 1.0 1.0 !PROBLEM GEOMETRY, DISTANCE:DISTANCE STEP
(X,Y,Z)
1.0 1.0 !SOURCE GEOMETRY
200.0 200.0 200.0 !SIMULATION TIME, PULSE TIME, TIME STEP
1.0 0.18 0.0 0.0 !VELOCITY, ALPHA (X,Y,Z)
2.0 1.0 1.0 !RETARDATION FACTORS
0.01 0.1 0.0 !DECAY CONSTANTS
1.0 1.0 !YIELD COEFFICIENTS
0 !BOUNDARY CONDITION TYPE
0.0 0.0 0.0 !SOURCE DECAY LAMDA
1.0 !SPECIE1
0.0 0.0 !SPECIE2
0.0 0.0 0.0 !SPECIE3
0 !SORPTION CONDITION
0.0 0.0 0.0 !INITIAL CONDITION CONCENTRATION
0.0 0.0 0.0 !INITIAL CONDITION EXPONENTIAL DECAY
```

Sample OUTPUT file

```
3 SPECIES SEQUENTIAL CHO EXAMPLE PROBLEM
TIME = 200.0000000000000
DISTANCE SPECIES001 SPECIES002 SPECIES003
0.10000E+02 0.90500E+00 0.58963E-01 0.36037E-01
0.20000E+02 0.81902E+00 0.75434E-01 0.10554E+00
0.30000E+02 0.74122E+00 0.76529E-01 0.18225E+00
0.40000E+02 0.67080E+00 0.72352E-01 0.25685E+00
0.50000E+02 0.60707E+00 0.66636E-01 0.32629E+00
0.60000E+02 0.54940E+00 0.60739E-01 0.38986E+00
0.70000E+02 0.49721E+00 0.55131E-01 0.44766E+00
0.80000E+02 0.44984E+00 0.49951E-01 0.50007E+00
0.90000E+02 0.39123E+00 0.44654E-01 0.54725E+00
0.10000E+03 0.19746E+00 0.31598E-01 0.58224E+00
0.11000E+03 0.19071E-01 0.10525E-01 0.58123E+00
0.12000E+03 0.17579E-03 0.18928E-02 0.54780E+00
0.13000E+03 0.10664E-06 0.30148E-03 0.50195E+00
0.14000E+03 0.48551E-11 0.47855E-04 0.44975E+00
0.15000E+03 0.14559E-16 0.75816E-05 0.39182E+00
0.16000E+03 0.28198E-23 0.11956E-05 0.32774E+00
0.17000E+03 0.34881E-31 0.18650E-06 0.25691E+00
0.18000E+03 0.27368E-40 0.28315E-07 0.17887E+00
0.19000E+03 0.13558E-50 0.40066E-08 0.97222E-01
0.20000E+03 0.42276E-62 0.46934E-09 0.32185E-01
```

APPENDIX J

Evaluation of Integral

The integral in equation(77) can be evaluated as:

$$I = \int_{\tau=0}^{\tau=t} f'_x d\tau = \frac{x}{\sqrt{\pi D_x}} \exp\left\{\frac{vx}{2D_x}\right\} \int_{\tau=0}^{\tau=t} \frac{\exp\left\{\frac{-v^2}{4D_x}\tau - k\tau + \frac{-x^2}{4D_x\tau}\right\}}{\tau^{3/2}} d\tau \quad (J1)$$

Applying Laplace transform to equation(J1) we get:

$$\ell[I] = \frac{x}{\sqrt{\pi D_x}} \exp\left\{\frac{vx}{2D_x}\right\} \frac{1}{s} \ell \left[\frac{\exp\left\{\frac{-v^2}{4D_x}\tau - k\tau + \frac{-x^2}{4D_x\tau}\right\}}{\tau^{3/2}} \right] \quad (J2)$$

Where, ℓ is the Laplace transform operator and 's' is the Laplace variable.

