ANALYTICALLY-DERIVED SENSITIVITIES IN ONE-DIMENSIONAL MODELS OF SOLUTE TRANSPORT IN POROUS MEDIA

by Debra S. Knopman

U.S. GEOLOGICAL SURVEY

Open-File Report 86-605

Reston, Virginia

1987

DEPARTMENT OF THE INTERIOR

DONALD PAUL HODEL, Secretary

U.S. GEOLOGICAL SURVEY

Dallas L. Peck, Director

For additional information write to:

Branch of Systems Analysis
U.S. Geological Survey
410 National Center
Reston, Virginia 22092

Copies of this report can be purchased from:

U.S. Geological Survey
Books and Open-File Reports
Box 25425, Federal Center, Bldg. 810
Denver, Colorado 80225

PREFACE

The material in this report was developed for the analysis of parameter sensitivities and the estimation of parameters associated with the transport of solutes in porous media [Knopman and Voss, 1987, 1988]. Sensitivities were derived analytically for a selected number of closed-form solutions to the one-dimensional advection-dispersion equation listed in van Genuchten and Alves [1982].

To facilitate an understanding of the spatial and temporal dependence of the analytical solutions on model parameters, FORTRAN programs of the solutions developed by van Genuchten and Alves [1982] were modified and extended to include the calculation of sensitivities. Output files compatible with TELLAGRAF[1] graphics software are created in these programs by separate subroutines.

Readers are requested to notify the author of any errors found in this report or in the accompanying computer programs. Correspondence should be addressed to:

Branch of Systems Analysis
U.S. Geological Survey
410 National Center
Reston, VA 22092

Copies of the computer programs on tape are available at cost of processing from:

---

[1]Use of the trade name TELLAGRAF in this report is for identification purposes only and does not constitute endorsement by the U.S. Geological Survey.

U.S. Geological Survey

WATSTORE Program Office

437 National Center

Reston, VA 22092

Telephone: 703-648-5687

CONTENTS

ANALYTICALLY-DERIVED SENSITIVITIES IN ONE-DIMENSIONAL MODELS OF SOLUTE TRANSPORT IN POROUS MEDIA

by Debra S. Knopman

ABSTRACT

Analytically-derived sensitivities are presented for parameters in one-dimensional models of solute transport in porous media. Sensitivities were derived by direct differentiation of closed form solutions for each of the models, and by a time integral method for two of the models. Models are based on the advection-dispersion equation and include adsorption and first-order chemical decay. Boundary conditions considered are: a constant step input of solute, constant flux input of solute, and exponentially decaying input of solute at the upstream boundary. A zero flux is assumed at the downstream boundary. Initial conditions include a constant and spatially varying distribution of solute. One model simulates the mixing of solute in an observation well from individual layers in a multilayer aquifer system. Computer programs produce output files compatible with graphics software in which sensitivities are plotted as a function of either time or space.

INTRODUCTION

This report presents analytically-derived parameter sensitivities associated with mathematical models of solute transport in porous media. A sensitivity is a partial derivative representing the change in solute concentration that results from a perturbation in a model parameter. The sensitivities were derived by differentiation of a selected number of closed-form solutions presented in van Genuchten and Alves [1982]. For two of the models, sensitivities also were calculated by numerical integration over the time domain.

The models listed in Table 1 are based on the one-dimensional form of the advection-dispersion equation:

$$R \frac{\partial C}{\partial t} = D \frac{\partial^2 C}{\partial x^2} - v \frac{\partial C}{\partial x} - \mu C , \qquad (1)$$

where C is the concentration of solute $[M/L^3]$,

D is the dispersion coefficient $[L^2/T]$,

v is the average linear velocity [L/T],

R is a dimensionless retardation factor [·],

μ is a first-order reaction coefficient $[T^{-1}]$,

x is the distance from the upstream boundary [L], and

t is time [T].

Model C is intended to represent one-dimensional transport of solute in a multilayer aquifer system when solute from individual layers is mixed and vertically averaged in observation wells [Knopman and Voss, 1988].

A partial differential equation, such as (1), may only be solved

Table 1.--One-dimensional models of solute transport[a]
[Modified from van Genuchten and Alves, 1982]

Model	Governing Equation	Upstream Boundary Condition[b] $\left[-D\frac{\partial C}{\partial x} + vC\right]\Big\|_{x=0}$	Initial Condition $C(x,t_0)$
A1	$R\frac{\partial C}{\partial t} = D\frac{\partial^2 C}{\partial x^2} - v\frac{\partial C}{\partial x}$	$\begin{cases} vC_a & t_0 < t \leq t_{max} \\ 0 & t > t_{max} \end{cases}$	C_o
A2	" " "	$v(C_b + C_c e^{-\lambda t})$	C_o
A3	" " "	$\begin{cases} vC_a & t_0 < t \leq t_{max} \\ 0 & t > t_{max} \end{cases}$	$C_1 + C_2 e^{-\alpha x}$
A4[b]	" " "	$C = \begin{cases} C_a & t_0 < t \leq t_{max} \\ 0 & t > t_{max} \end{cases}$	C_o
B1	$R\frac{\partial C}{\partial t} = D\frac{\partial^2 C}{\partial x^2} - v\frac{\partial C}{\partial x} - \mu C$	$\begin{cases} vC_a & t_0 < t \leq t_{max} \\ 0 & t > t_{max} \end{cases}$	C_o
B2	" " " "	$v(C_b + C_c e^{-\lambda t})$	C_o
C	[Model A1 equation and conditions in each layer y where $C_y = C$ and $\bar{C} = \sum_{y=1}^{Y} f_y C_y$]		

[a] Sensitivities are given for parameters in bold-face type.

[b] Only model A4 includes a first-type upstream boundary condition.

explicitly by imposing boundary and initial conditions. Table 1 includes models with a variety of conditions that describe the distribution of solute in space prior to the input of additional solute into the system at time t and the time rate of change of solute into and out of the system at its boundaries. The time when input of solute at the upstream boundary ceases is t_{max}. The parameter λ is defined as an upstream boundary decay factor $[T^{-1}]$, α is a spatial factor in the initial condition $[L^{-1}]$. The factors C_a, C_b, and C_c $[M/L^3]$ relate to the input source strength at the upstream boundary; the factors C_o, C_1, and C_2 $[M/L^3]$ relate to the initial condition. A semi-infinite homogeneous medium and uniform flow field are assumed.

The most complete compendium of analytical solutions to the one-dimensional advection-dispersion equation is the work of van Genuchten and Alves [1982]. FORTRAN programs are available for each of the solutions presented in their work. Sensitivities and integrands presented in this report are incorporated into modified and extended versions of the programs developed by van Genuchten and Alves [1982]. To facilitate an understanding of the spatial and temporal behavior of the sensitivities, output files compatible with TELLAGRAF graphics software are created by separate subroutines.

The material in this report was developed for application to work on the estimation of parameters associated with the advection-dispersion equation [Knopman and Voss, 1987, 1988]. This work focuses in detail on the role of sensitivities in the estimation of parameters by nonlinear regression and the implications of this connection on the design of a field sampling strategy to improve the precision of parameter estimates. Examples of sensitivities given in this report are illustrated in Knopman [1986] and Knopman and Voss [1987, 1987].

The author wishes to acknowledge the major contribution made by Clifford I. Voss, Research Hydrologist, U.S. Geological Survey, in the analysis and application of sensitivities to parameter estimation. Our joint work provided the motivation for developing the material in this report. Charles Appel, U.S. Geological Survey, assumed the unenviable task of serving as a technical reviewer, for which I am greatly indebted.

PARAMETER SENSITIVITIES

The expression for sensitivities, derivatives, and integrands are written to conform with the order of terms that arises from the application of the chain rule in differentiation. Programs listed in the Appendix are consistent with the expressions given in this section. Calculations are made in double precision arithmetic, and as a consequence, numerical error is considered to be insignificant. However, the user may want to simplify and combine terms to even further reduce the possibility of introducing round-off errors into the results.

The dispersion coefficient D has been assumed independent of velocity. In general, the effect of any dependence of D on v in $\partial C/\partial v$ would not be significant. Only at low Peclet numbers, when dispersion overwhelmingly dominates advection, will the velocity dependence of the dispersion coefficient become important in the calculation of $\partial C/\partial v$, in which case the sensitivity would be incorrect [Knopman and Voss, 1987].

For each model, the governing equation, boundary conditions, initial condition, and analytical solution is listed. The common terms listed below each analytical solution appear in the expressions of sensitivities, derivatives, and integrands. Note that the parameters v, D, and μ that appear in Table 1 are divided by the retardation factor R, thereby creating new parameters v/R, D/R, and μ/R. These parameters are identified in the sensitivity expressions by their numerator only, but the retardation factor is implicitly included in all calculations. Also note that when absolute time t is greater than the time t_{max} at which the input of solute at the upstream boundary ceases, the difference $(t-t_{max})$ should be substituted for t in the common terms appearing in expressions derived from $A(x,t-t_{max})$ and $B(x,t-t_{max})$. This substitution is built into the programs in the Appendix. The time t_{max} is also a parameter of the model that may be estimated.

Following the list of common terms, sensitivities formed by direct differentiation of the analytical solution are presented for each parameter in the model as noted in Table 1. For model A1, the sensitivity to velocity and sensitivity to the dispersion coefficient are also given in terms of a time integral. For model A4, the sensitivity to velocity is given in terms of both direct differentiation and a time integral.

Model A1

Governing equation:

$$R\,\frac{\partial C}{\partial t} = D\,\frac{\partial^2 C}{\partial x^2} - v\,\frac{\partial C}{\partial x}$$

Boundary conditions:

$$\left[-D\,\frac{\partial C}{\partial x} + vC\right]\Big|_{x=0} = \begin{cases} vC_a & t_o < t \leq t_{max} \\ 0 & t > t_{max} \end{cases}$$

$$\frac{\partial C}{\partial x}(\infty, t) = 0$$

Initial condition:

$$C(x, t_o) = C_o$$

Analytical solution:

$$C(x,t) = \begin{cases} C_o + (C_a - C_o)\;A(x,t) & t_o < t \leq t_{max} \\ C_o + (C_a - C_o)\;A(x,t) - C_a\;A(x, t - t_{max}) & t > t_{max}\,, \end{cases}$$

where

$$A(x,t) = \frac{1}{2}\,\mathrm{erfc}\left[\frac{Rx - vt}{2(DRt)^{1/2}}\right] + \left[\frac{v^2 t}{\pi DR}\right]^{1/2} \exp\left[-\,\frac{(Rx - vt)^2}{4DRt}\right]$$

$$-\;\frac{1}{2}\left[1 + \frac{vx}{D} + \frac{v^2 t}{DR}\right] \exp(vx/D)\;\;\mathrm{erfc}\left[\frac{Rx + vt}{2(DRt)^{1/2}}\right]$$

Common terms:

$$v = v/R$$

$$D = D/R$$

$$Q = \frac{vx}{D}$$

$$CM = \frac{x - vt}{2(Dt)^{1/2}}$$

$$CP = \frac{x + vt}{2(Dt)^{1/2}}$$

$$B1 = 1 + Q + \frac{v^2 t}{D}$$

$$B2 = \exp\left[-(CM)^2\right]$$

$$B3 = \frac{1}{2(Dt)^{1/2}}$$

$$B4 = \exp(Q)$$

$$B5 = v/D$$

$$B6 = \frac{x - vt}{2\ Dt}$$

$$B7 = \left[\frac{v^2 t}{\pi D}\right]^{1/2}$$

$$B8 = \exp(Q)\ \mathrm{erfc}\ (CP)$$

$$B9 = \exp\left[-(CP)^2\right]$$

$$B10 = x/D$$

$$B11 = \frac{x + vt}{2\ Dt}$$

Sensitivities By Direct Differentiation

$$\frac{\partial C(x,t)}{\partial v} = \begin{cases} (C_a - C_o)\dfrac{\partial A(x,t)}{\partial v} & t_o < t \leq t_{max} \\ (C_a - C_o)\dfrac{\partial A(x,t)}{\partial v} - C_a \dfrac{\partial A(x,t-t_{max})}{\partial v} & t > t_{max}, \end{cases}$$

where

$$\frac{\partial A(x,t)}{\partial v} = \frac{t}{\sqrt{\pi}} B2\ B3 + \frac{t}{\pi} B5\ B2/B7 + t\ B7\ B6\ B2$$

$$- \frac{1}{2}(B10 + 2t\ B5)\ B8 - \frac{1}{2} B1\ B10\ B8$$

$$+ \frac{t}{\sqrt{\pi}} B1\ B4\ B3\ B9$$

$$\frac{\partial C(x,t)}{\partial D} = \begin{cases} (C_a - C_o)\dfrac{\partial A(x,t)}{\partial D} & t_o < t \leq t_{max} \\ (C_a - C_o)\dfrac{\partial A(x,t)}{\partial D} - C_a \dfrac{\partial A(x,t-t_{max})}{\partial D} & t > t_{max}, \end{cases}$$

where

$$\frac{\partial A(x,t)}{\partial D} = \frac{t}{\sqrt{\pi}} B2\ B6\ B3 - \frac{t}{2\pi}(B5)^2\ B2/B7$$

$$+ t\ B7\ (B6)^2\ B2 + \frac{1}{2}(B5\ B10 + (B5)^2\ t)\ B8$$

$$+ \frac{Q}{2D} B1\ B8 - \frac{1}{2D\sqrt{\pi}} B1\ B4\ B9\ CP$$

$$\frac{\partial C(x,t)}{\partial C_a} = \begin{cases} A(x,t) & t_o < t \leq t_{max} \\ A(x,t) - A(x,t-t_{max}) & t > t_{max} \end{cases}$$

$$\frac{\partial C(x,t)}{\partial C_o} = 1 - A(x,t) \qquad t > t_{max}$$

$$\frac{\partial C(x,t)}{\partial t_{max}} = - C_a \frac{\partial A(x,t-t_{max})}{\partial t_{max}} \qquad t > t_{max} \; ,$$

where

$$\frac{\partial A(x,t-t_{max})}{\partial t_{max}} = - \frac{1}{\sqrt{\pi}} \; B2 \; (v \; B3 + D \; B6 \; B3) - \frac{v}{2\pi} \; B5 \; B2/B7$$

$$- B7 \; B2 \; [v \; B6 + D(B6)^2] + \frac{v}{2} \; B5 \; B8$$

$$+ \frac{1}{\sqrt{\pi}} \; B1 \; B4 \; B9 \; (-v \; B3 + D \; B11 \; B3)$$

Sensitivities By Integration

$$\frac{\partial C(x,t)}{\partial v} = \int_{t^*}^{t} D \frac{\partial}{\partial v} \left[\frac{\partial^2 C}{\partial x^2}\right] dt - \int_{t^*}^{t} \left[\frac{\partial C}{\partial x}\right] dt - v \int_{t^*}^{t} \frac{\partial}{\partial v} \left[\frac{\partial C}{\partial x}\right] dt \; ,$$

where

$$t^* = \begin{cases} t_o & \text{when} \quad t_o < t \leq t_{max} \\ t_{max} & \text{when} \quad t > t_{max} \end{cases}$$

$$\frac{\partial^n C(x,t)}{\partial x^n} = \begin{cases} (C_a - C_o)\dfrac{\partial^n A(x,t)}{\partial x^n} & t_o < t \le t_{max} \\ (C_a - C_o)\dfrac{\partial^n A(x,t)}{\partial x^n} - C_a \dfrac{\partial^n A(x,t-t_{max})}{\partial x^n} & t > t_{max}\,, \end{cases} \quad n=1,2$$

where

$$\frac{\partial}{\partial v}\left(\frac{\partial^2 A(x,t)}{\partial x^2}\right) = \frac{t}{\sqrt{\pi}}\,B3\ B2\ (B6)^2 - \frac{1}{2D\sqrt{\pi}}\,B3\ B2 - \frac{1}{2D\pi}\,B5\ B2/B7$$

$$- \frac{1}{2D}\,B7\ B2\ B6 + \frac{t}{\pi}\,B5\ (B6)^2\ B2/B7 - \frac{1}{D}\,B7\ B6\ B2$$

$$+ t\ B7\ (B6)^2\ B2\ B6 - \frac{1}{D}\,B5\ B8 - \frac{1}{2}\,(B5)^2\ B8\ B10$$

$$+ \frac{t}{\sqrt{\pi}}\,(B5)^2\ B4\ B9\ B3 + \frac{1}{D\sqrt{\pi}}\,B4\ B9\ B3$$

$$+ \frac{1}{\sqrt{\pi}}\,B5\ B4\ B10\ B9\ B3 - \frac{t}{\sqrt{\pi}}\,B5\ B4\ B9\ B11\ B3$$

$$- \frac{1}{2}\,(B10 + 2t\ B5)\ (B5)^2\ B8 - \frac{1}{D}\,B1\ B5\ B8$$

$$- \frac{1}{2}\,B1\ (B5)^2\ B8\ B10 + \frac{t}{\sqrt{\pi}}\,B1\ (B5)^2\ B4\ B3\ B9$$

$$- \frac{1}{D}\,B5\ B8 - \frac{1}{2}\,(B5)^2\ B8\ B10 + \frac{t}{\sqrt{\pi}}\,(B5)^2\ B4\ B9\ B3$$

$$+ \frac{1}{\sqrt{\pi}}\,(B10 + 2t\ B5)\ B4\ B9\ B3\ B5$$

$$+ \frac{1}{\sqrt{\pi}}\,B1\ B10\ B4\ B9\ B3\ B5$$

$$- \frac{t}{\sqrt{\pi}}\,B1\ B4\ B9\ B11\ B3\ B5 + \frac{1}{D\sqrt{\pi}}\,B1\ B4\ B9\ B3$$

$$+ \frac{1}{D\sqrt{\pi}}\,B4\ B9\ B3 + \frac{1}{\sqrt{\pi}}\,B5\ B10\ B4\ B9\ B3$$

$$- \frac{t}{\sqrt{\pi}}\,B5\ B4\ B9\ B11\ B3$$

$$+ \frac{1}{\sqrt{\pi}} (B10 + 2t\,B5)\ B5\ B4\ B9\ B3$$

$$+ \frac{1}{D\sqrt{\pi}} B1\ B4\ B9\ B3 + \frac{1}{\sqrt{\pi}} B1\ B5\ B4\ B9\ B3\ B10$$

$$- \frac{t}{\sqrt{\pi}} B1\ B5\ B4\ B9\ B11\ B3$$

$$- \frac{1}{\sqrt{\pi}} (B10 + 2t\ B5)\ B4\ B9\ B11\ B3$$

$$- \frac{1}{\sqrt{\pi}} B1\ B10\ B4\ B9\ B11\ B3 + \frac{t}{\sqrt{\pi}} B1\ B4\ B9\ (B11)^2\ B3$$

$$- \frac{1}{2D\sqrt{\pi}} B1\ B4\ B9\ B3$$

$$\frac{\partial^2 A\ (x,t)}{\partial x^2} = \frac{1}{\sqrt{\pi}} B2\ B6\ B3 - \frac{1}{2Dt} B7\ B2 + B7\ (B6)^2\ B2 - \frac{1}{2}(B5)^2\ B8$$

$$+ \frac{1}{\sqrt{\pi}} B5\ B4\ B9\ B3 - \frac{1}{2} (B5)^2\ B8 - \frac{1}{2} B1\ (B5)^2\ B8$$

$$+ \frac{2}{\sqrt{\pi}} B1\ B5\ B4\ B3\ B9 + \frac{1}{\sqrt{\pi}} B5\ B4\ B9\ B3$$

$$- \frac{1}{\sqrt{\pi}} B1\ B4\ B11\ B3\ B9$$

$$\frac{\partial A\ (x,t)}{\partial x} = - \frac{1}{\sqrt{\pi}} B2\ B3 - B7\ B6\ B2 - \frac{1}{2} B1\ B5\ B8 - \frac{1}{2} B5\ B8$$

$$+ \frac{1}{\sqrt{\pi}} B1\ B4\ B9\ B3$$

$$\frac{\partial}{\partial v}\left(\frac{\partial A(x,t)}{\partial x}\right) = - \frac{t}{\sqrt{\pi}} B2\ B6\ B3 - \frac{t}{\pi} B5\ B2\ B6/B7 - t\ B7\ B2\ (B6)^2$$

$$+ \frac{1}{2D} B7\ B2 - \frac{1}{2D} B8 - \frac{1}{2} B5\ B10\ B8$$

$$+ \frac{t}{\sqrt{\pi}} B5\ B4\ B9\ B3 - \frac{1}{2} (B10 + 2t\ B5)\ B5\ B8$$

$$- \frac{1}{2D} B1\ B8 - \frac{1}{2} B1\ B5\ B10\ B8 + \frac{t}{\sqrt{\pi}} B1\ B5\ B4\ B9\ B3$$

$$+\frac{1}{\sqrt{\pi}} (B10 + 2t\ B5)\ B4\ B9\ B3 + \frac{1}{\sqrt{\pi}} B1\ B10\ B4\ B9\ B3$$

$$-\frac{t}{\sqrt{\pi}} B1\ B4\ B9\ B11\ B3$$

$$\frac{\partial C(x,t)}{\partial D} = \int_{t^*}^{t} \left[\frac{\partial^2 C}{\partial x^2}\right] dt + \int_{t^*}^{t} D \frac{\partial}{\partial D} \left[\frac{\partial^2 C}{\partial x^2}\right] dt - \int_{t^*}^{t} v \frac{\partial}{\partial D} \left[\frac{\partial C}{\partial x}\right] dt ,$$

where t^* and $\frac{\partial^n C(x,t)}{\partial x^n}$ are defined as above and

$$\frac{\partial}{\partial D} \left(\frac{\partial A(x,t)}{\partial x}\right) = -\frac{1}{D\sqrt{\pi}} B2\ (CM)^2\ B3 + \frac{1}{2D\sqrt{\pi}} B2\ B3 + \frac{1}{D} B7\ B6\ B2$$

$$+ \frac{1}{2D} B7\ B6\ B2 - \frac{1}{D} B7\ B6\ B2\ (CM)^2$$

$$+ \frac{1}{2D} B5\ B8 + \frac{1}{2} (B5)^2\ B10\ B8$$

$$- \frac{1}{2D\sqrt{\pi}} B5\ B4\ B9\ CP + \frac{1}{2} \left(\frac{Q}{D} + t\ (B5)^2\right) B5\ B8$$

$$+ \frac{1}{2D} B1\ B5\ B8 + \frac{Q}{2D} B1\ B5\ B8$$

$$- \frac{1}{2D\sqrt{\pi}} B1\ B5\ B4\ B9\ CP$$

$$-\frac{1}{\sqrt{\pi}} \left(\frac{Q}{D} + t\ (B5)\right) B9\ B3\ B4$$

$$- \frac{Q}{D\sqrt{\pi}} \quad B1\ B4\ B9\ B3$$

$$+ \frac{1}{D\sqrt{\pi}} \quad B1\ B4\ B9\ (CP)^2\ B3$$

$$- \frac{1}{2D\sqrt{\pi}} \quad B1\ B4\ B9\ B3$$

$\frac{\partial}{\partial D}\left[\frac{\partial^2 A(x,t)}{\partial x^2}\right]$ is solved in the program for model A1 by substitution of other calculated terms.

Model A2

Governing equation;

$$R\frac{\partial C}{\partial t} = D\frac{\partial^2 C}{\partial x^2} - v\frac{\partial C}{\partial x}$$

Boundary conditions:

$$\left(-D\frac{\partial C}{\partial x} + vC\right)\Bigg|_{x=0} = v(C_b + C_c e^{-\lambda t}) \qquad t > t_o$$

$$\frac{\partial C}{\partial x}(\infty, t) = 0$$

Initial condition:

$$C(x, t_o) = C_o$$

Analytical solution:

$$C(x,t) = C_o + (C_b - C_o)\,A(x,t) + C_c B(x,t) \qquad t > t_o\,,$$

where

$$A(x,t) = \frac{1}{2}\,\mathrm{erfc}\left[\frac{Rx - vt}{2(DRt)^{1/2}}\right] + \left(\frac{v^2 t}{\pi DR}\right)^{1/2} \exp\left[-\frac{(Rx - vt)^2}{4DRt}\right]$$

$$- \frac{1}{2}\left(1 + \frac{vx}{D} + \frac{v^2 t}{DR}\right) \exp(vx/D)\ \mathrm{erfc}\left[\frac{Rx + vt}{2(DRt)^{1/2}}\right]$$

$$B(x,t) = e^{-\lambda t}\left\{\frac{v}{(v+y)} \exp\left[\frac{(v-y)x}{2D}\right] \mathrm{erfc}\left[\frac{Rx - yt}{2(DRt)^{1/2}}\right]\right.$$

$$\left. + \frac{v}{(v-y)} \exp\left[\frac{(v+y)x}{2D}\right] \mathrm{erfc}\left[\frac{Rx + yt}{2(DRt)^{1/2}}\right]\right\}$$

$$- \frac{v^2}{2\lambda DR} \exp(vx/D) \operatorname{erfc}\left[\frac{Rx + vt}{2(DRt)^{1/2}}\right]$$

and

$$y = v\left[1 - \frac{4\lambda DR}{v^2}\right]^{1/2}$$

Common terms:

$$v = v/R$$

$$D = D/R$$

$$P = \frac{vx}{D}$$

$$AM = \frac{x - vt}{2(Dt)^{1/2}}$$

$$AP = \frac{x + vt}{2(Dt)^{1/2}}$$

$$B1 = 1 + P + \frac{v^2 t}{D}$$

$$B2 = \exp\left[- (AM)^2 \right]$$

$$B3 = \frac{1}{2(Dt)^{1/2}}$$

$$B4 = \exp(P)$$

$$B5 = v/D$$

$$B6 = \frac{x - vt}{2\ Dt}$$

$$B7 = \left(\frac{v^2 t}{\pi D} \right)^{1/2}$$

$$B8 = \exp(P)\ \mathrm{erfc}(AP)$$

$$B9 = \exp\left[- (AP)^2 \right]$$

$$B10 = x/D$$

$$B11 = \frac{x + vt}{2\ Dt}$$

$$\frac{\partial y}{\partial v} = \frac{y}{v} + \frac{4\lambda D}{yv}$$

$$\frac{\partial y}{\partial D} = - \frac{2\lambda}{y}$$

$$\frac{\partial y}{\partial \lambda} = - \frac{2D}{y}$$

$$BM = \frac{x - yt}{2(Dt)^{1/2}}$$

$$BP = \frac{x + yt}{2(Dt)^{1/2}}$$

$$CM = \frac{(v - y)x}{2D}$$

$$CP = \frac{(v + y)x}{2D}$$

$$E1 = \exp(CM)\mathrm{erfc}(BM)$$

$$E2 = \exp(CM)$$

$$E3 = \exp[-(BM)^2]$$

$$E4 = \exp(CP)\mathrm{erfc}(BP)$$

$$E5 = \frac{1}{v - y}$$

$$E6 = \exp[-(BP)^2]$$

$$E7 = \frac{v^2}{2\lambda D}$$

$$E8 = \frac{1}{v + y}$$

$$E9 = \exp[-\lambda t]$$

$$E10 = \exp(CP)$$

$$\frac{\partial C(x,t)}{\partial v} = (C_b - C_o)\frac{\partial A(x,t)}{\partial v} + C_c\frac{\partial B(x,t)}{\partial v} ,$$

where

$$\frac{\partial A(x,t)}{\partial v} = \frac{t}{\sqrt{\pi}} B2\ B3 + \frac{t}{\pi} B5\ B2/B7 + t\ B7\ B6\ B2$$
$$- \frac{1}{2}(B10 + 2t\ B5)\ B8 - \frac{1}{2} B1\ B10\ B8$$
$$+ \frac{t}{\sqrt{\pi}} B1\ B4\ B3\ B9$$

$$\frac{\partial B(x,t)}{\partial v} = e^{-\lambda t}\left\{\left[E8 - v(E8)^2\left(1 + \frac{\partial y}{\partial v}\right)\right] E1\right.$$
$$+ \frac{v}{2} E8\left(1 - \frac{\partial y}{\partial v}\right) B10\ E1$$
$$+ \frac{2vt}{\sqrt{\pi}} E8\ E2\ E3\frac{\partial y}{\partial v} B3$$
$$+ \left[E5 - v(E5)^2\left(1 - \frac{\partial y}{\partial v}\right)\right] E4$$
$$+ \frac{v}{2} E5\left(1 + \frac{\partial y}{\partial v}\right) B10\ E4$$
$$\left. - \frac{2vt}{\sqrt{\pi}} E5\ E10\ E6\frac{\partial y}{\partial v} B3\right\}$$
$$- \frac{v}{\lambda D} B8 - E7\ B10\ B8 + \frac{2t}{\sqrt{\pi}} E7\ B4\ B9\ B3$$

$$\frac{\partial C(x,t)}{\partial D} = (C_b - C_o)\frac{\partial A(x,t)}{\partial D} + C_c \frac{\partial B(x,t)}{\partial D},$$

where

$$\frac{\partial A(x,t)}{\partial D} = \frac{t}{\sqrt{\pi}} \text{ B2 B6 B3} - \frac{t}{2\pi} (\text{B5})^2 \text{ B2/B7}$$

$$+ \text{ t B7 } (\text{B6})^2 \text{ B2} + \frac{1}{2}\left[\text{ B5 B10} + (\text{B5})^2 t \right] \text{B8}$$

$$+ \frac{P}{2D} \text{ B1 B8} - \frac{1}{2D\sqrt{\pi}} \text{ B1 B4 B9 AP}$$

$$\frac{\partial B(x,t)}{\partial D} = ve^{-\lambda t}\left[-(\text{E8})^2 \frac{\partial y}{\partial D} \text{ E1} - \text{E8} \left(\frac{1}{2}\frac{\partial y}{\partial D} \text{ B10} + \frac{CM}{D}\right) \text{E1}\right.$$

$$+ \frac{2}{\sqrt{\pi}} \text{ E8 E2 E3} \left(\frac{BM}{2D} + \text{t B3} \frac{\partial y}{\partial D}\right)$$

$$+ (\text{E5})^2 \frac{\partial y}{\partial D} \text{ E4} + \text{E5} \left(\frac{CP}{D} - \frac{1}{2} \text{B10} \frac{\partial y}{\partial D}\right) \text{E4}$$

$$\left. + \frac{2}{\sqrt{\pi}} \text{ E5 E10 E6} \left(\frac{BP}{2D} - \text{t B3} \frac{\partial y}{\partial D}\right)\right]$$

$$+ \frac{1}{D} \text{ E7} \left[\text{ B8} + \text{P B8} - \frac{1}{\sqrt{\pi}} \text{ B4 B9 AP} \right]$$

$$\frac{\partial C(x,t)}{\partial \lambda} = C_c \frac{\partial B(x,t)}{\partial \lambda},$$

where

$$\frac{\partial B(x,t)}{\partial \lambda} = ve^{-\lambda t}\left\{ -t(\text{E8 E1} + \text{E5 E4}) - \frac{\partial y}{\partial \lambda}\left[(\text{E8})^2 \text{ E1}\right.\right.$$

$$+ \frac{1}{2} \text{ E8 B10 E1} - \frac{2t}{\sqrt{\pi}} \text{ E8 E2 E3 B3}$$

$$- (E5)^2 \; E4 - \frac{1}{2} \; E5 \; B10 \; E4 + \frac{2t}{\sqrt{\pi}} \; E5 \; E10 \; E6 \; B3 \; \Big] \Big\} + \frac{1}{\lambda} \; E7 \; B8$$

$$\frac{\partial C(x,t)}{\partial C_b} = A(x,t)$$

$$\frac{\partial C(x,t)}{\partial C_c} = B(x,t)$$

$$\frac{\partial C(x,t)}{\partial C_o} = 1 - A(x,t)$$

Model A3

Governing equation:

$$R\frac{\partial C}{\partial t} = D\frac{\partial^2 C}{\partial x^2} - v\frac{\partial C}{\partial x}$$

Boundary conditions:

$$\left[-D\frac{\partial C}{\partial x} + vC\right]\Big|_{x=0} \quad \begin{cases} vC_a & t_o < t \leq t_{max} \\ 0 & t > t_{max} \end{cases}$$

$$\frac{\partial C}{\partial x}(\infty, t) = 0$$

Initial condition:

$$C(x,t_o) = C_1 + C_2 e^{-\alpha x}$$

Analytical solution:

$$C(x,t) = \begin{cases} C_1 + (C_a - C_1)\ A(x,t) + C_2\ B(x,t) & t_o < t \leq t_{max} \\ C_1 + (C_a - C_1)\ A(x,t) + C_2\ B(x,t) - C_a\ A(x,t-t_{max}) & t > t_{max} \end{cases}$$

where

$$A(x,t) = \frac{1}{2}\operatorname{erfc}\left[\frac{Rx-vt}{2(DRt)^{1/2}}\right] + \left[\frac{v^2 t}{\pi DR}\right]^{1/2}\exp\left[-\frac{(Rx-vt)^2}{4DRt}\right]$$

$$-\frac{1}{2}\left[1 + \frac{vx}{D} + \frac{v^2 t}{DR}\right]\exp(vx/D)\ \operatorname{erfc}\left[\frac{Rx+vt}{2(DRt)^{1/2}}\right]$$

$$B(x,t) = \exp\left[\frac{\alpha^2 Dt}{R} + \frac{\alpha vt}{R} - \alpha x\right]\left\{1 - \frac{1}{2}\operatorname{erfc}\left[\frac{Rx + (v+2\alpha D)t}{2(DRt)^{1/2}}\right]\right]$$

$$- \frac{1}{2}\left[1 + \frac{v}{\alpha D}\right] \exp\left[\frac{vx}{D} + 2\alpha x\right] \operatorname{erfc}\left[\frac{Rx + (v + 2\alpha D)t}{2(DRt)^{1/2}}\right] \Bigg\}$$

$$- \frac{v}{2\alpha D} \exp(vx/D) \operatorname{erfc}\left[\frac{Rx + vt}{2(DRt)^{1/2}}\right]$$

Common terms:

$$v = v/R$$

$$D = D/R$$

$$P = \frac{vx}{D}$$

$$AM = \frac{x - vt}{2(Dt)^{1/2}}$$

$$AP = \frac{x + vt}{2(Dt)^{1/2}}$$

$$B1 = 1 + P + \frac{v^2 t}{D}$$

$$B2 = \exp\left[-(AM)^2\right]$$

$$B3 = \frac{1}{2(Dt)^{1/2}}$$

$$B4 = \exp(P)$$

$$B5 = v/D$$

$$B6 = \frac{x - vt}{2\ Dt}$$

$$B7 = \left(\frac{v^2 t}{\pi D}\right)^{1/2}$$

$$B8 = \exp(P)\ \mathrm{erfc}\ (AP)$$

$$B9 = \exp\left[-(AP)^2\right]$$

$$B10 = x/D$$

$$B11 = \frac{x + vt}{2\ Dt}$$

$$BM = \frac{x - (v + 2\alpha D)t}{2(Dt)^{1/2}}$$

$$BP = \frac{x + (v + 2\alpha D)t}{2(Dt)^{1/2}}$$

$$D1 = \exp\left[\alpha\ (\alpha Dt + vt - x)\right]$$

$$D2 = 1 + \frac{v}{\alpha D}$$

$$D3 = \exp\left[P + 2\alpha x\right]$$

$$D4 = \frac{1}{2\alpha D}$$

$$D5 = \frac{1}{2}\left[2 - \mathrm{erfc}(BM) + D2\ D3\ \mathrm{erfc}\ (BP)\right]$$

$$D6 = \exp\left[-(BM)^2\right]$$

$$D7 = \exp\ (P + 2\alpha x)\ \mathrm{erfc}(BP)$$

$$D8 = \exp\left[-(BP)^2\right]$$

$$D9 = \frac{v}{2\alpha D}$$

$$\frac{\partial C(x,t)}{\partial v} = \begin{cases} (C_a - C_1)\dfrac{\partial A(x,t)}{\partial v} + C_2 \dfrac{\partial B(x,t)}{\partial v} & t_o < t \leq t_{max} \\ (C_a - C_1)\dfrac{\partial A(x,t)}{\partial v} + C_2 \dfrac{\partial B(x,t)}{\partial v} - C_a \dfrac{\partial A(x,t-t_{max})}{\partial v} & t > t_{max} \end{cases}$$

where

$$\frac{\partial A(x,t)}{\partial v} = \frac{t}{\sqrt{\pi}} \cdot B2 \cdot B3 + \frac{t}{\pi} \cdot B5 \cdot B2/B7 + t \cdot B7 \cdot B6 \cdot B2$$
$$- \frac{1}{2}(B10 + 2t \cdot B5) \cdot B8 - \frac{1}{2} \cdot B1 \cdot B10 \cdot B8$$
$$+ \frac{t}{\sqrt{\pi}} \cdot B1 \cdot B4 \cdot B3 \cdot B9$$

$$\frac{\partial B(x,t)}{\partial v} = \alpha t \; D1 \; D5 - \frac{t}{\sqrt{\pi}} D1 \; D6 \; B3 + D1 \; D4 \; D7$$
$$+ \frac{1}{2} D1 \; D2 \; B10 \; D7 - \frac{t}{\sqrt{\pi}} D1 \; D2 \; D3 \; D8 \; B3$$
$$- D4 \; B8 - D9 \; B10 \; B8 + \frac{2t}{\sqrt{\pi}} D9 \; B4 \; B9 \; B3$$

$$\frac{\partial C(x,t)}{\partial D} = \begin{cases} (C_a - C_1)\dfrac{\partial A(x,t)}{\partial D} + C_2 \dfrac{\partial B(x,t)}{\partial D} & t_o < t \leq t_{max} \\ (C_a - C_1)\dfrac{\partial A(x,t)}{\partial D} + C_2 \dfrac{\partial B(x,t)}{\partial D} - C_a \dfrac{\partial A(x,t-t_{max})}{\partial D} & t > t_{max} \end{cases}$$

where

$$\frac{\partial A(x,t)}{\partial D} = \frac{t}{\sqrt{\pi}} \; B2 \; B6 \; B3 - \frac{t}{2\pi} \; (B5)^2 \; B2/B7$$
$$+ t \; B7 \; (B6)^2 \; B2 + \frac{1}{2}\left[B5 \; B10 + (B5)^2 \; t\right] B8$$

$$+ \frac{P}{2D} \quad B1\ B8 - \frac{1}{2D\sqrt{\pi}} \quad B1\ B4\ B9\ AP$$

$$\frac{\partial B(x,t)}{\partial D} = \alpha^2 t\ D1\ D5 - \frac{1}{2\sqrt{\pi}} D1\ D6\ (B3\ B10 - t\ B3\ B5 + 2\alpha t\ B3)$$

$$- \frac{1}{D} D1\ D2\ D7 - \frac{P}{2D} D1\ D2\ D7$$

$$+ \frac{1}{2\sqrt{\pi}} D1\ D2\ D3\ D8\ (B10\ B3 + t\ B3\ B5 - 2\alpha t\ B3)$$

$$+ \frac{1}{D} D9\ B8\ (1 + P) - \frac{1}{D\sqrt{\pi}} D9\ B4\ B9\ AP$$

$$\frac{\partial C(x,t)}{\partial \alpha} = C_2 \frac{\partial B(x,t)}{\partial \alpha} ,$$

where

$$\frac{\partial B(x,t)}{\partial \alpha} = (2\alpha Dt + vt - x)\ D1\ D5 - \frac{2Dt}{\sqrt{\pi}} D1\ D6\ B3$$

$$- \frac{1}{\alpha} D1\ D9\ D7 + x\ D1\ D2\ D7$$

$$- \frac{2Dt}{\sqrt{\pi}} D1\ D2\ D3\ D8\ B3 + \frac{1}{\alpha} D9\ B8$$

$$\frac{\partial C(x,t)}{\partial C_a} = \begin{cases} A(x,t) & t_o < t \leq t_{max} \\ A(x,t) - A(x,t-t_{max}) & t > t_{max} \end{cases}$$

$$\frac{\partial C(x,t)}{\partial C_1} = 1 - A(x,t)$$

$$\frac{\partial C(x,t)}{\partial C_2} = B(x,t)$$

$$\frac{\partial C(x,t)}{\partial t_{max}} = - C_a \frac{\partial A(x,t-t_{max})}{\partial t_{max}} \qquad t > t_{max} ,$$

where

$$\frac{\partial \Lambda(x,t-t_{max})}{\partial t_{max}} = -\frac{1}{\sqrt{\pi}} \ B2 \ (v \ B3 + D \ B6 \ B3) - \frac{v}{2\pi} \ B5 \ B2/B7$$

$$- B7 \ B2 \ [v \ B6 + D(B6)^2] + \frac{v}{2} \ B5 \ B8$$

$$+ \frac{1}{\sqrt{\pi}} \ B1 \ B4 \ B9 \ (-v \ B3 + D \ B11 \ B3)$$

Model A4

Governing equation:

$$R\frac{\partial C}{\partial t} = D\frac{\partial^2 C}{\partial x^2} - v\frac{\partial C}{\partial x}$$

Boundary conditions:

$$C(0,t) = \begin{cases} C_a & t_o < t \leq t_{max} \\ 0 & t > t_{max} \end{cases}$$

$$\frac{\partial C}{\partial x}(\infty,t) = 0$$

Initial condition:

$$C(x,t_o) = C_o$$

Analytical solution:

$$C(x,t) = \begin{cases} C_o + (C_a - C_o)\ A(x,t) & t_o < t \leq t_{max} \\ C_o + (C_a - C_o)\ A(x,t) - C_a\ A(x,t-t_{max}) & t > t_{max}, \end{cases}$$

where

$$A(x,t) = \frac{1}{2}\,\mathrm{erfc}\left[\frac{Rx - vt}{2(DRt)^{1/2}}\right] + \frac{1}{2}\exp\left[\frac{vx}{D}\right]\mathrm{erfc}\left[\frac{Rx + vt}{2(DRt)^{1/2}}\right]$$