Equation(J2) can be expressed as:

$$\ell[I] = \frac{x}{\sqrt{\pi D_x}} \frac{1}{s} \exp\left\{\frac{vx}{2D_x}\right\} \ell \left[\exp\left\{-\left\langle \frac{v^2}{4D_x} + k \right\rangle \tau\right\} \frac{\exp\left\{\frac{-x^2}{4D_x\tau}\right\}}{\tau^{3/2}} \right] \quad (J3)$$

The second term within the Laplace operator can be evaluated by using Selby [40]

(See equation(82), p: 497) as:

$$\ell \left[\frac{\exp \left\{ \frac{-x^2}{4D_x \tau} \right\}}{\tau^{3/2}} \right] = \left[\frac{2\sqrt{\pi D_x}}{x} \exp \left\{ -\frac{x}{\sqrt{D_x}} \sqrt{s} \right\} \right] \quad (J4)$$

The entire expression within the Laplace operator can be evaluated by using Selby [40] (See equation(11), p: 491) and equation(J4) as:

$$\ell[I] = \frac{2}{s} \exp \left\{ \frac{vx}{2D_x} \right\} \left[\exp \left\{ -\frac{x}{\sqrt{D_x}} \sqrt{s + \left(\frac{v^2}{4D_x} + k \right)} \right\} \right] \quad (J5)$$

Inverse Laplace transform of the above equation yields[5]:

$$I = \left(\exp \left\{ \frac{x}{2\alpha_x} \left[1 - \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2} \right] \right\} \operatorname{erfc} \left\{ \frac{x - vt \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2}}{2(\alpha_x vt)^{1/2}} \right\} \right) + \left(\exp \left\{ \frac{x}{2\alpha_x} \left[1 + \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2} \right] \right\} \operatorname{erfc} \left\{ \frac{x + vt \left(1 + \frac{4k\alpha_x}{v} \right)^{1/2}}{2(\alpha_x vt)^{1/2}} \right\} \right) \quad (J6)$$

where, $\alpha_x = \frac{D_x}{v}$

APPENDIX K

Derivation of the three-dimensional solution to zero longitudinal dispersion case

The governing transport equation is:

$$\frac{\partial c}{\partial t} + v \frac{\partial c}{\partial x} - D_y \frac{\partial^2 c}{\partial y^2} - D_z \frac{\partial^2 c}{\partial z^2} = -kc \quad (\text{K1})$$

The initial and boundary conditions are:

$$\begin{aligned} c(x, y, z, 0) &= 0 \quad \forall \quad 0 < x < \infty, \quad -\infty < y < \infty, \quad -\infty < z < \infty \\ c(0, y, z, t) &= c_0 \quad -\frac{Z}{2} < z < \frac{Z}{2}, \quad -\frac{Y}{2} < y < \frac{Y}{2}, \quad \forall \quad t > 0 \\ &= 0 \quad \text{otherwise,} \quad \forall \quad t > 0 \end{aligned} \quad (\text{K2})$$

$$\lim_{y \rightarrow \pm\infty} \frac{\partial c(x, y, z, t)}{\partial y} = 0$$

$$\lim_{z \rightarrow \pm\infty} \frac{\partial c(x, y, z, t)}{\partial z} = 0$$

Applying Laplace transform to $c(x, y, z, t)$ in equation(K1) gives:

$$\frac{\partial p}{\partial x} - \alpha_y \frac{\partial^2 p}{\partial y^2} - \alpha_z \frac{\partial^2 p}{\partial z^2} = -\frac{(s+k)}{v} p \quad (\text{K3})$$

where ; $\alpha_y = \frac{D_y}{v}$ and $\alpha_z = \frac{D_z}{v}$

where; 's' is the Laplace variable and 'p' is the concentration in the Laplace domain. The boundary conditions get modified as:

$$\begin{aligned}
p(0, y, z) &= \frac{c_0}{s} & -\frac{Z}{2} < z < \frac{Z}{2}, & \quad -\frac{Y}{2} < y < \frac{Y}{2} \\
&= 0 & \text{otherwise} &
\end{aligned}$$

$$\lim_{y \rightarrow \pm\infty} \frac{\partial p(x, y, z)}{\partial y} = 0$$

$$\lim_{z \rightarrow \pm\infty} \frac{\partial p(x, y, z)}{\partial z} = 0$$
(K4)

Equation(K3) can be interpreted as a transient two dimensional diffusive reactive transport problem. Its boundary conditions given by equation(K4) represent an instantaneous pulse of a plane source. The solution to this problem without the decay term can be readily deduced from Hunt [25] by ignoring the advection term and reducing the problem to two dimensions. Thus the solution to the above problem without the reaction term is given by:

$$p'(x, y, z) = \frac{c_0}{4s} f_y(x, y) f_z(x, z)$$

$$\text{where ; } f_y(x, y) = \left[\operatorname{erf} \left\{ \frac{y + \frac{Y}{2}}{2(\alpha_y x)^{1/2}} \right\} - \operatorname{erf} \left\{ \frac{y - \frac{Y}{2}}{2(\alpha_y x)^{1/2}} \right\} \right]$$
(K5)

$$\text{and } f_z(x, z) = \left[\operatorname{erf} \left\{ \frac{z + \frac{Z}{2}}{2(\alpha_z x)^{1/2}} \right\} - \operatorname{erf} \left\{ \frac{z - \frac{Z}{2}}{2(\alpha_z x)^{1/2}} \right\} \right]$$

Now we make use of a method similar to the Danckwert's method described by Crank [15] to include the reaction term. If p' is the solution for the diffusion problem without reaction; the solution for the same problem with a first order reaction (with a rate constant $\left(\frac{s+k}{v}\right)$) for the same initial and boundary condition is given as:

$$p = p' \exp \left\{ - \left(\frac{s+k}{v} \right) x \right\} \quad (\text{K6})$$

This can be easily verified by checking if the solution p satisfies the governing equation and the initial and boundary conditions. Since p' is the solution to equation (K3) without the reaction term, it must satisfy:

$$\frac{\partial p'}{\partial x} - \alpha_y \frac{\partial^2 p'}{\partial y^2} - \alpha_z \frac{\partial^2 p'}{\partial z^2} = 0 \quad (\text{K7})$$

Also, differentiating p with respect to x , y and z to the respective orders yields:

$$\begin{aligned} \frac{dp}{dx} &= - \left(\frac{s+k}{v} \right) \exp \left\{ - \left(\frac{s+k}{v} \right) x \right\} p' + \exp \left\{ - \left(\frac{s+k}{v} \right) x \right\} \frac{dp'}{dx} \\ \frac{\partial^2 p}{\partial y^2} &= \exp \left\{ - \left(\frac{s+k}{v} \right) x \right\} \frac{\partial^2 p'}{\partial y^2} \\ \frac{\partial^2 p}{\partial z^2} &= \exp \left\{ - \left(\frac{s+k}{v} \right) x \right\} \frac{\partial^2 p'}{\partial z^2} \end{aligned} \quad (\text{K8})$$

From equations (K6), (K7) and (K8) we get equation(K3). This proves that solution p satisfies the governing differential equation. To check for the initial condition, we substitute $x = 0$ in equation(K6). When $x = 0$, the exponential term becomes unity and hence equation(K6) reduces to $p = p'$; thus the initial condition is satisfied. In order to check for the boundary condition in the y direction, we need to take the derivative of the solution p with respect to y . This is given as:

$$\frac{dp}{dy} = \exp \left\{ - \left(\frac{s+k}{v} \right) x \right\} \frac{dp'}{dy} \quad (\text{K9})$$

In the limiting case, when y approaches $\pm \infty$, the derivative of p' with respect to y becomes 0. From equation(K9) we can conclude that the derivative of p with respect

to y also becomes 0. This proves that the boundary condition in the y direction is satisfied. On similar lines, we see that the boundary condition is also satisfied in the z direction. Hence it is proved that equation(K6) is the solution for the system of equations described by (K3) and (K4).

Equation(K6) can be written as:

$$p(x, y, z) = \frac{c_o}{4s} \exp\left\{-\left(\frac{s+k}{v}\right)x\right\} f_y(x, y) f_z(x, z) \quad (\text{K10})$$

Inverse Laplace transform of equation(K10) gives the final solution as [40] (See equation(61), p: 495):

$$c(x, y, z, t) = \frac{c_o}{8} f_x^o(x, t) f_y(x, y) f_z(x, z)$$

where ; $f_x^o(x, t) = 2 \exp\left(-\frac{kx}{v}\right) u\left\{t - \frac{x}{v}\right\}$

and $u\left\{t - \frac{x}{v}\right\}$ is the step function given by;

$$u\left\{t - \frac{x}{v}\right\} = \begin{cases} 0 & \text{if } t \leq \frac{x}{v} \\ 1 & \text{if } t > \frac{x}{v} \end{cases} \quad (\text{K11})$$