Common terms:

$$v = v/R$$

$$D = D/R$$

$$Q = \frac{vx}{D}$$

$$CM = \frac{x - vt}{2(Dt)^{1/2}}$$

$$CP = \frac{x + vt}{2(Dt)^{1/2}}$$

$$B1 = 1 + Q + \frac{v^2 t}{D}$$

$$B2 = \exp\left[- (CM)^2 \right]$$

$$B3 = \frac{1}{2(Dt)^{1/2}}$$

$$B4 = \exp(Q)$$

$$B5 = v/D$$

$$B6 = \frac{x - vt}{2\ Dt}$$

$$B7 = \left[\frac{v^2 t}{\pi D} \right]^{1/2}$$

$$B8 = \exp(Q)\ \mathrm{erfc}(CP)$$

$$B9 = \exp\left[- (CP)^2 \right]$$

$$B10 = x/D$$

$$B11 = \frac{x + vt}{2\ Dt}$$

$$\frac{\partial C(x,t)}{\partial v} = \begin{cases} (C_a - C_o)\,\dfrac{\partial A(x,t)}{\partial v} & t_o < t \leq t_{max} \\[2ex] (C_a - C_o)\,\dfrac{\partial A(x,t)}{\partial v} - C_a\,\dfrac{\partial A(x,t-t_{max})}{\partial v} & t > t_{max}\,, \end{cases}$$

where

$$\frac{\partial A(x,t)}{\partial v} = \frac{t}{\sqrt{\pi}}\text{ B2 B3} + \frac{1}{2}\text{ B10 B8} - \frac{t}{\sqrt{\pi}}\text{ B4 B9 B3}$$

$$\frac{\partial C(x,t)}{\partial D} = \begin{cases} (C_a - C_o)\,\dfrac{\partial A(x,t)}{\partial D} & t_o < t \leq t_{max} \\[2ex] (C_a - C_o)\,\dfrac{\partial A(x,t)}{\partial D} - C_a\,\dfrac{\partial A(x,t-t_{max})}{\partial D} & t > t_{max}\,, \end{cases}$$

where

$$\frac{\partial A(x,t)}{\partial D} = \frac{1}{2D\sqrt{\pi}}\text{ B2 CM} - \frac{Q}{2D}\text{ B8} + \frac{1}{2D\sqrt{\pi}}\text{ B4 B9 CP}$$

$$\frac{\partial C(x,t)}{\partial C_a} = \begin{cases} A(x,t) & t_o < t \leq t_{max} \\ A(x,t) - A(x,t-t_{max}) & t > t_{max} \end{cases}$$

$$\frac{\partial C(x,t)}{\partial C_o} = 1 - A(x,t)$$

$$\frac{\partial C(x,t)}{\partial t_{max}} = -\,C_a\,\frac{\partial A(x,t-t_{max})}{\partial t_{max}} \qquad t > t_{max}\,,$$

where

$$\frac{\partial A(x,t-t_{max})}{\partial t_{max}} = -\frac{1}{\sqrt{\pi}} \ B2 \ (v\ B3 + D\ B6\ B3) + \frac{1}{\sqrt{\pi}} \ B4\ B9\ (v\ B3 - D\ B11\ B3)$$

Sensitivity To Velocity By Integration

$$\frac{\partial C(x,t)}{\partial v} = \int_{t^*}^{t} D \frac{\partial}{\partial v}\left[\frac{\partial^2 C}{\partial x^2}\right] dt - \int_{t^*}^{t} \left[\frac{\partial C}{\partial x}\right] dt - \int_{t^*}^{t} v \frac{\partial}{\partial v}\left[\frac{\partial C}{\partial x}\right] dt ,$$

where

$$t^* = \begin{cases} t_o & \text{when} \quad t_o < t \leq t_{max} \\ t_{max} & \text{when} \quad t > t_{max} \end{cases}$$

$$\frac{\partial^n C(x,t)}{\partial x^n} = \begin{cases} (C_a - C_o) \dfrac{\partial^n A(x,t)}{\partial x^n} & t_o < t \leq t_{max} \\ (C_a - C_o) \dfrac{\partial^n A(x,t)}{\partial x^n} - C_a \dfrac{\partial^n A(x,t-t_{max})}{\partial x^n} & t > t_{max} , \end{cases}$$

$$n = 1,2$$

where

$$\frac{\partial}{\partial v}\left(\frac{\partial^2 A(x,t)}{\partial x^2}\right) = \frac{t}{\sqrt{\pi}} B3\ B2\ (B6)^2 - \frac{1}{2D\sqrt{\pi}} B3\ B2 + \frac{1}{D} B5\ B8$$

$$+ \frac{1}{2} (B5)^2\ B8\ B10 - \frac{t}{\sqrt{\pi}} (B5)^2\ B4\ B3\ B9$$

$$- \frac{1}{\sqrt{\pi}} B10\ B4\ B9\ B3\ B5 + \frac{t}{\sqrt{\pi}} B4\ B9\ B11\ B3\ B5$$

$$- \frac{2}{D\sqrt{\pi}} B4\ B9\ B3 - \frac{1}{\sqrt{\pi}} B5\ B4\ B9\ B3\ B10$$

$$+ \frac{t}{\sqrt{\pi}} B5\ B4\ B9\ B11\ B3 + \frac{1}{\sqrt{\pi}} B10\ B4\ B9\ B11\ B3$$

$$- \frac{t}{\sqrt{\pi}} B4\ B9\ (B11)^2\ B3 + \frac{1}{2D\sqrt{\pi}} B4\ B9\ B3$$

$$\frac{\partial^2 A(x,t)}{\partial x^2} = \frac{1}{\sqrt{\pi}} B2\ B6\ B3 + \frac{1}{2} (B5)^2\ B8 - \frac{2}{\sqrt{\pi}} B5\ B4\ B3\ B9$$

$$+ \frac{1}{\sqrt{\pi}} B4\ B11\ B3\ B9$$

$$\frac{\partial A(x,t)}{\partial x} = - \frac{1}{\sqrt{\pi}} B2\ B3 + \frac{1}{2} B5\ B8 - \frac{1}{\sqrt{\pi}} B4\ B9\ B3$$

$$\frac{\partial}{\partial v}\left(\frac{\partial A(x,t)}{\partial x}\right) = - \frac{t}{\sqrt{\pi}} B2\ B6\ B3 + \frac{1}{2D} B8 + \frac{1}{2} B5\ B10\ B8$$

$$- \frac{t}{\sqrt{\pi}} B5\ B4\ B9\ B3 - \frac{1}{\sqrt{\pi}} B10\ B4\ B9\ B3$$

$$+ \frac{t}{\sqrt{\pi}} B4\ B9\ B11\ B3$$

Model B1

Governing equation:

$$R \frac{\partial C}{\partial t} = D \frac{\partial^2 C}{\partial x^2} - v \frac{\partial C}{\partial x} - \mu C$$

Boundary conditions:

$$\left[-D \frac{\partial C}{\partial x} + vC\right]\Big|_{x=0} = \begin{cases} vC_a & t_o < t \leq t_{max} \\ 0 & t > t_{max} \end{cases}$$

$$\frac{\partial C}{\partial x}(\infty, t) = 0$$

Initial condition:

$$C(x, t_o) = C_o$$

Analytical solution:

$$C(x,t) = \begin{cases} C_o \; A(x,t) + C_a \; B(x,t) & t_o < t \leq t_{max} \\ C_o \; A(x,t) + C_a \; B(x,t) - C_a \; B(x, t-t_{max}) & t > t_{max}, \end{cases}$$

where

$$A(x,t) = \exp(-\mu t/R)\left\{1 - \frac{1}{2} \operatorname{erfc}\left[\frac{Rx - vt}{2(DRt)^{1/2}}\right]\right.$$

$$- \left[\frac{v^2 t}{\pi DR}\right]^{1/2} \exp\left[-\frac{(Rx-vt)^2}{4DRt}\right]$$

$$\left. + \frac{1}{2}\left[1 + \frac{vx}{D} + \frac{v^2 t}{DR}\right] \exp(vx/D) \operatorname{erfc}\left[\frac{Rx + vt}{2(DRt)^{1/2}}\right]\right\}$$

$$B(x,t) = \left[\frac{v}{v+u}\right] \exp\left[\frac{(v-u)x}{2D}\right] \operatorname{erfc}\left[\frac{Rx - ut}{2(DRt)^{1/2}}\right]$$

$$+ \left[\frac{v}{v-u}\right] \exp\left[\frac{(v+u)x}{2D}\right] \operatorname{erfc}\left[\frac{Rx + ut}{2(DRt)^{1/2}}\right]$$

$$+ \frac{v^2}{2\mu D} \exp\left[\frac{vx}{D} - \frac{\mu t}{R}\right] \operatorname{erfc}\left[\frac{Rx + vt}{2(DRt)^{1/2}}\right]$$

and

$$u = v\left[1 + \frac{4\mu D}{v^2}\right]^{1/2}.$$

Common terms:

$$v = v/R$$

$$D = D/R$$

$$\mu = \mu/R$$

$$P = \frac{vx}{D}$$

$$CM = \frac{x - vt}{2(Dt)^{1/2}}$$

$$CP = \frac{x + vt}{2(Dt)^{1/2}}$$

$$B1 = 1 + P + \frac{v^2t}{D}$$

$$B2 = \exp[-(CM)^2]$$

$$B3 = \frac{1}{2(Dt)^{1/2}}$$

$$B4 = \exp(P)$$

$$B5 = v/D$$

$$B6 = \frac{x - vt}{2\ Dt}$$

$$B7 = \left[\frac{v^2t}{\pi D}\right]^{1/2}$$

$$B8 = \exp(P)\ \mathrm{erfc}(CP)$$

$$B9 = \exp[-(CP)^2]$$

$$B10 = x/D$$

$$B11 = \frac{x + vt}{2\ Dt}$$

$$\frac{\partial u}{\partial v} = \frac{u}{v} - \frac{4\mu D}{uv}$$

$$\frac{\partial u}{\partial D} = \frac{2\mu}{u}$$

$$\frac{\partial u}{\partial \mu} = \frac{2D}{u}$$

$$BM = \frac{x - ut}{2(Dt)^{1/2}}$$

$$BP = \frac{x + ut}{2(Dt)^{1/2}}$$

$$AM = \frac{(v - u)x}{2D}$$

$$AP = \frac{(v + u)x}{2D}$$

$$E1 = \exp(AM)\mathrm{erfc}(BM)$$

$$E2 = \exp(AM)$$

$$E3 = \exp[-(BM)^2]$$

$$E4 = \exp(AP)\mathrm{erfc}(BP)$$

$$E5 = \frac{1}{v - u}$$

$$E6 = \exp[-(BP)^2]$$

$$E7 = \frac{v^2}{2\mu D}$$

$$E8 = \frac{1}{v + u}$$

$$E9 = \exp[-\mu t]$$

$$E10 = \exp(AP)$$

$$E11 = \exp(P - \mu t)\mathrm{erfc}(CP)$$

$$\frac{\partial C(x,t)}{\partial v} = \begin{cases} C_o \dfrac{\partial A(x,t)}{\partial v} + C_a \dfrac{\partial B(x,t)}{\partial v} & t_o < t \leq t_{max} \\[2ex] C_o \dfrac{\partial A(x,t)}{\partial v} + C_a \dfrac{\partial B(x,t)}{\partial v} - C_a \dfrac{\partial B(x,t-t_{max})}{\partial v} & t > t_{max} , \end{cases}$$

where

$$\frac{\partial A(x,t)}{\partial v} = e^{-\mu t}\Big[- \frac{t}{\sqrt{\pi}} B2\ B3 - \frac{t}{\pi} B5\ B2/B7 - t\ B7\ B6\ B2$$

$$+ \frac{1}{2} (B10 + 2t\ B5)\ B8 + \frac{1}{2} B1\ B10\ B8$$

$$- \frac{t}{\sqrt{\pi}} B1\ B4\ B3\ B9 \Big]$$

$$\frac{\partial B(x,t)}{\partial v} = \left[E8 - v(E8)^2 \left(1 + \frac{\partial u}{\partial v}\right) \right] E1$$

$$+ \frac{v}{2} E8 \left(1 - \frac{\partial u}{\partial v}\right) B10\ E1$$

$$+ \frac{2vt}{\sqrt{\pi}} E8\ E2\ E3 \frac{\partial u}{\partial v} B3$$

$$+ \left[E5 - v(E5)^2 \left(1 - \frac{\partial u}{\partial v}\right) \right] E4$$

$$+ \frac{v}{2} E5 \left(1 + \frac{\partial u}{\partial v}\right) B10\ E4$$

$$- \frac{2vt}{\sqrt{\pi}} E5\ E10\ E6 \frac{\partial u}{\partial v} B3 + \frac{1}{\mu} B5\ E11$$

$$+ E7\ B10\ E11 - \frac{2t}{\sqrt{\pi}} E7 \exp(P - \mu t)\ B9\ B3$$

$$\frac{\partial C(x,t)}{\partial D} = \begin{cases} C_o \dfrac{\partial A(x,t)}{\partial D} + C_a \dfrac{\partial B(x,t)}{\partial D} & t_o < t \leq t_{max} \\ C_o \dfrac{\partial A(x,t)}{\partial D} + C_a \dfrac{\partial B(x,t)}{\partial D} - C_a \dfrac{\partial B(x,t-t_{max})}{\partial D} & t > t_{max}, \end{cases}$$

where

$$\frac{\partial A(x,t)}{\partial D} = e^{-\mu t}\Big\{ - \frac{t}{\sqrt{\pi}} \text{ B2 B6 B3} + \frac{t}{2\pi} (\text{B5})^2 \text{ B2/B7}$$

$$- t \text{ B7 } (\text{B6})^2 \text{ B2} - \frac{1}{2} \Big[\text{B5 B10} + (\text{B5})^2 t \Big] \text{ B8}$$

$$- \frac{P}{2D} \text{ B1 B8} + \frac{1}{2D\sqrt{\pi}} \text{ B1 B4 B9 CP} \Big\}$$

$$\frac{\partial B(x,t)}{\partial D} = -v \Big[(\text{E8})^2 \frac{\partial u}{\partial D} \text{ E1} + \text{E8} \left(\frac{1}{2} \frac{\partial u}{\partial D} \text{ B10} + \frac{AM}{D} \right) \text{E1}$$

$$- \frac{2}{\sqrt{\pi}} \text{ E8 E2 E3} \left(\frac{BM}{2D} + t \text{ B3} \frac{\partial u}{\partial D} \right)$$

$$-(\text{E5})^2 \frac{\partial u}{\partial D} \text{ E4} + \text{E5} \left(\frac{AP}{D} - \frac{1}{2} \text{ B10} \frac{\partial u}{\partial D} \right) \text{E4}$$

$$- \frac{2}{\sqrt{\pi}} \text{ E5 E10 E6} \left(\frac{BP}{2D} - t \text{ B3} \frac{\partial u}{\partial D} \right) \Big]$$

$$- \frac{1}{D} \text{ E7 E11 } (1 + P) + \frac{1}{D\sqrt{\pi}} \text{ E7 } \exp(P - \mu t) \text{ B9 CP}$$

$$\frac{\partial C(x,t)}{\partial \mu} = \begin{cases} - C_o t \, A(x,t) + C_a \dfrac{\partial B(x,t)}{\partial \mu} & t_o < t \leq t_{max} \\ - C_o t \, A(x,t) + C_a \dfrac{\partial B(x,t)}{\partial \mu} - C_a \dfrac{\partial B(x,t-t_{max})}{\partial \mu} & t > t_{max}, \end{cases}$$

where

$$\frac{\partial B(x,t)}{\partial \mu} = -v \Big[(E8)^2 \frac{\partial u}{\partial \mu} E1 + \frac{1}{2} E8\ B10 \frac{\partial u}{\partial \mu} E1$$

$$- \frac{2t}{\sqrt{\pi}} E8\ E2\ E3\ B3 \frac{\partial u}{\partial \mu} - (E5)^2 \frac{\partial u}{\partial \mu} E4$$

$$- \frac{1}{2} E5\ B10 \frac{\partial u}{\partial \mu} E4 + \frac{2t}{\sqrt{\pi}} E5\ E10\ E6\ B3 \frac{\partial u}{\partial \mu} \Big]$$

$$- E7\ E11 \left[\frac{1}{\mu} + t \right]$$

$$\frac{\partial C(x,t)}{\partial C_a} = \begin{cases} B(x,t) & t_o < t \leq t_{max} \\ B(x,t) - B(x,t-t_{max}) & t > t_{max} \end{cases}$$

$$\frac{\partial C(x,t)}{\partial C_o} = A(x,t) \qquad t > t_{max}$$

$$\frac{\partial C(x,t)}{\partial t_{max}} = - C_a \frac{\partial B(x,t-t_{max})}{\partial t_{max}} \qquad t > t_{max} ,$$

where

$$\frac{\partial B(x,t-t_{max})}{\partial t_{max}} = - \frac{2v}{\sqrt{\pi}} E8\ E2\ E3 \left[u\ B3 + 2D\ BM\ (B3)^2 \right]$$

$$+ \frac{2v}{\sqrt{\pi}} E5\ E10\ E6 \left[u\ B3 + 2D\ BP\ (B3)^2 \right]$$

$$- \frac{v}{2} B5\ E11 + \frac{2}{\sqrt{\pi}} E7 \exp (P - \mu t)\ B9\ (v\ B3 - D\ B11\ B3)$$

Model B2

Governing equation:

$$R\frac{\partial C}{\partial t} = D\frac{\partial^2 C}{\partial x^2} - v\frac{\partial C}{\partial x}$$

Boundary conditions:

$$\left[-D\frac{\partial C}{\partial x} + vC\right]\Bigg|_{x=0} = v(C_b + C_c e^{-\lambda t}) \qquad t > t_o$$

$$\frac{\partial C}{\partial x}(\infty,t) = 0$$

Initial condition:

$$C(x,t_o) = C_o$$

Analytical solution:

$$C(x,t) = \begin{cases} C_o\,A(x,t) + C_b\,B(x,t) + C_c\,E(x,t) & \mu \neq \lambda R \\ (C_o - C_c)\,A(x,t) + C_b\,B(x,t) + C_c e^{-\lambda t} & \mu = \lambda R\,, \end{cases}$$

where

$$A(x,t) = \exp(-\mu t/R)\left\{1 - \frac{1}{2}\,\mathrm{erfc}\left[\frac{Rx - vt}{(DRt)^{1/2}}\right]\right.$$

$$- \left[\frac{v^2 t}{\pi DR}\right]^{1/2} \exp\left[-\frac{(Rx - vt)^2}{4DRt}\right]$$

$$\left. + \frac{1}{2}\left[1 + \frac{vx}{D} + \frac{v^2 t}{DR}\right]\exp(vx/D)\;\mathrm{erfc}\left[\frac{Rx + vt}{2(DRt)^{1/2}}\right]\right\}$$

$$B(x,t) = \left[\frac{v}{v+u}\right] \exp\left[\frac{(v-u)x}{2D}\right] \operatorname{erfc}\left[\frac{Rx - ut}{2(DRt)^{1/2}}\right]$$

$$+ \left[\frac{v}{v-u}\right] \exp\left[\frac{(v+u)x}{2D}\right] \operatorname{erfc}\left[\frac{Rx + ut}{2(DRt)^{1/2}}\right]$$

$$+ \frac{v^2}{2\mu D} \exp\left[\frac{vx}{D} - \frac{\mu t}{R}\right] \operatorname{erfc}\left[\frac{Rx + vt}{2(DRt)^{1/2}}\right]$$

$$E(x,t) = e^{-\lambda t} \left\{ \left[\frac{v}{v+w}\right] \exp\left[\frac{(v-w)x}{2D}\right] \operatorname{erfc}\left[\frac{Rx - wt}{2(DRt)^{1/2}}\right] \right.$$

$$\left. + \left[\frac{v}{v-w}\right] \exp\left[\frac{(v+w)x}{2D}\right] \operatorname{erfc}\left[\frac{Rx + wt}{2(DRt)^{1/2}}\right] \right\}$$

$$+ \frac{v^2}{2D(\mu - \lambda R)} \exp\left[\frac{vx}{D} - \frac{\mu t}{R}\right] \operatorname{erfc}\left[\frac{Rx + vt}{2(DRt)^{1/2}}\right]$$

and

$$u = v \left[1 + \frac{4\mu D}{v^2}\right]^{1/2}$$

$$w = v \left[1 + \frac{4D}{v^2} (\mu - \lambda R)\right]^{1/2}$$

Common terms:

$$v = v/R$$

$$D = D/R$$

$$\mu = \mu/R$$

$$P = \frac{vx}{D}$$

$$AM = \frac{x - vt}{2(Dt)^{½}}$$

$$AP = \frac{x + vt}{2(Dt)^{½}}$$

$$B1 = 1 + P + \frac{v^2t}{D}$$

$$B2 = \exp\,[-(AM)^2]$$

$$B3 = \frac{1}{2(Dt)^{½}}$$

$$B4 = \exp\,(P)$$

$$B5 = v/D$$

$$B6 = \frac{x - vt}{2\ Dt}$$

$$B7 = \left[\frac{v^2t}{\pi D}\right]^{½}$$

$$B8 = \exp\,(P)\ \mathrm{erfc}\,(AP)$$

$$B9 = \exp\,[-(AP)^2]$$

$$B10 = x/D$$

$$B11 = \frac{x + vt}{2\ Dt}$$

$$\frac{\partial u}{\partial v} = \frac{u}{v} - \frac{4\mu D}{uv}$$

$$\frac{\partial u}{\partial D} = \frac{2\mu}{u}$$

$$\frac{\partial u}{\partial \mu} = \frac{2D}{u}$$

$$BM = \frac{x - ut}{2(Dt)^{½}}$$

$$BP = \frac{x + ut}{2(Dt)^{½}}$$

$$DM = \frac{(v - u)x}{2D}$$

$$DP = \frac{(v + u)x}{2D}$$

$$E1 = \exp(DM)\mathrm{erfc}(BM)$$

$$E2 = \exp(DM)$$

$$E3 = \exp[-(BM)^2]$$

$$E4 = \exp(DP)\mathrm{erfc}(BP)$$

$$E5 = \frac{1}{v - u}$$

$$E6 = \exp[-(BP)^2]$$

$$E7 = \frac{v^2}{2\mu D}$$

$$E8 = \frac{1}{v + u}$$

$$E9 = \exp[-\mu t]$$

$$E10 = \exp(DP)$$

$$E11 = \exp(P - \mu t)\mathrm{erfc}(AP)$$

Common terms, continued:

$$\frac{\partial w}{\partial v} = \frac{w}{v} - \frac{4D(\mu - \lambda)}{vw}$$

$$\frac{\partial w}{\partial D} = \frac{2}{w}(\mu - \lambda)$$

$$\frac{\partial w}{\partial \mu} = \frac{2D}{w}$$

$$\frac{\partial w}{\partial \lambda} = -\frac{2D}{w}$$

$$CM = \frac{x - wt}{2(Dt)^{1/2}}$$

$$CP = \frac{x + wt}{2(Dt)^{1/2}}$$

$$EM = \frac{(v - w)x}{2D}$$

$$EP = \frac{(v + w)x}{2D}$$

$$F1 = \exp(EM)\operatorname{erfc}(CM)$$

$$F2 = \exp(EM)$$

$$F3 = \exp[-(CM)^2]$$

$$F4 = \exp(EP)\operatorname{erfc}(CP)$$

$$F5 = \frac{1}{v - w}$$

$$F6 = \exp[-(CP)^2]$$

$$F7 = \frac{v^2}{2D(\mu - \lambda)}$$

$$F8 = \frac{1}{v + w}$$

$$F9 = \exp[-\mu t]$$

$$F10 = \exp(EP)$$

$$\frac{\partial C(x,t)}{\partial v} = \begin{cases} C_o \dfrac{\partial A(x,t)}{\partial v} + C_b \dfrac{\partial B(x,t)}{\partial v} + C_c \dfrac{\partial E(x,t)}{\partial v} & \mu \neq \lambda \\ (C_o - C_c) \dfrac{\partial A(x,t)}{\partial v} + C_b \dfrac{\partial B(x,t)}{\partial v} & \mu = \lambda , \end{cases}$$

where

$$\frac{\partial A(x,t)}{\partial v} = e^{-\mu t}\Big[- \frac{t}{\sqrt{\pi}} B2\ B3 - \frac{t}{\pi} B5\ B2/B7 - t\ B7\ B6\ B2$$
$$+ \frac{1}{2} (B10 + 2t\ B5)\ B8 + \frac{1}{2} B1\ B10\ B8$$
$$- \frac{t}{\sqrt{\pi}} B1\ B4\ B3\ B9 \Big]$$

$$\frac{\partial B(x,t)}{\partial v} = \Big[E8 - v(E8)^2 \ (1 + \frac{\partial u}{\partial v}) \Big] E1$$
$$+ \frac{v}{2} E8\ (1 - \frac{\partial u}{\partial v})\ B10\ E1$$
$$+ \frac{2vt}{\sqrt{\pi}} E8\ E2\ E3 \frac{\partial u}{\partial v} B3$$
$$+ \Big[E5 - v(E5)^2 \ (1 - \frac{\partial u}{\partial v}) \Big] E4$$
$$+ \frac{v}{2} E5\ (1 + \frac{\partial u}{\partial v})\ B10\ E4$$
$$- \frac{2vt}{\sqrt{\pi}} E5\ E10\ E6 \frac{\partial u}{\partial v} B3 + \frac{1}{\mu} B5\ E11$$
$$+ E7\ B10\ E11 - \frac{2t}{\sqrt{\pi}} E7 \exp(P - \mu t)\ B9\ B3$$

$$\frac{\partial E(x,t)}{\partial v} = F9 \left\{ \left[F8 - v(F8)^2 \left(1 + \frac{\partial w}{\partial v}\right) \right] F1 \right.$$

$$+ \frac{v}{2} F8 \left(1 - \frac{\partial w}{\partial v}\right) B10\ F1$$

$$+ \frac{2vt}{\sqrt{\pi}} F8\ F2\ F3 \frac{\partial w}{\partial v} B3$$

$$+ \left[F5 - v(F5)^2 \left(1 - \frac{\partial w}{\partial v}\right) \right] F4$$

$$+ \frac{v}{2} F5 \left(1 + \frac{\partial w}{\partial v}\right) B10\ F4$$

$$\left. - \frac{2vt}{\sqrt{\pi}} F5\ F10\ F6 \frac{\partial w}{\partial v} B3 \right\}$$

$$+ \frac{v}{D(\mu - \lambda)} E11 + F7\ B10\ E11 - \frac{2t}{\sqrt{\pi}} F7 \exp(P - \mu t)\ B9\ B3$$

$$\frac{\partial C(x,t)}{\partial D} = \begin{cases} C_o \dfrac{\partial A(x,t)}{\partial D} + C_b \dfrac{\partial B(x,t)}{\partial D} + C_c \dfrac{\partial E(x,t)}{\partial D} & \mu \neq \lambda \\ (C_o - C_c) \dfrac{\partial A(x,t)}{\partial D} + C_b \dfrac{\partial B(x,t)}{\partial D} & \mu = \lambda\ , \end{cases}$$

where

$$\frac{\partial A(x,t)}{\partial D} = e^{-\mu t} \left[- \frac{t}{\sqrt{\pi}} B2\ B6\ B3 + \frac{t}{2\pi} (B5)^2\ B2/B7 \right.$$

$$- t\ B7\ (B6)^2\ B2 - \frac{1}{2} \left[B5\ B10 + (B5)^2 t \right] B8$$

$$- \frac{P}{2D} B1\ B8 + \frac{1}{2D\sqrt{\pi}} B1\ B4\ B9\ AP$$

$$\frac{\partial B(x,t)}{\partial D} = -v\Big[(E8)^2 \frac{\partial u}{\partial D} E1 + E8\ (\frac{1}{2}\frac{\partial u}{\partial D} B10 + \frac{DM}{D})\ E1$$

$$- \frac{2}{\sqrt{\pi}} E8\ E2\ E3\ (\frac{BM}{2D} + t\ B3 \frac{\partial u}{\partial D})$$

$$-(E5)^2 \frac{\partial u}{\partial D} E4 + E5\ (\frac{DP}{D} + \frac{1}{2} B10 \frac{\partial u}{\partial D})\ E4$$

$$- \frac{2}{\sqrt{\pi}} E5\ E10\ E6\ (\frac{BP}{2D} - t\ B3 \frac{\partial u}{\partial D})\Big]$$

$$- \frac{1}{D} E7\ E11\ (1 + P) + \frac{1}{D\sqrt{\pi}} E7\ \exp(P - \mu t)\ B9\ AP$$

$$\frac{\partial E(x,t)}{\partial D} = -v\ F9 \Big[(F8)^2 \frac{\partial w}{\partial D} F1 + F8\ (\frac{1}{2}\frac{\partial w}{\partial D} B10 + \frac{DM}{D})\ F1$$

$$- \frac{2}{\sqrt{\pi}} F8\ F2\ F3\ (\frac{CM}{2D} + t\ B3 \frac{\partial w}{\partial D})$$

$$-(F5)^2 \frac{\partial w}{\partial D} F4 + F5\ (\frac{EP}{D} - \frac{1}{2} B10 \frac{\partial w}{\partial D})\ F4$$

$$- \frac{2}{\sqrt{\pi}} F5\ F10\ F6\ (\frac{CP}{2D} - t\ B3 \frac{\partial w}{\partial D})\Big]$$

$$- \frac{1}{D} F7\ E11\ (1 + P) + \frac{1}{D\sqrt{\pi}} F7\ \exp(P - \mu t)\ B9\ AP$$

$$\frac{\partial C(x,t)}{\partial \mu} = \begin{cases} - C_o t\ A(x,t) + C_b \dfrac{\partial B(x,t)}{\partial \mu} + C_c \dfrac{\partial E(x,t)}{\partial \mu} & \mu \neq \lambda \\ - (C_o - C_c)\ t\ A(x,t) + C_b \dfrac{\partial B(x,t)}{\partial \mu} & \mu = \lambda\ , \end{cases}$$

where

$$\frac{\partial B(x,t)}{\partial \mu} = -v \left[(E8)^2 \frac{\partial u}{\partial \mu} E1 + \frac{1}{2} E8\ B10 \frac{\partial u}{\partial \mu} E1 \right.$$

$$- \frac{2t}{\sqrt{\pi}} E8\ E2\ E3\ B3 \frac{\partial u}{\partial \mu} - (E5)^2 \frac{\partial u}{\partial \mu} E4$$

$$\left. - \frac{1}{2} E5\ B10 \frac{\partial u}{\partial \mu} E4 + \frac{2t}{\sqrt{\pi}} E5\ E10\ E6\ B3 \frac{\partial u}{\partial \mu} \right]$$

$$- E7\ E11 \left[\frac{1}{\mu} + t \right]$$

$$\frac{\partial E(x,t)}{\partial \mu} = -ve^{-\lambda t} \left[(F8)^2 \frac{\partial w}{\partial \mu} F1 + \frac{1}{2} F8\ B10 \frac{\partial w}{\partial \mu} F1 \right.$$

$$- \frac{2t}{\sqrt{\pi}} F8\ F2\ F3\ B3 \frac{\partial w}{\partial \mu} - (F5)^2 \frac{\partial w}{\partial \mu} F4$$

$$\left. - \frac{1}{2} F5\ B10 \frac{\partial w}{\partial \mu} F4 + \frac{2t}{\sqrt{\pi}} F5\ F10\ F6\ B3 \frac{\partial w}{\partial \mu} \right]$$

$$- F7\ E11 \left[\frac{1}{\mu - \lambda} + t \right]$$

$$\frac{\partial C(x,t)}{\partial \lambda} = \begin{cases} C_C \dfrac{\partial E(x,t)}{\partial \lambda} & \mu \neq \lambda \\ -t\ C_C\ e^{-\lambda t} & \mu = \lambda , \end{cases}$$

where

$$\frac{\partial E(x,t)}{\partial \lambda} = ve^{-\lambda t} \left\{ -t(F8\ F1 + F5\ F4) - \frac{\partial w}{\partial \lambda} \left[(F8)^2 F1 \right. \right.$$

$$+ \frac{1}{2} F8\ B10\ F1 - \frac{2t}{\sqrt{\pi}} F8\ F2\ F3\ B3$$

$$- (F5)^2 \; F4 - \frac{1}{2} \; F5 \; B10 \; F4 + \frac{2t}{\sqrt{\pi}} \; F5 \; F10 \; F6 \; B3 \; \Big] \Big\} + \frac{1}{\mu-\lambda} \; F7 \; E11$$

$$\frac{\partial C(x,t)}{\partial C_b} = B(x,t)$$

$$\frac{\partial C(x,t)}{\partial C_c} = \begin{cases} E(x,t) & \mu \neq \lambda \\ -A(x,t) + e^{-\lambda t} & \mu = \lambda \end{cases}$$

$$\frac{\partial C(x,t)}{\partial C_o} = A(x,t)$$

Model C

Governing equation:

$$R_y \frac{\partial C_y}{\partial t} = D_y \frac{\partial^2 C_y}{\partial x^2} - v_y \frac{\partial C_y}{\partial x} \qquad \text{for } y = 1, \ldots, Y$$

Boundary conditions:

$$\left[-D_y \frac{\partial C_y}{\partial x} + v_y C_y\right]\Big|_{x=0} = \begin{cases} v_y (C_a)_y & t_o < t \leq t_{max} \quad \text{for } y = 1, \ldots, Y \\ 0 & t > t_{max} \quad \text{for } y = 1, \ldots, Y \end{cases}$$

$$\frac{\partial C_y}{\partial x}(\infty, t) = 0 \qquad \text{for } y = 1, \ldots, Y$$

Initial condition:

$$C_y(x, t_o) = (C_o)_y \qquad \text{for } y = 1, \ldots, Y$$

Analytical solution:

$$\bar{C}\ (x,t) = \sum_{y=1}^{Y} f_y\ C_y\ ,$$

where

$C_y = C(x,t,v_y D_y)$ from Model A1 for $y = 1, \ldots, Y$ and

$$\sum_{y=1}^{Y} f_y = 1$$

Sensitivities By Direct Differentiation

$$\frac{\partial \bar{C}}{\partial v_y} = f_y \frac{\partial C_y}{\partial v_y}, \qquad \text{where } \frac{\partial C_y}{\partial v_y} \text{ is from Model A1 for } y = 1, \ldots, Y$$

$$\frac{\partial \bar{C}}{\partial D_y} = f_y \frac{\partial C_y}{\partial D_y}, \qquad \text{where } \frac{\partial C_y}{\partial D_y} \text{ is from Model A1 for } y = 1, \ldots, Y$$

$$\frac{\partial \bar{C}}{\partial (C_a)_y} = f_y \frac{\partial C_y}{\partial (C_a)_y}, \quad \text{where } \frac{\partial C_y}{\partial (C_a)_y} \text{ is from Model A1 for } y = 1, \ldots, Y$$

$$\frac{\partial \bar{C}}{\partial (C_o)_y} = f_y \frac{\partial C_y}{\partial (C_o)_y}, \quad \text{where } \frac{\partial C_y}{\partial (C_o)_y} \text{ is from Model A1 for } y = 1, \ldots, Y$$

Note that for purposes of parameter estimation, the sensitivity of $\bar{C}$ to f_y may only be calculated for Y-1 of the f_y's because of their linear dependence. Hence,

$$\frac{\partial \bar{C}}{\partial f_y} = C_y - C_Y, \quad \text{where } C_y \text{ and } C_Y \text{ are from Model A1 for } y = 1, \ldots, Y$$

SUMMARY

This report presents analytically-derived expressions for sensitivities of solute concentration to parameters in the one-dimensional advection-dispersion equation and associated boundary and initial conditions. These sensitivities indicate the response of the analytical solution to small changes in parameters. The magnitude of the sensitivities and their behavior as a function of x and t are dependent on the particular values of the parameters used in the expressions.

For each model listed in Table 1, sensitivity expressions were derived by direct differentiation of the analytical solution to the boundary value problem defined by the model. Selected sensitivities in models A1 and A4 were derived by integration of the advection-dispersion equation over the time domain.

Programs listed in the Appendix are designed for the calculation of all parameters associated with a given model. Programs may be modified to restrict the output to only those parameters of interest.

REFERENCES CITED

Knopman, D.S., Optimal design of sampling for parameter estimation and discrimination among one-dimensional models of transient solute transport in porous media, 315 pp., Ph.D. thesis, Department of Geography and Environmental Engineering, The Johns Hopkins University, Baltimore, Maryland, 1986.

Knopman, D.S., and C.I. Voss, Behavior of sensitivities in the one-dimensional advection-dispersion equation: Implications for parameter estimation and sampling design, Water Resources Research, 23(2), 253-272, 1987.

Knopman, D.S., anc C.I. Voss, Further comments on sensitivities, parameter estimation, and sampling design in one-dimensional analysis of solute transport in porous media, Water Resources Research, in press, 1988.

van Genuchten, M.Th., and W.J. Alves, Analytical solutions of the one-dimensional convective-dispersive solute transport equation, U.S. Department of Agriculture, Technical Bulletin No. 1661, 151 pp., 1982.

APPENDIX: PROGRAM LISTINGS

Programs are written in FORTRAN and closely follow the structure and notation of programs written to accompany the analytical solutions in van Genuchten and Alves [1982]. To avoid opening all file units automatically, the OPEN statements may be deleted from the main program. The output subroutines produce output files compatible with the TELAGRAF graphics software. DO loops for writing into output files should be adjusted as appropriate to include only opened file units. Arrays are dimensioned in COMMON blocks in the main program and output subroutines.

Programs for models B1 and B2 include a zero-order production term GAMMA. When GAMMA is set to zero, the solutions given for B1 and B2 in the previous section apply. Van Genuchten and Alves [1982] present analytical solutions that include the zero-order production term.

Input data fields are generally in the format of 8F10.0 or 10F10.0. Note the exceptions for models A1, A4, and C. READ statements in the beginning of each model define the order of parameters for input.

The program function EXF [van Genuchten and Alves, 1982] is called in each of the programs but is only listed once at the end of the Appendix. The function may be compiled separately from the program file, but the two must be loaded together to create a run file prior to program execution.

Definition of common variables

ALPHA	α, as defined in initial condition $C_1 + C_2 e^{-\alpha x}$
CA	C_a in constant upstream boundary condition
CB	C_b as defined in exponentially decaying upstream condition $C_b + C_c e^{-\lambda t}$
CBAR	well concentration from vertical averaging
CC	C_c as defined in exponentially decaying upstream condition $C_b + C_c e^{-\lambda t}$
CONC	C at the Ith point in space and Jth point in time
CO	C_o in constant initial condition
C1 C2	C_1 and C_2 as defined in spatially varying initial condition $C_1 + C_2 e^{-\alpha x}$
D	dispersion coefficient
DCDALPHA	sensitivity of C to α
DCDCA	sensitivity of C to C_a
DCDCB	sensitivity of C to C_b
DCDCC	sensitivity of C to C_c
DCDCO	sensitivity of C to C_o
DCDC1	sensitivity of C to C_1
DCDC2	sensitivity of C to C_2
DCDD	sensitivity of C to D
DCDLAM	sensitivity of C to λ
DCDMU	sensitivity of C to μ
DCDV	sensitivity of C to v

DLAM	λ as defined in upstream boundary condition $C_b + C_c e^{-\lambda t}$
DMU	μ, first-order decay coefficient
DT	time interval for sensitivity calculations
DTMAX	sensitivity of C to t_{max}
DX	distance interval for sensitivity calculations
EXF(a1,a2)	program function defined as exp(a1)erfc(a2)
GAMMA	γ, zero-order production coefficient
IMAX	number of points in space at which calculations are made
JMAX	number of points in time at which calculations are made
R	retardation factor
T	time relative to t_{max}
TI	initial time for sensitivity calculations
TIME	absolute time
TM	ending time for sensitivity calculations
TMAX	time when solute input at the upstream boundary ceases
V	v, average linear velocity
X	distance from the upstream boundary
XI	initial distance for sensitivity calculations
XM	ending distance for sensitivity calculations

```
C
C
C *********************************************************************
C *                                                                   *
C *      ONE-DIMENSIONAL ADVECTION-DISPERSION EQUATION          A1    *
C *                                                                   *
C *      SEMI-INFINITE PROFILE                                        *
C *      THIRD TYPE BOUNDARY CONDITION                                *
C *      NO PRODUCTION OR DECAY                                       *
C *      LINEAR ADSORPTION (R)                                        *
C *      CONSTANT INITIAL CONCENTRATION (CO)                          *
C *      INPUT CONCENTRATION = CA ( TO < TIME < TMAX )                *
C *                          = 0 ( TIME > TMAX )                      *
C *                                                                   *
C *********************************************************************
C
      IMPLICIT REAL*8 (A-H,O-Z)
C
C Array dimensions are given as (X,TIME)
C
      COMMON /TXC/TIME(200),X(200),CONC(200,200)
      COMMON /DERIV1/DCDV(200,200),DCDD(200,200),DCDCA(200,200)
      COMMON /DERIV2/DTMAX(200,200),DCDCO(200,200)
      COMMON /INTEG1/DDX(200,200),DVD(200,200),DXINT(200,200)
      COMMON /INTEG2/ DDD(200,200),DCDXT(200,200),DVINT(200,200)
      COMMON /INTEG3/DDINT(200,200),SXINT(200,200),SDINT(200,200)
      COMMON /INTEG4/SVINT(200,200),DDCX(200,200),VDCX(200,200)
      COMMON /SPACE/X1X(200,200),X2X(200,200)
C
C Open input file
C
      OPEN(5,FILE='A1.DATA')
C
C Output in file units 6 to 20 is given as a function of time at
C   any number of locations in space.
C
      OPEN(6,FILE='A1.CONC')
      OPEN(7,FILE='A1.DV')
      OPEN(8,FILE='A1.DD')
      OPEN(9,FILE='A1.DCA')
      OPEN(10,FILE='A1.DCO')
      OPEN(11,FILE='A1.DTMAX')
      OPEN(12,FILE='A1.DXINT')
      OPEN(13,FILE='A1.DVINT')
      OPEN(14,FILE='A1.DDINT')
      OPEN(15,FILE='A1.DVT')
      OPEN(16,FILE='A1.X1')
      OPEN(17,FILE='A1.X2')
      OPEN(18,FILE='A1.SXINT')
      OPEN(19,FILE='A1.SDINT')
      OPEN(20,FILE='A1.SVINT')
C
```

```
C Output in file units 29 to 40 is given as a function of space at
C   any number of points in time.
C
      OPEN(29,FILE='A1.CONCX')
      OPEN(30,FILE='A1.DVX')
      OPEN(31,FILE='A1.DDX')
      OPEN(32,FILE='A1.DCAX')
      OPEN(33,FILE='A1.DCOX')
      OPEN(34,FILE='A1.DTMAXX')
      OPEN(35,FILE='A1.DXINTX')
      OPEN(36,FILE='A1.DVINTX')
      OPEN(37,FILE='A1.DDINTX')
      OPEN(38,FILE='A1.DVTX')
      OPEN(39,FILE='A1.X1X')
      OPEN(40,FILE='A1.X2X')
C
C Read input parameters
C
      READ(5,1003) V,D,R,TMAX,C0,CA
      READ(5,1003) XI,DX,XM,TI,DT,TM
C
C ISTEP and JSTEP control the printing interval of sensitivities when
C   a fine discretization is used to calculate time integrals. Set
C   to 1 for all other cases.
C IFLAG controls the set of file units into which output is written
C ILOG, when set to 1, transforms sensitivities to log form.
C
      READ(5,1004) ISTEP,JSTEP,IFLAG,ILOG
C
C
      D=D/R
      V=V/R
      IMAX=(XM+DX-XI)/DX
      JMAX=(TM+DT-TI)/DT
      E=0.0
C
C Begin calculations for each point in space (counted by I) and
C   time (counted by J).
C
      DO 4 J=1,JMAX
      TIME(J)=(TI+(J-1)*DT)
      DO 4 I=1,IMAX
      X(I)=XI+(I-1)*DX
      DO 2 M=1,2
      A=0.0
      DV=0.0
      DD=0.0
      DTM=0.0
      DDDX=0.0
      DVDX=0.0
      DXDX=0.0
      X1=0.0
      X2=0.0
```

```
      DCX=0.0
      VCX=0.0
      T=TIME(J)+((1-M)*TMAX)
      IF(T.LE.0.) GO TO 2
C
C Common terms
C   (Note: overflow in exponents Q,CMA, and CPA will cause
C     program termination.)
C
      CM=(X(I)-V*T)/DSQRT(4.*D*T)
      CMA=DABS(CM)
      CP=(X(I)+V*T)/DSQRT(4.*D*T)
      CPA=DABS(CP)
      Q=V*X(I)/D
      DPI=0.5641895
      API=3.1415926
      B1=1.+Q+(T*(V**2.)/D)
      B2=1./DEXP(CMA**2.)
      B3=.5/DSQRT(D*T)
      B4=DEXP(Q)
      B5=V/D
      B6=.5*(X(I)-(V*T))/(D*T)
      AB6=DABS(B6)
      B7=DSQRT(T*(V**2.)/(API*D))
      B8=EXF(Q,CP)
      B9=1./DEXP(CPA**2.)
      B10=X(I)/D
      B11=.5*(X(I)+(V*T))/(D*T)
C
C Concentration:  A(x,t)
C
      A=0.5*EXF(E,CM)+V*DSQRT(.3183099*T/D)*EXF(-CM*CM,E)-0.5*(1.+Q+V*V
     1*T/D)*EXF(Q,CP)
C
C Sensitivities
C
      DV=DPI*B2*B3*T
      DV=DV+(1./B7)*(B5*T/API)*B2
      DV=DV+B7*B6*T*B2
      DV=DV-.5*(B10+(2.*B5*T))*B8
      DV=DV-.5*B1*B10*B8
      DV=DV+B1*B4*DPI*T*B3*B9
C
      DD=DPI*B2*B6*T*B3
      DD=DD-.5*(1./B7)*(B5**2.)*(T/API)*B2
      DD=DD+B7*(AB6**2.)*T*B2
      DD=DD+.5*(B5*B10+(B5**2.)*T)*B8
      DD=DD+.5*B1*(Q/D)*B8
      DD=DD+B1*B4*DPI*B9*(.5/D)*(-CP)
C
      DTM=-DPI*B2*(V*B3+D*B6*B3)
      DTM=DTM-.5*(V/API)*B5*B2/B7
      DTM=DTM-B7*B2*(V*B6+D*(AB6**2.))
```

```
      DTM=DTM+.5*V*B5*B8
      DTM=DTM+DPI*B1*B4*B9*(-V*B3+D*B11*B3)
C
C Integrands for time integral method (velocity and dispersion)
C
      DDDX=-DPI*B2*B3
      DDDX=DDDX+B7*B2*(-B6)
      DDDX=DDDX-.5*B5*B8
      DDDX=DDDX-.5*B1*B5*B8
      DDDX=DDDX+B1*B4*DPI*B9*B3
C
      DVDX=-DPI*B2*B6*T*B3
      DVDX=DVDX+(.5/B7)*(2.*B5*T/API)*B2*(-B6)
      DVDX=DVDX+B7*B2*T*B6*(-B6)
      DVDX=DVDX+B7*B2*(.5/D)
      DVDX=DVDX+(-.5/D)*B8
      DVDX=DVDX-.5*B5*B10*B8
      DVDX=DVDX+B5*B4*DPI*B9*T*B3
      DVDX=DVDX-.5*(B10+(2.*B5*T))*B5*B8
      DVDX=DVDX-.5*(B1/D)*B8
      DVDX=DVDX-.5*B1*B5*B10*B8
      DVDX=DVDX+B1*B5*B4*DPI*B9*T*B3
      DVDX=DVDX+(B10+(2.*T*B5))*B4*DPI*B9*B3
      DVDX=DVDX+B1*B10*B4*DPI*B9*B3
      DVDX=DVDX+B1*B4*DPI*B9*(-B11*T)*B3
C
      DXDX=(-DPI)*B3*B2*(-B6)*B6*T
      DXDX=DXDX-DPI*B3*B2*(.5/D)
      DXDX=DXDX+.5*(1./B7)*(2.*V*T/(API*D))*B2*(-.5/(D*T))
      DXDX=DXDX+B7*B2*(-.5/(D*T))*B6*T
      DXDX=DXDX+.5*(1./B7)*(2.*V*T/(API*D))*(AB6**2.)*B2
      DXDX=DXDX-B7*(B6/D)*B2
      DXDX=DXDX+B7*(AB6**2.)*B2*B6*T
      DXDX=DXDX-(B5/D)*B8
      DXDX=DXDX-.5*(B5**2.)*B8*B10
      DXDX=DXDX+(B5**2.)*B4*DPI*B9*T*B3
      DXDX=DXDX+(1./D)*B4*DPI*B9*B3
      DXDX=DXDX+B5*B10*B4*DPI*B9*B3
      DXDX=DXDX+B5*B4*DPI*B9*(-B11*T)*B3
      DXDX=DXDX-.5*(B10+(2.*T*B5))*(B5**2.)*B8
      DXDX=DXDX-B1*(B5/D)*B8
      DXDX=DXDX-.5*B1*(B5**2.)*B8*B10
      DXDX=DXDX+B1*(B5**2.)*B4*DPI*T*B3*B9
      DXDX=DXDX-.5*(2.*B5/D)*B8
      DXDX=DXDX-.5*(B5**2.)*B8*B10
      DXDX=DXDX+(B5**2.)*B4*DPI*B9*T*B3
      DXDX=DXDX+(B10+(2.*T*B5))*B4*DPI*B9*B3*B5
      DXDX=DXDX+B1*B10*B4*DPI*B9*B3*B5
      DXDX=DXDX+B1*B4*DPI*B9*(-B11*T)*B3*B5
      DXDX=DXDX+B1*(1./D)*B4*DPI*B9*B3
      DXDX=DXDX+(1./D)*B4*DPI*B9*B3
      DXDX=DXDX+B5*B10*B4*DPI*B9*B3
      DXDX=DXDX+B5*B4*DPI*B9*(-B11*T)*B3
```

```
      DXDX=DXDX+(B10+(2.*T*B5))*B5*B4*DPI*B9*B3
      DXDX=DXDX+B1*(1./D)*B4*DPI*B9*B3
      DXDX=DXDX+B1*B5*B4*DPI*B9*B3*B10
      DXDX=DXDX+B1*B5*B4*DPI*B9*(-B11*T)*B3
      DXDX=DXDX+(B10+(2.*T*B5))*B4*DPI*B9*
     1 (-.5*(X(I)+(V*T))/(D*T))*B3
      DXDX=DXDX+B1*B10*B4*DPI*B9*(-B11)*B3
      DXDX=DXDX+B1*B4*DPI*B9*(B11**2.)*B3*T
      DXDX=DXDX+B1*B4*DPI*B9*(-.5/D)*B3
C
      VCX=-DPI*B2*(CMA**2.)*B3/D
      VCX=VCX+.5*DPI*B2*B3/D
      VCX=VCX+.5*B7*B6*B2/D
      VCX=VCX+B7*B6*B2/D
      VCX=VCX-B7*B6*B2*(CMA**2.)/D
      VCX=VCX+.5*B5*B8/D
      VCX=VCX+.5*(B5**2.)*B10*B8
      VCX=VCX-.5*B5*B4*DPI*B9*CP/D
      VCX=VCX+.5*((Q/D)+(B5**2.)*T)*B5*B8
      VCX=VCX+.5*B1*B5*B8/D
      VCX=VCX+.5*B1*B5*(Q/D)*B8
      VCX=VCX-.5*DPI*B1*B5*B4*B9*(CP/D)
      VCX=VCX-DPI*((Q/D)+(B5**2.)*T)*B9*B3*B4
      VCX=VCX-DPI*B1*(Q/D)*B4*B9*B3
      VCX=VCX+DPI*B1*B4*B9*(CP**2.)*B3/D
      VCX=VCX-.5*DPI*B1*B4*B9*B3/D
C
C Spatial derivatives
C
      X1=-DPI*B2*B3
      X1=X1-B7*B6*B2
      X1=X1-.5*B5*B8
      X1=X1-.5*B1*B5*B8
      X1=X1+DPI*B1*B4*B9*B3
C
      X2=DPI*B2*B6*B3
      X2=X2-B7*(.5/(D*T))*B2
      X2=X2+B7*(AB6**2.)*B2
      X2=X2-.5*(B5**2.)*B8
      X2=X2+DPI*B5*B4*B9*B3
      X2=X2-.5*(B5**2.)*B8
      X2=X2-.5*B1*(B5**2.)*B8
      X2=X2+DPI*B1*B5*B4*B3*B9
      X2=X2+DPI*B5*B4*B9*B3
      X2=X2+DPI*B1*B5*B4*B9*B3
      X2=X2-DPI*B1*B4*B11*B3*B9
C
C Form complete expressions ( TO < TIME < TMAX )
C
      IF(M.EQ.2) GO TO 3
      CONC(I,J)=C0+(CA-C0)*A
      DCDV(I,J)=(CA-C0)*DV
      DCDD(I,J)=(CA-C0)*DD
```

```
      DCDCA(I,J)=A
      DCDCO(I,J)=1.-A
      DDX(I,J)=(CA-CO)*DDDX
      DVD(I,J)=(CA-CO)*DVDX*V
      DDD(I,J)=(CA-CO)*DXDX*D
      X1X(I,J)=(CA-CO)*X1
      X2X(I,J)=(CA-CO)*X2
      DDCX(I,J)=(CA-CO)*DCX*D
      VDCX(I,J)=(CA-CO)*VCX*V
    2 CONTINUE
C
C Form complete expressions (TIME > TMAX )
C
    3 CONC(I,J)=CONC(I,J)-CA*A
      DCDV(I,J)=DCDV(I,J)-CA*DV
      DCDD(I,J)=DCDD(I,J)-CA*DD
      DCDCA(I,J)=DCDCA(I,J)-A
      DCDCO(I,J)=DCDCO(I,J)
      DTMAX(I,J)=-CA*DTM
      DDX(I,J)=DDX(I,J)-CA*DDDX
      DVD(I,J)=DVD(I,J)-CA*DVDX*V
      DDD(I,J)=DDD(I,J)-CA*DXDX*D
      X1X(I,J)=X1X(I,J)-CA*X1
      X2X(I,J)=X2X(I,J)-CA*X2
      DDCX(I,J)=DDCX(I,J)-CA*DCX*D
      VDCX(I,J)=VDCX(I,J)-CA*VCX*V
    4 CONTINUE
C
C Optional log transformation
C
      IF (ILOG.NE.1) GO TO 6
      DO 5 I=1,IMAX
      DO 5 J=1,JMAX
      IF (CONC(I,J).LT.1.D-50) CONC(I,J)=1.D-50
        DCDV(I,J)=DCDV(I,J)/CONC(I,J)
        DCDD(I,J)=DCDD(I,J)/CONC(I,J)
        DCDCA(I,J)=DCDCA(I,J)/CONC(I,J)
    5   CONC(I,J)=DLOG(CONC(I,J))
C
C Integration using the trapezoidal rule
C
    6 DO 8 I=1,IMAX
      DO 8 J=1,JMAX
      IF (J.NE.1) GO TO 7
      DXINT(I,J)=0.
      DVINT(I,J)=0.
      DDINT(I,J)=0.
      SXINT(I,J)=0.
      SVINT(I,J)=0.
      GO TO 8
    7   DXINT(I,J)=DXINT(I,J-1)+.5*(DDX(I,J-1)+DDX(I,J))*DT
        DVINT(I,J)=DVINT(I,J-1)+.5*(DVD(I,J-1)+DVD(I,J))*DT
        DDINT(I,J)=DDINT(I,J-1)+.5*(DDD(I,J-1)+DDD(I,J))*DT
```

```
          DCDXT(I,J)=DDINT(I,J)-DXINT(I,J)-DVINT(I,J)
          SXINT(I,J)=SXINT(I,J-1)+.5*(X2X(I,J-1)+X2X(I,J))*DT
          SVINT(I,J)=SVINT(I,J-1)+.5*(VDCX(I,J-1)+VDCX(I,J))*DT
          SDINT(I,J)=DCDD(I,J)-SXINT(I,J)+SVINT(I,J)
    8 CONTINUE
C
C Print sensitivities and other terms
C
      CALL PLOTA1(IMAX,JMAX,ISTEP,JSTEP,IFLAG)
C
C
 1003 FORMAT(8F10.0)
 1004 FORMAT(4I5)
      STOP
      END
C
C
      SUBROUTINE PLOTA1(IMAX,JMAX,ISTEP,JSTEP,IFLAG)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /TXC/TIME(200),X(200),CONC(200,200)
      COMMON /DERIV1/DCDV(200,200),DCDD(200,200),DCDCA(200,200)
      COMMON /DERIV2/DTMAX(200,200),DCDC0(200,200)
      COMMON /INTEG1/DDX(200,200),DVD(200,200),DXINT(200,200)
      COMMON /INTEG2/DDD(200,200),DCDXT(200,200),DVINT(200,200)
      COMMON /INTEG3/DDINT(200,200),SXINT(200,200),SDINT(200,200)
      COMMON /INTEG4/SVINT(200,200),DDCX(200,200),VDCX(200,200)
      COMMON /SPACE/X1X(200,200),X2X(200,200)
      ISTEP1=1
      JSTEP1=1
      IF (ISTEP.NE.1) ISTEP1=ISTEP
      IF (JSTEP.NE.1) JSTEP1=JSTEP
      DO 10 K=6,20
   10   WRITE(K,1008)
      DO 20 K=29,40
   20   WRITE(K,1008)
      IF (IFLAG.GT.1) GO TO 90
      DO 30 I=ISTEP1,IMAX,ISTEP
      DO 40 K=6,20
   40   WRITE(K,1010)X(I)
        DO 30 J=JSTEP1,JMAX,JSTEP
          WRITE(6,1020)TIME(J),CONC(I,J)
          WRITE(7,1020)TIME(J),DCDV(I,J)
          WRITE(8,1020)TIME(J),DCDD(I,J)
          WRITE(9,1020)TIME(J),DCDCA(I,J)
          WRITE(10,1020)TIME(J),DCDC0(I,J)
          WRITE(11,1020)TIME(J),DTMAX(I,J)
          WRITE(12,1020)TIME(J),DXINT(I,J)
          WRITE(13,1020)TIME(J),DVINT(I,J)
          WRITE(14,1020)TIME(J),DDINT(I,J)
          WRITE(15,1020)TIME(J),DCDXT(I,J)
          WRITE(16,1020)TIME(J),X1X(I,J)
          WRITE(17,1020)TIME(J),X2X(I,J)
          WRITE(18,1020)TIME(J),SXINT(I,J)
```

```
          WRITE(19,1020)TIME(J),SDINT(I,J)
          WRITE(20,1020)TIME(J),SVINT(I,J)
   30 CONTINUE
      DO 50 K=6,20
   50   WRITE(K,1009)
   90 IF (IFLAG.EQ.0) RETURN
      DO 60 J=JSTEP1,JMAX,JSTEP
      DO 70 K=29,40
   70   WRITE(K,1030)TIME(J)
        DO 60 I=ISTEP1,IMAX,ISTEP
          WRITE(29,1020)X(I),CONC(I,J)
          WRITE(30,1020)X(I),DCDV(I,J)
          WRITE(31,1020)X(I),DCDD(I,J)
          WRITE(32,1020)X(I),DCDCA(I,J)
          WRITE(33,1020)X(I),DCDCO(I,J)
          WRITE(34,1020)X(I),DTMAX(I,J)
          WRITE(35,1020)X(I),DXINT(I,J)
          WRITE(36,1020)X(I),DVINT(I,J)
          WRITE(37,1020)X(I),DDINT(I,J)
          WRITE(38,1020)X(I),DCDXT(I,J)
          WRITE(39,1020)X(I),X1X(I,J)
          WRITE(40,1020)X(I),X2X(I,J)
   60 CONTINUE
      DO 80 K=29,40
   80   WRITE(K,1009)
C
C
 1008 FORMAT('INPUT DATA.')
 1009 FORMAT('END OF DATA.')
 1010 FORMAT('"DISTANCE=',F6.0,'"')
 1020 FORMAT(F11.2,1X,E11.3E4)
 1030 FORMAT('"TIME=',F11.0,'"')
 1050 FORMAT(2(F8.2,2X),E12.6E2)
      RETURN
      END
```

```
C
C     *****************************************************************
C     *                                                               *
C     *       ONE-DIMENSIONAL ADVECTION-DISPERSION EQUATION       A2  *
C     *                                                               *
C     *       THIRD-TYPE BOUNDARY CONDITION                           *
C     *       SEMI-INFINITE PROFILE                                   *
C     *       LINEAR ADSORPTION (R)                                   *
C     *       CONSTANT INITIAL CONCENTRATION (C0)                     *
C     *       INPUT CONCENTRATION = CB + CC*EXP(-DLAM*T)              *
C     *                                                               *
C     *****************************************************************
C
      IMPLICIT REAL*8 (A-H,O-Z)
C
C Array dimensions are given as (X,TIME)
C
      COMMON /TXC/TIME(200),X(200),CONC(200,200)
      COMMON /DERIV1/DCDV(200,200),DCDD(200,200),DCDC0(200,200)
      COMMON /DERIV2/DCDCB(200,200),DCDCC(200,200),DCDLAM(200,200)
C
C Open input file
C
      OPEN(5,FILE='A2.DATA')
C
C Output in file units 6 to 12 is given as a function of time at
C   any number of locations in space.
C
      OPEN(6,FILE='A2.CONC')
      OPEN(7,FILE='A2.DV')
      OPEN(8,FILE='A2.DD')
      OPEN(9,FILE='A2.DC0')
      OPEN(10,FILE='A2.DCB')
      OPEN(11,FILE='A2.DCC')
      OPEN(12,FILE='A2.DLAMDA')
C
C Output in file unit 29 is given as a function of space at
C   any number of points in time.
      OPEN(29,FILE='A2.CONCX')
C
C Read input parameters
C
      READ(5,1003) V,D,R,DLAM,C0,CB,CC
      READ(5,1003) XI,DX,XM,TI,DT,TM
C
C
      D=D/R
      V=V/R
      S=1.-4.*DLAM*D/V**2
      IF(S.LE.0.) GO TO 5
      Y=V*DSQRT(S)
      DYV=DSQRT(S)+.5*V*(1./DSQRT(S))*8.*DLAM*D/(V**3.)
      DYD=.5*V*(1./DSQRT(S))*(-4.)*DLAM/(V**2.)
```

```
      DYLAM=.5*V*(1./DSQRT(S))*(-4.)*D/(V**2.)
      IMAX=(XM+DX-XI)/DX
      JMAX=(TM+DT-TI)/DT
      E=0.0
C
C Begin calculations for each point in space (counted by I) and
C   time (counted by J)
C
      DO 4 J=1,JMAX
      T=TI+(J-1)*DT
      TIME(J)=T
      DO 4 I=1,IMAX
      X(I)=XI+(I-1)*DX
C
C Common terms
C
      P=V*X(I)/D
      S=DSQRT(4.*D*T)
      AM=(X(I)-V*T)/S
      AMA=DABS(AM)
      AP=(X(I)+V*T)/S
      APA=DABS(AP)
      BM=(X(I)-Y*T)/S
      BMA=DABS(BM)
      BP=(X(I)+Y*T)/S
      BPA=DABS(BP)
      CM=0.5*(V-Y)*X(I)/D
      CP=0.5*(V+Y)*X(I)/D
C
      DPI=0.5641895
      API=3.1415926
      B1=1.+P+(T*(V**2.)/D)
      B2=1./DEXP(AMA**2.)
      B3=.5/DSQRT(D*T)
      B4=DEXP(P)
      B5=V/D
      B6=.5*(X(I)-(V*T))/(D*T)
      AB6=DABS(B6)
      B7=DSQRT(T*(V**2.)/(API*D))
      B8=EXF(P,AP)
      B9=1./DEXP(APA**2.)
      B10=X(I)/D
      B11=.5*(X(I)+(V*T))/(D*T)
C
      E1=EXF(CM,BM)
      E2=DEXP(CM)
      E3=1./DEXP(BMA**2.)
      E4=EXF(CP,BP)
      E5=1./(V-Y)
      E6=1./DEXP(BPA**2.)
      E7=(V**2.)/(2.*DLAM*D)
      E8=1./(V+Y)
      E9=DEXP(-DLAM*T)
```

```
      E10=DEXP(CP)
C
C Concentration: A(x,t)
C
      A=0.5*EXF(E,AM)+V*DSQRT(0.3183099*T/D)*EXF(-AM*AM,E)-0.5*(1.+P+V
     1*V*T/D)*EXF(P,AP)
C
C Concentration: B(x,t)
      B=DEXP(-DLAM*T)*(V/(V+Y)*EXF(CM,BM)+V/(V-Y)*EXF(CP,BP))-0.5*V*V/
     1(DLAM*D)*EXF(P,AP)
C
C Sensitivities (derivatives of A(x,t))
C
      DVA=DPI*B2*B3*T
      DVA=DVA+(1./B7)*(B5*T/API)*B2
      DVA=DVA+B7*B6*T*B2
      DVA=DVA-.5*(B10+(2.*B5*T))*B8
      DVA=DVA-.5*B1*B10*B8
      DVA=DVA+B1*B4*DPI*T*B3*B9
C
      DDA=DPI*B2*B6*T*B3
      DDA=DDA-.5*(1./B7)*(B5**2.)*(T/API)*B2
      DDA=DDA+B7*(AB6**2.)*T*B2
      DDA=DDA+.5*(B5*B10+(B5**2.)*T)*B8
      DDA=DDA+.5*B1*(P/D)*B8
      DDA=DDA+B1*B4*DPI*B9*(.5/D)*(-AP)
C
C Sensitivities (derivatives of B(x,t))
C
      DVB=E9*(E8-V*(E8**2.)*(1.+DYV))*E1
      DVB=DVB+E9*V*E8*(1.-DYV)*.5*B10*E1
      DVB=DVB+E9*V*E8*E2*2.*DPI*E3*DYV*T*B3
      DVB=DVB+E9*(E5-V*(E5**2.)*(1.-DYV))*E4
      DVB=DVB+E9*V*E5*(1.+DYV)*.5*B10*E4
      DVB=DVB-E9*V*E5*E10*2.*DPI*DYV*T*B3*E6
      DVB=DVB-(V/(DLAM*D))*B8
      DVB=DVB-E7*B10*B8
      DVB=DVB+E7*B4*2.*DPI*B9*T*B3
C
      DDB=-E9*V*(E8**2.)*DYD*E1
      DDB=DDB-E9*V*E8*(DYD*.5*B10+(CM/D))*E1
      DDB=DDB+E9*V*E8*E2*DPI*2.*E3*((.5*BM/D)+B3*T*DYD)
      DDB=DDB+E9*V*(E5**2.)*DYD*E4
      DDB=DDB-E9*V*E5*((CP/D)-.5*B10*DYD)*E4
      DDB=DDB+E9*V*E5*E10*2.*DPI*E6*((.5*BP/D)-B3*T*DYD)
      DDB=DDB+(E7/D)*B8
      DDB=DDB+E7*(P/D)*B8
      DDB=DDB-E7*B4*DPI*B9*AP/D
C
      DL=-T*DEXP(-DLAM*T)*(V/(V+Y)*EXF(CM,BM)+V/(V-Y)*EXF(CP,BP))
      DL=DL-E9*V*(E8**2.)*DYLAM*E1
      DL=DL-E9*V*E8*.5*B10*DYLAM*E1
      DL=DL+E9*V*E8*E2*2.*DPI*E3*T*B3*DYLAM
```

```
      DL=DL+E9*V*(E5**2.)*DYLAM*E4
      DL=DL+E9*V*E5*.5*B10*DYLAM*E4
      DL=DL-E9*V*E5*E10*2.*DPI*E6*T*B3*DYLAM
      DL=DL+(E7/DLAM)*B8
C
C Form complete expressions
C
      CONC(I,J)=C0+(CB-C0)*A+CC*B
      DCDV(I,J)=(CB-C0)*DVA+CC*DVB
      DCDD(I,J)=(CB-C0)*DDA+CC*DDB
      DCDC0(I,J)=1.-A
      DCDCB(I,J)=A
      DCDCC(I,J)=B
      DCDLAM(I,J)=CC*DL
    4 CONTINUE
C
C Print sensitivities
C
      CALL PLOTA2(IMAX,JMAX)
      STOP
C
    5 WRITE(6,1007)
 1003 FORMAT(8F10.0)
 1007 FORMAT(///5X,6(1H*),' DLAM TOO LARGE, THIS CASE NOT EXECUTED',6(1H
     1*))
      STOP
      END
C
C
C
      SUBROUTINE PLOTA2(IMAX,JMAX)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /TXC/TIME(200),X(200),CONC(200,200)
      COMMON /DERIV1/DCDV(200,200),DCDD(200,200),DCDC0(200,200)
      COMMON /DERIV2/DCDCB(200,200),DCDCC(200,200),DCDLAM(200,200)
      DO 5 K=6,12
    5 WRITE(K,1008)
      WRITE(29,1008)
      DO 10 I=1,IMAX
      DO 15 K=6,12
   15   WRITE(K,1010)X(I)
        DO 10 J=1,JMAX
          WRITE(6,1020)TIME(J),CONC(I,J)
          WRITE(7,1020)TIME(J),DCDV(I,J)
          WRITE(8,1020)TIME(J),DCDD(I,J)
          WRITE(9,1020)TIME(J),DCDC0(I,J)
          WRITE(10,1020)TIME(J),DCDCB(I,J)
          WRITE(11,1020)TIME(J),DCDCC(I,J)
          WRITE(12,1020)TIME(J),DCDLAM(I,J)
   10 CONTINUE
      DO 20 J=1,JMAX
        WRITE(29,1030)TIME(J)
        DO 20 I=1,IMAX
```

```
          WRITE(29,1020)X(I),CONC(I,J)
   20  CONTINUE
       DO 30 K=6,12
    30 WRITE(K,1009)
       WRITE(29,1009)
C
C
 1008 FORMAT('INPUT DATA.')
 1009 FORMAT('END OF DATA.')
 1010 FORMAT('" ',F10.0,' METERS"')
 1020 FORMAT(F11.2,1X,E12.4E4)
 1030 FORMAT('" ',F10.0,' DAYS"')
 1050 FORMAT(2(F8.2,2X),E12.6E2)
      RETURN
      END
```

```
C     ************************************************************************
C     *                                                                      *
C     *       ONE-DIMENSIONAL ADVECTION-DISPERSION EQUATION           A3   *
C     *                                                                      *
C     *       THIRD-TYPE BOUNDARY CONDITION                                  *
C     *       SEMI-INFINITE PROFILE                                          *
C     *       LINEAR ADSORPTION (R)                                          *
C     *       INITIAL CONCENTRATION  = C1+C2*EXP(-ALPHA*X)                   *
C     *       INPUT CONCENTRATION    = CA ( T0 < TIME < TMAX )               *
C     *                              = 0  ( TIME > TMAX )                    *
C     *                                                                      *
C     ************************************************************************
C
      IMPLICIT REAL*8 (A-H,O-Z)
C
C Array dimensions are given as (X,TIME)
C
      COMMON /TXC/TIME(200),X(200),CONC(200,200)
      COMMON /DERIV1/DCDV(200,200),DCDD(200,200),DCDCA(200,200)
      COMMON /DERIV2/DCDC1(200,200),DCDC2(200,200),DCDALPHA(200,200)
      COMMON /DERIV3/DTMAX(200,200)
C
C Open input file
C
      OPEN(5,FILE='A3.DATA')
C
C Output in file units 6 to 13 is given as a function of time at
C   any number of locations in space.
C
      OPEN(6,FILE='A3.CONC')
      OPEN(7,FILE='A3.DV')
      OPEN(8,FILE='A3.DD')
      OPEN(9,FILE='A3.DCA')
      OPEN(10,FILE='A3.DC1')
      OPEN(11,FILE='A3.DC2')
      OPEN(12,FILE='A3.DALPHA')
      OPEN(13,FILE='A3.DTMAX')
C
C Output in file unit 29 is given as function of space at
C   any number of points in time.
C
      OPEN(29,FILE='A3.CONCX')
C
C Read input parameters
C
      READ(5,1003) V,D,R,TMAX,C1,C2,CA
      READ(5,1003) XI,DX,XM,ALPHA,TI,DT,TM
C
C
      D=D/R
      V=V/R
      IMAX=(XM+DX-XI)/DX
      JMAX=(TM+DT-TI)/DT
```

```
      E=0.0
C
C Begin calculations for each point in space (counted by I) and
C   time (counted by J).
C
      DO 4 J=1,JMAX
      TIME(J)=TI+(J-1)*DT
      DO 4 I=1,IMAX
      X(I)=XI+(I-1)*DX
      DO 2 M=1,2
      A=0.
      DVA=0.
      DDA=0.
      DTM=0.
      T=TIME(J)+(1-M)*TMAX
      IF(T.LE.0.) GO TO 2
C
C Common terms
C
      P=V*X(I)/D
      S=DSQRT(4.*D*T)
      AM=(X(I)-V*T)/S
      AMA=DABS(AM)
      AP=(X(I)+V*T)/S
      APA=DABS(AP)
      BM=(X(I)-(V+2.*ALPHA*D)*T)/S
      BMA=DABS(BM)
      BP=(X(I)+(V+2.*ALPHA*D)*T)/S
      BPA=DABS(BP)
C
      DPI=0.5641895
      API=3.1415926
      B1=1.+P+(T*(V**2.)/D)
      B2=1./DEXP(AMA**2.)
      B3=.5/DSQRT(D*T)
      B4=DEXP(P)
      B5=V/D
      B6=.5*(X(I)-(V*T))/(D*T)
      AB6=DABS(B6)
      B7=DSQRT(T*(V**2.)/(API*D))
      B8=EXF(P,AP)
      B9=1./DEXP(APA**2.)
      B10=X(I)/D
      B11=.5*(X(I)+(V*T))/(D*T)
C
      D1=DEXP(ALPHA*(ALPHA*D*T+V*T-X(I)))
      D2=1.+V/(ALPHA*D)
      D2A=P+2.*ALPHA*X(I)
      D3=DEXP(D2A)
      D4=.5/(ALPHA*D)
      D5=.5*(2.-EXF(E,BM)+(1.+V/(ALPHA*D))*EXF(P+2.*ALPHA*X(I),BP))
      D6=1./DEXP(BMA**2.)
      D7=EXF(P+2.*ALPHA*X(I),BP)
```

```
      D8=1./DEXP(BPA**2.)
      D9=.5*V/(ALPHA*D)
C
C Concentration: A(x,t)
C
      A=0.5*EXF(E,AM)+V*DSQRT(.3183099*T/D)*EXF(-AM*AM,E)-0.5*(1.+P+V*V*
     1T/D)*EXF(P,AP)
C
C Sensitivities (derivatives of A(x,t))
C
      DVA=DPI*B2*B3*T
      DVA=DVA+(1./B7)*(B5*T/API)*B2
      DVA=DVA+B7*B6*T*B2
      DVA=DVA-.5*(B10+(2.*B5*T))*B8
      DVA=DVA-.5*B1*B10*B8
      DVA=DVA+B1*B4*DPI*T*B3*B9
C
      DDA=DPI*B2*B6*T*B3
      DDA=DDA-.5*(1./B7)*(B5**2.)*(T/API)*B2
      DDA=DDA+B7*(AB6**2.)*T*B2
      DDA=DDA+.5*(B5*B10+(B5**2.)*T)*B8
      DDA=DDA+.5*B1*(P/D)*B8
      DDA=DDA+B1*B4*DPI*B9*(.5/D)*(-AP)
C
      DTM=-DPI*B2*(V*B3+D*B6*B3)
      DTM=DTM-.5*(V/API)*B5*B2/B7
      DTM=DTM-B7*B2*(V*B6+D*(AB6**2.))
      DTM=DTM+.5*V*B5*B8
      DTM=DTM+DPI*B1*B4*B9*(-V*B3+D*B11*B3)
C
C Concentration ( T0 < TIME < TMAX )
C
      IF(M.EQ.2) GO TO 3
      CONC(I,J)=C1+(CA-C1)*A+.5*C2*(EXF(ALPHA*(ALPHA*D*T+V*T-X(I)),E)*
     1(2.-EXF(E,BM)+(1.+V/(ALPHA*D))*EXF(P+2.*ALPHA*X(I),BP))-
     2V/(ALPHA*D)*EXF(P,AP))
C
C Sensitivities (derivatives of B(x,t))
C
      DVB=D1*ALPHA*T*D5
      DVB=DVB-D1*DPI*D6*T*B3
      DVB=DVB+D1*D4*D7
      DVB=DVB+.5*D1*D2*B10*D7
      DVB=DVB-D1*D2*D3*D8*T*B3*DPI
      DVB=DVB-D4*B8
      DVB=DVB-D9*B10*B8
      DVB=DVB+2.*DPI*D9*B4*B9*T*B3
C
      DDB=((ALPHA**2.)*T)*D1*D5
      DDB=DDB-.5*D1*DPI*D6*(B3*B10-B3*B5*T+B3*ALPHA*T*2.)
      DDB=DDB-D1*(D9/D)*D7
      DDB=DDB-.5*D1*D2*(P/D)*D7
      DDB=DDB+.5*D1*D2*DPI*D3*D8*(B10*B3+B3*B5*T-B3*ALPHA*T*2.)
```

```
      DDB=DDB+D9*B8/D
      DDB=DDB+D9*P*B8/D
      DDB=DDB-D9*B4*DPI*B9*AP/D
C
      DALPHA=(2.*ALPHA*D*T+V*T-X(I))*D1*D5
      DALPHA=DALPHA-D1*DPI*D6*2.*D*T*B3
      DALPHA=DALPHA-D1*(D9/ALPHA)*D7
      DALPHA=DALPHA+D1*D2*X(I)*D7
      DALPHA=DALPHA-D1*D2*D3*DPI*D8*D*T*B3*2.
      DALPHA=DALPHA+(D9/ALPHA)*B8
C
C Form complete expressions for sensitivities ( T0 < TIME < TMAX )
C
      DCDV(I,J)=(CA-C1)*DVA+C2*DVB
      DCDD(I,J)=(CA-C1)*DDA+C2*DDB
      DCDCA(I,J)=A
      DCDC1(I,J)=1.-A
      DCDALPHA(I,J)=C2*DALPHA
      DCDC2(I,J)=D1*D5-D9*B8
    2 CONTINUE
C
C Form complete expressions for sensitivities ( TIME > TMAX )
C
    3 CONC(I,J)=CONC(I,J)-CA*A
      DCDV(I,J)=DCDV(I,J)-CA*DVA
      DCDD(I,J)=DCDD(I,J)-CA*DDA
      DCDCA(I,J)=DCDCA(I,J)-A
      DCDC1(I,J)=DCDC1(I,J)
      DCDC2(I,J)=DCDC2(I,J)
      DCDALPHA(I,J)=DCDALPHA(I,J)
      DTMAX(I,J)=-CA*DTM
    4 CONTINUE
C
C Print sensitivities
C
      CALL PLOTA3(IMAX,JMAX)
C
C
 1003 FORMAT(8F10.0)
      STOP
      END
C
C
C
      SUBROUTINE PLOTA3(IMAX,JMAX)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /TXC/TIME(200),X(200),CONC(200,200)
      COMMON /DERIV1/DCDV(200,200),DCDD(200,200),DCDCA(200,200)
      COMMON /DERIV2/DCDC1(200,200),DCDC2(200,200),DCDALPHA(200,200)
      COMMON /DERIV3/DTMAX(200,200)
      DO 5 K=6,13
    5 WRITE(K,1008)
      WRITE(29,1008)
```

```
      DO 10 I=1,IMAX
      DO 15 K=6,13
   15   WRITE(K,1010)X(I)
        DO 10 J=1,JMAX
          WRITE(6,1020)TIME(J),CONC(I,J)
          WRITE(7,1020)TIME(J),DCDV(I,J)
          WRITE(8,1020)TIME(J),DCDD(I,J)
          WRITE(9,1020)TIME(J),DCDCA(I,J)
          WRITE(10,1020)TIME(J),DCDC1(I,J)
          WRITE(11,1020)TIME(J),DCDC2(I,J)
          WRITE(12,1020)TIME(J),DCDALPHA(I,J)
          WRITE(13,1020)TIME(J),DTMAX(I,J)
   10 CONTINUE
      DO 20 J=1,JMAX
        WRITE(29,1030)TIME(J)
        DO 20 I=1,IMAX
          WRITE(29,1020)X(I),CONC(I,J)
  20  CONTINUE
      DO 30 K=6,13
   30 WRITE(K,1009)
      WRITE(29,1009)
C
C
 1008 FORMAT('INPUT DATA.')
 1009 FORMAT('END OF DATA.')
 1010 FORMAT('" ',F10.0,' METERS"')
 1020 FORMAT(F11.2,1X,E12.4E4)
 1030 FORMAT('" ',F10.0,' DAYS"')
 1050 FORMAT(2(F8.2,2X),E12.6E2)
      RETURN
      END
```

```
C
C
C     *******************************************************************
C     *                                                                 *
C     *       ONE-DIMENSIONAL ADVECTION-DISPERSION EQUATION       A4   *
C     *                                                                 *
C     *       SEMI-INFINITE PROFILE                                     *
C     *       FIRST TYPE BOUNDARY CONDITION                             *
C     *       NO PRODUCTION OR DECAY                                    *
C     *       LINEAR ADSORPTION (R)                                     *
C     *       CONSTANT INITIAL CONCENTRATION (C0)                       *
C     *       INPUT CONCENTRATION   = CA ( T0 < TIME < TMAX )           *
C     *                             = 0  ( TIME > TMAX )                *
C     *                                                                 *
C     *******************************************************************
C
      IMPLICIT REAL*8 (A-H,O-Z)
C
C Array dimensions are given as (X,TIME)
C
      COMMON /TXC/TIME(200),X(200),CONC(200,200)
      COMMON /DERIV1/DCDV(200,200),DCDD(200,200),DCDCA(200,200)
      COMMON /DERIV2/DTMAX(200,200),DCDC0(200,200)
      COMMON /INTEG1/DDX(200,200),DVD(200,200),DXINT(200,200)
      COMMON /INTEG2/DDD(200,200),DCDXT(200,200),DVINT(200,200)
      COMMON /INTEG3/DDINT(200,200)
      COMMON /SPACE/X1X(200,200),X2X(200,200)
C
C Open input file
C
      OPEN(5,FILE='A4.DATA')
C
C Output in file units 6 to 17 is given as a function of time at
C   any number of locations in space.
C
      OPEN(6,FILE='A4.CONC')
      OPEN(7,FILE='A4.DV')
      OPEN(8,FILE='A4.DD')
      OPEN(9,FILE='A4.DCA')
      OPEN(10,FILE='A4.DC0')
      OPEN(11,FILE='A4.DTMAX')
      OPEN(12,FILE='A4.DXINT')
      OPEN(13,FILE='A4.DVINT')
      OPEN(14,FILE='A4.DDINT')
      OPEN(15,FILE='A4.DVT')
      OPEN(16,FILE='A4.X1')
      OPEN(17,FILE='A4.X2')
C
C Output in file units 29 to 40 is given as a function of space at
C   any number of points in time.
C
      OPEN(29,FILE='A4.CONCX')
      OPEN(30,FILE='A4.DVX')
```

```
      OPEN(31,FILE='A4.DDX')
      OPEN(32,FILE='A4.DCAX')
      OPEN(33,FILE='A4.DCOX')
      OPEN(34,FILE='A4.DTMAXX')
      OPEN(35,FILE='A4.DXINTX')
      OPEN(36,FILE='A4.DVINTX')
      OPEN(37,FILE='A4.DDINTX')
      OPEN(38,FILE='A4.DVTX')
      OPEN(39,FILE='A4.X1X')
      OPEN(40,FILE='A4.X2X')
C
C Read input parameters
C
      READ(5,1003) V,D,R,TMAX,CO,CA
      READ(5,1003) XI,DX,XM,TI,DT,TM
C
C ISTEP and JSTEP control the printing interval of sensitivities when
C   a fine discretization is used to calculate time integrals. Set
C   to 1 for all other cases.
C IFLAG controls the set of file units into which output is written
C ILOG, when set to 1, transforms sensitivities to log form.
C
      READ(5,1004) ISTEP,JSTEP,IFLAG,ILOG
C
C
      D=D/R
      V=V/R
      IMAX=(XM+DX-XI)/DX
      JMAX=(TM+DT-TI)/DT
      E=0.0
C
C Begin calculations for each point in space (counted by I) and
C   time (counted by J).
C
      DO 4 J=1,JMAX
      TIME(J)=TI+(J-1)*DT
      DO 4 I=1,IMAX
      X(I)=XI+(I-1)*DX
      DO 2 M=1,2
      A=0.0
      DV=0.0
      DD=0.0
      DTM=0.0
      DDDX=0.0
      DVDX=0.0
      DXDX=0.0
      X1=0.0
      X2=0.0
      T=TIME(J)+(1-M)*TMAX
      IF(T.LE.0.) GO TO 2
C
C Common terms
C   (Note: overflow in exponents Q, CMA, and CPA will cause
```

```
C     program termination.)
C
      CM=(X(I)-V*T)/DSQRT(4.*D*T)
      CMA=DABS(CM)
      CP=(X(I)+V*T)/DSQRT(4.*D*T)
      CPA=DABS(CP)
      Q=V*X(I)/D
      DPI=0.5641895
      API=3.1415926
      B1=1.+Q+(T*(V**2.)/D)
      B2=1./DEXP(CMA**2.)
      B3=.5/DSQRT(D*T)
      B4=DEXP(Q)
      B5=V/D
      B6=.5*(X(I)-(V*T))/(D*T)
      AB6=DABS(B6)
      B7=DSQRT(T*(V**2.)/(API*D))
      B8=EXF(Q,CP)
      B9=1./DEXP(CPA**2.)
      B10=X(I)/D
      B11=.5*(X(I)+(V*T))/(D*T)
C
C Concentration
C
      A=0.5*(EXF(E,CM)+EXF(Q,CP))
C
C Sensitivities
C
      DV=DPI*B2*B3*T
      DV=DV+.5*B10*B8
      DV=DV-DPI*B4*B9*B3*T
C
      DD=DPI*B2*(.5/D)*CM
      DD=DD-.5*(Q/D)*B8
      DD=DD+B4*DPI*B9*(.5/D)*CP
C
      DTM=-DPI*B2*(V*B3+D*B6*B3)
      DTM=DTM+DPI*B4*B9*(V*B3-D*B11*B3)
C
C Integrands for time integral method (velocity sensitivity only)
C
      DDDX=-DPI*B2*B3
      DDDX=DDDX+.5*B5*B8
      DDDX=DDDX-B4*DPI*B9*B3
C
      DVDX=-DPI*B2*B6*T*B3
      DVDX=DVDX+(.5/D)*B8
      DVDX=DVDX+.5*B5*B10*B8
      DVDX=DVDX-B5*B4*DPI*B9*T*B3
      DVDX=DVDX-B10*B4*DPI*B9*B3
      DVDX=DVDX-B4*DPI*B9*(-B11*T)*B3
C
      DXDX=(-DPI)*B3*B2*(-B6)*B6*T
```

```
      DXDX=DXDX-DPI*B3*B2*(.5/D)
      DXDX=DXDX+(B5/D)*B8
      DXDX=DXDX+.5*(B5**2.)*B8*B10
      DXDX=DXDX-(B5**2.)*B4*DPI*T*B3*B9
      DXDX=DXDX-B10*B4*DPI*B9*B3*B5
      DXDX=DXDX-B4*DPI*B9*(-B11*T)*B3*B5
      DXDX=DXDX-(1./D)*B4*DPI*B9*B3
      DXDX=DXDX-(1./D)*B4*DPI*B9*B3
      DXDX=DXDX-B5*B4*DPI*B9*B3*B10
      DXDX=DXDX-B5*B4*DPI*B9*(-B11*T)*B3
      DXDX=DXDX-B10*B4*DPI*B9*(-B11)*B3
      DXDX=DXDX-B4*DPI*B9*(B11**2.)*B3*T
      DXDX=DXDX-B4*DPI*B9*(-.5/D)*B3
C
C Spatial derivatives
C
      X1=-DPI*B2*B3
      X1=X1+.5*B5*B8
      X1=X1-DPI*B4*B9*B3
C
      X2=DPI*B2*B6*B3
      X2=X2+.5*(B5**2.)*B8
      X2=X2-DPI*B5*B4*B3*B9
      X2=X2-DPI*B5*B4*B9*B3
      X2=X2+DPI*B4*B11*B3*B9
C
C Form complete expressions ( TO < TIME < TMAX )
C
      IF(M.EQ.2) GO TO 3
      CONC(I,J)=C0+(CA-C0)*A
      DCDV(I,J)=(CA-C0)*DV
      DCDD(I,J)=(CA-C0)*DD
      DCDCA(I,J)=A
      DCDC0(I,J)=1.-A
      DDX(I,J)=(CA-C0)*DDDX
      DVD(I,J)=(CA-C0)*DVDX*V
      DDD(I,J)=(CA-C0)*DXDX*D
      X1X(I,J)=(CA-C0)*X1
      X2X(I,J)=(CA-C0)*X2
    2 CONTINUE
C
C Form complete expressions ( TIME > TMAX )
C
    3 CONC(I,J)=CONC(I,J)-CA*A
      DCDV(I,J)=DCDV(I,J)-CA*DV
      DCDD(I,J)=DCDD(I,J)-CA*DD
      DCDCA(I,J)=DCDCA(I,J)-A
      DCDC0(I,J)=DCDC0(I,J)
      DTMAX(I,J)=-CA*DTM
      DDX(I,J)=DDX(I,J)-CA*DDDX
      DVD(I,J)=DVD(I,J)-CA*DVDX*V
      DDD(I,J)=DDD(I,J)-CA*DXDX*D
      X1X(I,J)=X1X(I,J)-CA*X1
```

```
      X2X(I,J)=X2X(I,J)-CA*X2
    4 CONTINUE
C
C Optional log transformation
C
      IF (ILOG.NE.1) GO TO 6
      DO 5 I=1,IMAX
      DO 5 J=1,JMAX
      IF (CONC(I,J).LT.1.D-50) CONC(I,J)=1.D-50
        DCDV(I,J)=DCDV(I,J)/CONC(I,J)
        DCDD(I,J)=DCDD(I,J)/CONC(I,J)
        DCDCA(I,J)=DCDCA(I,J)/CONC(I,J)
    5   CONC(I,J)=DLOG(CONC(I,J))
C
C Integration using the trapezoidal rule
C
    6 DO 8 I=1,IMAX
      DO 8 J=1,JMAX
      IF (J.NE.1) GO TO 7
      DXINT(I,J)=0.
      DVINT(I,J)=0.
      DDINT(I,J)=0.
      GO TO 8
    7   DXINT(I,J)=DXINT(I,J-1)+.5*(DDX(I,J-1)+DDX(I,J))*DT
        DVINT(I,J)=DVINT(I,J-1)+.5*(DVD(I,J-1)+DVD(I,J))*DT
        DDINT(I,J)=DDINT(I,J-1)+.5*(DDD(I,J-1)+DDD(I,J))*DT
        DCDXT(I,J)=DDINT(I,J)-DXINT(I,J)-DVINT(I,J)
    8 CONTINUE
C
C Print sensitivities and other terms
C
      CALL PLOTA4(IMAX,JMAX,ISTEP,JSTEP,IFLAG)
C
C
 1003 FORMAT(8F10.0)
 1004 FORMAT(4I5)
      STOP
      END
C
C
C
      SUBROUTINE PLOTA4(IMAX,JMAX,ISTEP,JSTEP,IFLAG)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /TXC/TIME(200),X(200),CONC(200,200)
      COMMON /DERIV1/DCDV(200,200),DCDD(200,200),DCDCA(200,200)
      COMMON /DERIV2/DTMAX(200,200),DCDC0(200,200)
      COMMON /INTEG1/DDX(200,200),DVD(200,200),DXINT(200,200)
      COMMON /INTEG2/DDD(200,200),DCDXT(200,200),DVINT(200,200)
      COMMON /INTEG3/DDINT(200,200)
      COMMON /SPACE/X1X(200,200),X2X(200,200)
      ISTEP1=1
      JSTEP1=1
      IF (ISTEP.NE.1) ISTEP1=ISTEP
```

```
      IF (JSTEP.NE.1) JSTEP1=JSTEP
      DO 10 K=6,17
   10   WRITE(K,1008)
      DO 20 K=29,40
   20   WRITE(K,1008)
      IF (IFLAG.GT.1) GO TO 90
      DO 30 I=ISTEP1,IMAX,ISTEP
      DO 40 K=6,17
   40   WRITE(K,1010)X(I)
        DO 30 J=JSTEP1,JMAX,JSTEP
          WRITE(6,1020)TIME(J),CONC(I,J)
          WRITE(7,1020)TIME(J),DCDV(I,J)
          WRITE(8,1020)TIME(J),DCDD(I,J)
          WRITE(9,1020)TIME(J),DCDCA(I,J)
          WRITE(10,1020)TIME(J),DCDCO(I,J)
          WRITE(11,1020)TIME(J),DTMAX(I,J)
          WRITE(12,1020)TIME(J),DXINT(I,J)
          WRITE(13,1020)TIME(J),DVINT(I,J)
          WRITE(14,1020)TIME(J),DDINT(I,J)
          WRITE(15,1020)TIME(J),DCDXT(I,J)
          WRITE(16,1020)TIME(J),X1X(I,J)
          WRITE(17,1020)TIME(J),X2X(I,J)
   30 CONTINUE
      DO 50 K=6,17
   50   WRITE(K,1009)
   90 IF (IFLAG.EQ.0) RETURN
      DO 60 J=JSTEP1,JMAX,JSTEP
      DO 70 K=29,40
   70   WRITE(K,1030)TIME(J)
        DO 60 I=ISTEP1,IMAX,ISTEP
          WRITE(29,1020)X(I),CONC(I,J)
          WRITE(30,1020)X(I),DCDV(I,J)
          WRITE(31,1020)X(I),DCDD(I,J)
          WRITE(32,1020)X(I),DCDCA(I,J)
          WRITE(33,1020)X(I),DCDCO(I,J)
          WRITE(34,1020)X(I),DTMAX(I,J)
          WRITE(35,1020)X(I),DXINT(I,J)
          WRITE(36,1020)X(I),DVINT(I,J)
          WRITE(37,1020)X(I),DDINT(I,J)
          WRITE(38,1020)X(I),DCDXT(I,J)
          WRITE(39,1020)X(I),X1X(I,J)
          WRITE(40,1020)X(I),X2X(I,J)
   60 CONTINUE
      DO 80 K=29,40
   80   WRITE(K,1009)
C
C
 1008 FORMAT('INPUT DATA.')
 1009 FORMAT('END OF DATA.')
 1010 FORMAT('"',F6.0,' METERS"')
 1020 FORMAT(F8.2,1X,E11.3E4)
 1030 FORMAT('"',F6.0,' DAYS"')
```

```
1050 FORMAT(2(F8.2,2X),E12.6E2)
     RETURN
     END
```

```
C
C      ******************************************************************
C      *                                                                *
C      *       ONE-DIMENSIONAL ADVECTION-DISPERSION EQUATION        B1  *
C      *                                                                *
C      *       THIRD-TYPE BOUNDARY CONDITION                            *
C      *       SEMI-INFINITE PROFILE                                    *
C      *       FIRST-ORDER DECAY (DMU)                                  *
C      *       LINEAR ADSORPTION (R)                                    *
C      *       CONSTANT INITIAL CONCENTRATION (CO)                      *
C      *       INPUT CONCENTRATION   = CA ( TO < TIME < TMAX )          *
C      *                             = 0  ( TIME > TMAX )               *
C      *                                                                *
C      ******************************************************************
C
      IMPLICIT REAL*8 (A-H,O-Z)
C
C Array dimensions are given as (X,TIME)
C
      COMMON /TXC/ TIME(200),X(200),CONC(200,200)
      COMMON /DERIV1/DCDV(200,200),DCDD(200,200),DCDCO(200,200)
      COMMON /DERIV2/DCDCA(200,200),DCDGAM(200,200),DCDMU(200,200)
      COMMON /DERIV3/DTMAX(200,200)
C
C Open input file
C
      OPEN(5,FILE='B1.DATA')
C
C Output in file units 6 to 13 is given as a function of time at
C   any number of locations in space.
C
      OPEN(6,FILE='B1.CONC')
      OPEN(7,FILE='B1.DV')
      OPEN(8,FILE='B1.DD')
      OPEN(9,FILE='B1.DCO')
      OPEN(10,FILE='B1.DCA')
      OPEN(11,FILE='B1.DGAMMA')
      OPEN(12,FILE='B1.DMU')
      OPEN(13,FILE='B1.DTMAX')
C
C Output in file unit 29 is given as a function of space at
C   any number of points in time.
C
      OPEN(29,FILE='B1.CONCX')
C
C Read input parameters
C
      READ(5,1003) V,D,R,TMAX,DMU,GAMMA,CO,CA
      READ(5,1003) XI,DX,XM,TI,DT,TM
C
C
      D=D/R
      V=V/R
```

```
      GAMMA=GAMMA/R
      DMU=DMU/R
      GD=GAMMA/(DMU**2.)
      U=V*DSQRT(1.+4.*DMU*D/V**2)
      DUV=(U/V)-4.*DMU*D/(U*V)
      DUD=2.*DMU/U
      DUMU=2.*D/U
      IMAX=(XM+DX-XI)/DX
      JMAX=(TM+DT-TI)/DT
      E=0.0
C
C Begin calculations for each point in space (counted by I) and
C   time (counted by J).
C
      DO 4 J=1,JMAX
      TIME(J)=TI+(J-1)*DT
      DO 4 I=1,IMAX
      X(I)=XI+(I-1)*DX
      DO 2 M=1,2
      B=0.
      DVB=0.
      DDB=0.
      DMB=0.
      DTM=0.
      T=TIME(J)+(1-M)*TMAX
      IF(T.LE.0.) GO TO 2
C
C Common terms
C
      P=V*X(I)/D
      S=DSQRT(4.*D*T)
      AM=0.5*(V-U)*X(I)/D
      AP=0.5*(V+U)*X(I)/D
      BM=(X(I)-U*T)/S
      BMA=DABS(BM)
      BP=(X(I)+U*T)/S
      CM=(X(I)-V*T)/S
      CMA=DABS(CM)
      CP=(X(I)+V*T)/S
C
      DPI=0.5641895
      API=3.1415926
      B1=1.+P+(T*(V**2.)/D)
      B2=1./DEXP((CMA)**2.)
      B3=.5/DSQRT(D*T)
      B4=DEXP(P)
      B5=V/D
      B6=.5*(X(I)-(V*T))/(D*T)
      AB6=DABS(B6)
      B7=DSQRT(T*(V**2.)/(API*D))
      B8=EXF(P,CP)
      B9=1./DEXP((CP)**2.)
      B10=X(I)/D
```

```
      B11=.5*(X(I)+(V*T))/(D*T)
C
      E1=EXF(AM,BM)
      E2=DEXP(AM)
      E3=1./DEXP(BMA**2.)
      E4=EXF(AP,BP)
      E5=1./(V-U)
      E5A=DABS(E5)
      E6=1./DEXP(BP**2.)
      E7=(V**2.)/(2.*DMU*D)
      E8=1./(V+U)
      E9=DEXP(-DMU*T)
      E10=DEXP(AP)
      E11=EXF(P-DMU*T,CP)
C
C Concentration: A(x,t) and B(x,t)
C
      A=DEXP(-DMU*T)*(1.-0.5*EXF(E,CM)-V*DSQRT(.3183099*T/D)*EXF(-CM*CM,
     1E)+0.5*(1.+P+V**2.*T/D)*EXF(P,CP))
C
      B=V/(V+U)*EXF(AM,BM)+V/(V-U)*EXF(AP,BP)+0.5*V*V/(DMU*D)*EXF(P-D
     1MU*T,CP)
C
C Sensitivities (derived from A(x,t))
C
      DVA=-DPI*B2*B3*T
      DVA=DVA-(1./B7)*(B5*T/API)*B2
      DVA=DVA-B7*B6*T*B2
      DVA=DVA+.5*(B10+(2.*B5*T))*B8
      DVA=DVA+.5*B1*B10*B8
      DVA=DVA-B1*B4*DPI*T*B3*B9
      DVA=DEXP(-DMU*T)*DVA
C
      DDA=-DPI*B2*B6*T*B3
      DDA=DDA+.5*(1./B7)*(B5**2.)*(T/API)*B2
      DDA=DDA-B7*(AB6**2.)*T*B2
      DDA=DDA-.5*(B5*B10+(B5**2.)*T)*B8
      DDA=DDA-.5*B1*(P/D)*B8
      DDA=DDA+B1*B4*DPI*B9*(.5/D)*CP
      DDA=DEXP(-DMU*T)*DDA
C
C Sensitivities (derived from B(x,t))
C
      DVB=(E8-V*(E8**2.)*(1.+DUV))*E1
      DVB=DVB+V*E8*(1.-DUV)*.5*B10*E1
      DVB=DVB+V*E8*E2*2.*DPI*E3*DUV*T*B3
      DVB=DVB+(E5-V*(E5A**2.)*(1.-DUV))*E4
      DVB=DVB+V*E5*(1.+DUV)*.5*B10*E4
      DVB=DVB-V*E5*E10*2.*DPI*DUV*T*B3*E6
      DVB=DVB+(B5/DMU)*E11
      DVB=DVB+E7*B10*E11
      DVB=DVB-E7*DEXP(P-DMU*T)*2.*DPI*B9*T*B3
C
```

```
      DDB=-V*(E8**2.)*DUD*E1
      DDB=DDB-V*E8*(DUD*.5*B10+(AM/D))*E1
      DDB=DDB+V*E8*E2*DPI*2.*E3*((.5*BM/D)+B3*T*DUD)
      DDB=DDB+V*(E5A**2.)*DUD*E4
      DDB=DDB-V*E5*((AP/D)-.5*B10*DUD)*E4
      DDB=DDB+V*E5*E10*2.*DPI*E6*((.5*BP/D)-B3*T*DUD)
      DDB=DDB-(E7/D)*E11
      DDB=DDB-E7*(P/D)*E11
      DDB=DDB+E7*DEXP(P-DMU*T)*DPI*B9*CP/D
C
      DMB=-V*(E8**2.)*DUMU*E1
      DMB=DMB-V*E8*.5*B10*DUMU*E1
      DMB=DMB+V*E8*E2*2.*DPI*E3*T*B3*DUMU
      DMB=DMB+V*(E5A**2.)*DUMU*E4
      DMB=DMB+V*E5*.5*B10*DUMU*E4
      DMB=DMB-V*E5*E10*2.*DPI*E6*T*B3*DUMU
      DMB=DMB-(E7/DMU)*E11
      DMB=DMB-E7*T*E11
C
      DTM=-2.*V*DPI*E8*E2*E3*(U*B3+2.*D*BM*(B3**2.))
      DTM=DTM+2.*V*DPI*E5*E10*E6*(U*B3+2.*D*BP*(B3**2.))
      DTM=DTM-.5*V*B5*E11
      DTM=DTM+2.*DPI*E7*(DEXP(P-DMU*T))*B9*(V*B3-D*B11*B3)
C
C Form complete expressions ( T0 < TIME < TMAX )
C
      IF(M.EQ.2) GO TO 3
      CONC(I,J)=GAMMA/DMU+(C0-GAMMA/DMU)*DEXP(-DMU*T)*(1.-0.5*
     1EXF(E,CM)-V*DSQRT(.3183099*T/D)*EXF(-CM*CM,E)+0.5*(1.+P+V**2*T/D)*
     2EXF(P,CP))+(CA-GAMMA/DMU)*B
      DCDV(I,J)=(C0-(GAMMA/DMU))*DVA+(CA-(GAMMA/DMU))*DVB
      DCDD(I,J)=(C0-(GAMMA/DMU))*DDA+(CA-(GAMMA/DMU))*DDB
      DCDC0(I,J)=A
      DCDCA(I,J)=B
      DCDGAM(I,J)=(1./DMU)-(1./DMU)*A-(1./DMU)*B
      DCDMU(I,J)=-GD+GD*A-(C0-GAMMA/DMU)*A*T+GD*B+(CA-GAMMA/DMU)*DMB
    2 CONTINUE
C
C Form complete expressions ( TIME > TMAX )
C
    3 CONC(I,J)=CONC(I,J)-CA*B
      DCDV(I,J)=DCDV(I,J)-CA*DVB
      DCDD(I,J)=DCDD(I,J)-CA*DDB
      DCDC0(I,J)=DCDC0(I,J)
      DCDCA(I,J)=DCDCA(I,J)-B
      DCDGAM(I,J)=DCDGAM(I,J)
      DCDMU(I,J)=DCDMU(I,J)-CA*DMB
      DTMAX(I,J)=-CA*DTM
    4 CONTINUE
C
C Print sensitivities
C
      CALL PLOTB1(IMAX,JMAX)
```

```
C
C
 1003 FORMAT(9F8.0)
      STOP
      END
C
C
C
      SUBROUTINE PLOTB1(IMAX,JMAX)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /TXC/TIME(200),X(200),CONC(200,200)
      COMMON /DERIV1/DCDV(200,200),DCDD(200,200),DCDCO(200,200)
      COMMON /DERIV2/DCDCA(200,200),DCDGAM(200,200),DCDMU(200,200)
      COMMON /DERIV3/DTMAX(200,200)
      DO 5 K=6,13
    5 WRITE(K,1008)
      WRITE(29,1008)
      DO 10 I=1,IMAX
      DO 15 K=6,13
   15   WRITE(K,1010)X(I)
        DO 10 J=1,JMAX
          WRITE(6,1020)TIME(J),CONC(I,J)
          WRITE(7,1020)TIME(J),DCDV(I,J)
          WRITE(8,1020)TIME(J),DCDD(I,J)
          WRITE(9,1020)TIME(J),DCDCO(I,J)
          WRITE(10,1020)TIME(J),DCDCA(I,J)
          WRITE(11,1020)TIME(J),DCDGAM(I,J)
          WRITE(12,1020)TIME(J),DCDMU(I,J)
          WRITE(13,1020)TIME(J),DTMAX(I,J)
   10 CONTINUE
      DO 20 J=1,JMAX
        WRITE(29,1030)TIME(J)
        DO 20 I=1,IMAX
          WRITE(29,1020)X(I),CONC(I,J)
  20  CONTINUE
      DO 30 K=6,13
  30    WRITE(K,1009)
      WRITE(29,1009)
C
C
 1008 FORMAT('INPUT DATA.')
 1009 FORMAT('END OF DATA.')
 1010 FORMAT('" ',F10.0,' METERS"')
 1020 FORMAT(F11.2,1X,E12.4E4)
 1030 FORMAT('" ',F10.0,' DAYS"')
 1050 FORMAT(2(F8.2,2X),E12.6E2)
      RETURN
      END
```

```
C
C     ******************************************************************
C     *                                                                *
C     *       ONE-DIMENSIONAL ADVECTION-DISPERSION EQUATION       B2  *
C     *                                                                *
C     *       THIRD-TYPE BOUNDARY CONDITION                            *
C     *       SEMI-INFINITE PROFILE                                    *
C     *       ZERO-ORDER PRODUCTION (GAMMA)                            *
C     *       FIRST-ORDER DECAY (DMU)                                  *
C     *       LINEAR ADSORPTION (R)                                    *
C     *       CONSTANT INITIAL CONCENTRATION (C0)                      *
C     *       INPUT CONCENTRATION = CB+CC*EXP(-DLAM*T)                 *
C     *                                                                *
C     ******************************************************************
C
      IMPLICIT REAL*8 (A-H,O-Z)
C
C Array dimensions are given as (X,TIME)
C
      COMMON /TXC/TIME(200),X(200),CONC(200,200)
      COMMON /DERIV1/DCDV(200,200),DCDD(200,200),DCDC0(200,200)
      COMMON /DERIV2/DCDCB(200,200),DCDCC(200,200),DCDGAM(200,200)
      COMMON /DERIV3/DCDMU(200,200),DCDLAM(200,200)
C
C Open input file
C
      OPEN(5,FILE='B2.DATA')
C
C Output file units 6 to 14 is given as a function of time at
C   any number of locations in space.
C
      OPEN(6,FILE='B2.CONC')
      OPEN(7,FILE='B2.DV')
      OPEN(8,FILE='B2.DD')
      OPEN(9,FILE='B2.DC0')
      OPEN(10,FILE='B2.DCB')
      OPEN(11,FILE='B2.DCC')
      OPEN(12,FILE='B2.DGAMMA')
      OPEN(13,FILE='B2.DMU')
      OPEN(14,FILE='B2.DLAM')
C
C Output in file unit 29 is given as a function of space at
C   any number of points in time.
C
      OPEN(29,FILE='B2.CONCX')
C
C Read input parameters
C
      READ(5,1003) V,D,R,GAMMA,DMU,DLAM,C0,CB,CC
      READ(5,1003) XI,DX,XM,TI,DT,TM
C
C
      D=D/R
```

```
      V=V/R
      GAMMA=GAMMA/R
      DMU=DMU/R
      GD=GAMMA/(DMU**2.)
      DZD=GAMMA/DMU
      U=DSQRT(V**2+4.*DMU*D)
      DUV=(U/V)-4.*DMU*D/(U*V)
      DUD=2.*DMU/U
      DUMU=2.*D/U
      S=V**2+4.*(DMU-DLAM)*D
      IF(S.LE.0.) GO TO 5
      W=DSQRT(S)
      DWV=(W/V)-4.*D*(DMU-DLAM)/(V*W)
      DWD=2.*(DMU-DLAM)/W
      DWMU=2.*D/W
      DWLAM=-2.*D/W
      IMAX=(XM+DX-XI)/DX
      JMAX=(TM+DT-TI)/DT
C
C Begin calculations for each point in space (counted by I) and
C   time (counted by J).
C
      DO 4 J=1,JMAX
      T=TI+(J-1)*DT
      TIME(J)=T
      DO 4 I=1,IMAX
      X(I)=XI+(I-1)*DX
C
C Common terms
C
      P=V*X(I)/D
      S=DSQRT(4.*D*T)
      AM=(X(I)-V*T)/S
      AMA=DABS(AM)
      AP=(X(I)+V*T)/S
      BM=(X(I)-U*T)/S
      BMA=DABS(BM)
      BP=(X(I)+U*T)/S
      DM=0.5*(V-U)*X(I)/D
      DP=0.5*(V+U)*X(I)/D
C
      DPI=0.5641895
      API=3.1415926
      B1=1.+P+(T*(V**2.)/D)
      B2=1./DEXP((AMA)**2.)
      B3=.5/DSQRT(D*T)
      B4=DEXP(P)
      B5=V/D
      B6=.5*(X(I)-(V*T))/(D*T)
      AB6=DABS(B6)
      B7=DSQRT(T*(V**2.)/(API*D))
      B8=EXF(P,AP)
      B9=1./DEXP((AP)**2.)
```

```
      B10=X(I)/D
      B11=.5*(X(I)+(V*T))/(D*T)
C
      E1=EXF(DM,BM)
      E2=DEXP(DM)
      E3=1./DEXP(BMA**2.)
      E4=EXF(DP,BP)
      E5=1./(V-U)
      E5A=DABS(E5)
      E6=1./DEXP(BP**2.)
      E7=(V**2.)/(2.*DMU*D)
      E8=1./(V+U)
      E9=DEXP(-DMU*T)
      E10=DEXP(DP)
      E11=EXF(P-DMU*T,AP)
C
C Concentration: A(x,t) and B(x,t)
C
      A=DEXP(-DMU*T)*(1.-0.5*EXF(0.D0,AM)-V*DSQRT(.3183099*T/D)*
     1EXF(-AM*AM,0.D0)+0.5*(1.+P+V**2*T/D)*EXF(P,AP))
C
      B=V/(V+U)*EXF(DM,BM)+V/(V-U)*EXF(DP,BP)+0.5*V**2/(DMU*D)*
     1EXF(P-DMU*T,AP)
C
      CONC(I,J)=DZD+(CO-CC-DZD)*A+(CB-DZD)*B+CC*DEXP(-DLAM*T)
C
C Sensitivities (derived from A(x,t)
C
      DVA=-DPI*B2*B3*T
      DVA=DVA-(1./B7)*(B5*T/API)*B2
      DVA=DVA-B7*B6*T*B2
      DVA=DVA+.5*(B10+(2.*B5*T))*B8
      DVA=DVA+.5*B1*B10*B8
      DVA=DVA-B1*B4*DPI*T*B3*B9
      DVA=DEXP(-DMU*T)*DVA
C
      DDA=-DPI*B2*B6*T*B3
      DDA=DDA+.5*(1./B7)*(B5**2.)*(T/API)*B2
      DDA=DDA-B7*(AB6**2.)*T*B2
      DDA=DDA-.5*(B5*B10+(B5**2.)*T)*B8
      DDA=DDA-.5*B1*(P/D)*B8
      DDA=DDA+B1*B4*DPI*B9*(.5/D)*AP
      DDA=DEXP(-DMU*T)*DDA
C
C Sensitivities (derived from B(x,t))
C
      DVB=(E8-V*(E8**2.)*(1.+DUV))*E1
      DVB=DVB+V*E8*(1.-DUV)*.5*B10*E1
      DVB=DVB+V*E8*E2*2.*DPI*E3*DUV*T*B3
      DVB=DVB+(E5-V*(E5A**2.)*(1.-DUV))*E4
      DVB=DVB+V*E5*(1.+DUV)*.5*B10*E4
      DVB=DVB-V*E5*E10*2.*DPI*DUV*T*B3*E6
      DVB=DVB+(B5/DMU)*E11
```

```
      DVB=DVB+E7*B10*E11
      DVB=DVB-E7*DEXP(P-DMU*T)*2.*DPI*B9*T*B3
C
      DDB=-V*(E8**2.)*DUD*E1
      DDB=DDB-V*E8*(DUD*.5*B10+(DM/D))*E1
      DDB=DDB+V*E8*E2*DPI*2.*E3*((.5*BM/D)+B3*T*DUD)
      DDB=DDB+V*(E5A**2.)*DUD*E4
      DDB=DDB-V*E5*((DP/D)-.5*B10*DUD)*E4
      DDB=DDB+V*E5*E10*2.*DPI*E6*((.5*BP/D)-B3*T*DUD)
      DDB=DDB-(E7/D)*E11
      DDB=DDB-E7*(P/D)*E11
      DDB=DDB+E7*DEXP(P-DMU*T)*DPI*B9*AP/D
C
      DMB=-V*(E8**2.)*DUMU*E1
      DMB=DMB-V*E8*.5*B10*DUMU*E1
      DMB=DMB+V*E8*E2*2.*DPI*E3*T*B3*DUMU
      DMB=DMB+V*(E5A**2.)*DUMU*E4
      DMB=DMB+V*E5*.5*B10*DUMU*E4
      DMB=DMB-V*E5*E10*2.*DPI*E6*T*B3*DUMU
      DMB=DMB-(E7/DMU)*E11
      DMB=DMB-E7*T*E11
C
C Form complete sensitivity expressions ( DMU = DLAM )
C
      DCDV(I,J)=(CO-CC-DZD)*DVA+(CB-DZD)*DVB
      DCDD(I,J)=(CO-CC-DZD)*DDA+(CB-DZD)*DDB
      DCDCO(I,J)=A
      DCDCB(I,J)=B
      DCDCC(I,J)=-A+DEXP(-DLAM*T)
      DCDGAM(I,J)=(1./DMU)*(1.-A-B)
      DCDMU(I,J)=-GD*(1.-A-B)-(CO-CC-DZD)*T*A+(CB-DZD)*DMB
      DCDLAM(I,J)=-T*CC*DEXP(-DLAM*T)
C
      IF(DMU.EQ.DLAM) GO TO 4
C
C Common terms, continued
C
      CM=(X(I)-W*T)/S
      CMA=DABS(CM)
      CP=(X(I)+W*T)/S
      EM=0.5*(V-W)*X(I)/D
      EP=0.5*(V+W)*X(I)/D
C
      F1=EXF(EM,CM)
      F2=DEXP(EM)
      F3=1./DEXP(CMA**2.)
      F4=EXF(EP,CP)
      F5=1./(V-W)
      F5A=DABS(F5)
      F6=1./DEXP(CP**2.)
      F7=(V**2.)/(2.*D*(DMU-DLAM))
      F8=1./(V+W)
      F9=DEXP(-DLAM*T)
```

```
      F10=DEXP(EP)
C
C Concentration: E(x,t)
C
      E=DEXP(-DLAM*T)*(V/(V+W)*EXF(EM,CM)+V/(V-W)*EXF(EP,CP))+0.5*V**2
     1/(D*(DMU-DLAM))*EXF(P-DMU*T,AP)
C
C Sensitivities (derived from E(x,t))
C
      DVE=F9*(F8-V*(F8**2.)*(1.+DWV))*F1
      DVE=DVE+F9*V*F8*(1.-DWV)*.5*B10*F1
      DVE=DVE+F9*V*F8*F2*2.*DPI*F3*DWV*T*B3
      DVE=DVE+F9*(F5-V*(F5A**2.)*(1.-DWV))*F4
      DVE=DVE+F9*V*F5*(1.+DWV)*.5*B10*F4
      DVE=DVE-F9*V*F5*F10*2.*DPI*DWV*T*B3*F6
      DVE=DVE+(V/(D*(DMU-DLAM)))*E11
      DVE=DVE+F7*B10*E11
      DVE=DVE-F7*DEXP(P-DMU*T)*2.*DPI*B9*T*B3
C
      DDE=-F9*V*(F8**2.)*DWD*F1
      DDE=DDE-F9*V*F8*(DWD*.5*B10+(EM/D))*F1
      DDE=DDE+F9*V*F8*F2*DPI*2.*F3*((.5*CM/D)+B3*T*DWD)
      DDE=DDE+F9*V*(F5A**2.)*DWD*F4
      DDE=DDE-F9*V*F5*((EP/D)-.5*B10*DWD)*F4
      DDE=DDE+F9*V*F5*F10*2.*DPI*F6*((.5*CP/D)-B3*T*DWD)
      DDE=DDE-(F7/D)*E11
      DDE=DDE-F7*(P/D)*E11
      DDE=DDE+F7*DEXP(P-DMU*T)*DPI*B9*AP/D
C
      DL=-T*DEXP(-DLAM*T)*(V/(V+W)*EXF(EM,CM)+V/(V-W)*EXF(EP,CP))
      DL=DL-F9*V*(F8**2.)*DWLAM*F1
      DL=DL-F9*V*F8*.5*B10*DWLAM*F1
      DL=DL+F9*V*F8*F2*2.*DPI*F3*T*B3*DWLAM
      DL=DL+F9*V*(F5A**2.)*DWLAM*F4
      DL=DL+F9*V*F5*.5*B10*DWLAM*F4
      DL=DL-F9*V*F5*F10*2.*DPI*F6*T*B3*DWLAM
      DL=DL+(F7/(DMU-DLAM))*E11
C
      DME=-V*(F8**2.)*DWMU*F1
      DME=DME-V*F8*.5*B10*DWMU*F1
      DME=DME+V*F8*F2*2.*DPI*F3*T*B3*DWMU
      DME=DME+V*(F5A**2.)*DWMU*F4
      DME=DME+V*F5*.5*B10*DWMU*F4
      DME=DME-V*F5*F10*2.*DPI*F6*T*B3*DWMU
      DME=DME*F9
      DME=DME-(F7/(DMU-DLAM))*E11
      DME=DME-F7*T*E11
C
C Form complete expressions ( DMU not equal to DLAM )
C
      CONC(I,J)=DZD+(CO-DZD)*A+(CB-DZD)*B+CC*E
      DCDV(I,J)=(CO-DZD)*DVA+(CB-DZD)*DVB+CC*DVE
      DCDD(I,J)=(CO-DZD)*DDA+(CB-DZD)*DDB+CC*DDE
```

```
      DCDCO(I,J)=A
      DCDCB(I,J)=B
      DCDCC(I,J)=E
      DCDGAM(I,J)=(1./DMU)*(1.-A-B)
      DCDMU(I,J)=-GD*(1.-A-B)-(CO-DZD)*T*A+(CB-DZD)*DMB+CC*DME
      DCDLAM(I,J)=CC*DL
    4 CONTINUE
C
C Print sensitivities
C
      CALL PLOTB2(IMAX,JMAX)
C
C
      STOP
    5 WRITE(6,1007)
 1003 FORMAT(10F8.0)
 1007 FORMAT(///5X,6(1H*),' DLAM IS TOO LARGE, THIS CASE NOT EXECUTED',6
     1(1H*))
      STOP
      END
C
C
C
      SUBROUTINE PLOTB2(IMAX,JMAX)
      IMPLICIT REAL*8 (A-H,O-Z)
      COMMON /TXC/TIME(200),X(200),CONC(200,200)
      COMMON /DERIV1/DCDV(200,200),DCDD(200,200),DCDCO(200,200)
      COMMON /DERIV2/DCDCB(200,200),DCDCC(200,200),DCDGAM(200,200)
      COMMON /DERIV3/DCDMU(200,200),DCDLAM(200,200)
      DO 5 K=6,14
    5 WRITE(K,1008)
      WRITE(29,1008)
      DO 10 I=1,IMAX
      DO 15 K=6,14
   15   WRITE(K,1010)X(I)
        DO 10 J=1,JMAX
          WRITE(6,1020)TIME(J),CONC(I,J)
          WRITE(7,1020)TIME(J),DCDV(I,J)
          WRITE(8,1020)TIME(J),DCDD(I,J)
          WRITE(9,1020)TIME(J),DCDCO(I,J)
          WRITE(10,1020)TIME(J),DCDCB(I,J)
          WRITE(11,1020)TIME(J),DCDCC(I,J)
          WRITE(12,1020)TIME(J),DCDGAM(I,J)
          WRITE(13,1020)TIME(J),DCDMU(I,J)
          WRITE(14,1020)TIME(J),DCDLAM(I,J)
   10 CONTINUE
      DO 20 J=1,JMAX
        WRITE(29,1030)TIME(J)
        DO 20 I=1,IMAX
          WRITE(29,1020)X(I),CONC(I,J)
 20   CONTINUE
      DO 30 K=6,14
```

```
 30    WRITE(K,1009)
      WRITE(29,1009)
C
C
 1008 FORMAT('INPUT DATA.')
 1009 FORMAT('END OF DATA.')
 1010 FORMAT('" ',F10.0,' METERS"')
 1020 FORMAT(F11.2,1X,E12.4E4)
 1030 FORMAT('" ',F10.0,' DAYS"')
 1050 FORMAT(2(F8.2,2X),E12.6E2)
      RETURN
      END
```

```
C
C
C     ************************************************************************
C     *                                                                      *
C     *       ONE-DIMENSIONAL ADVECTION-DISPERSION EQUATION        C         *
C     *                                                                      *
C     *       SOLUTE IN MULTIPLE LAYERS MIXED IN OBSERVATION WELL            *
C     *       MODEL A1 REPRESENTS TRANSPORT IN EACH LAYER                    *
C     *                                                                      *
C     *       SEMI-INFINITE PROFILE                                          *
C     *       THIRD TYPE BOUNDARY CONDITION                                  *
C     *       NO PRODUCTION OR DECAY                                         *
C     *       LINEAR ADSORPTION (R)                                          *
C     *       CONSTANT INITIAL CONCENTRATION (CO)                            *
C     *       INPUT CONCENTRATION   = CA ( TO < TIME < TMAX )                *
C     *                             = 0  ( TIME > TMAX )                     *
C     *                                                                      *
C     ************************************************************************
C
      IMPLICIT REAL*8 (A-H,O-Z)
C
C Array dimensions are given as (X,TIME)
C
      COMMON /TXC/TIME(200),X(200)
      COMMON /CONC/CONC(200,200,5),CBAR(200,200)
      COMMON /DERIV1/DCDV(200,200,5)
      COMMON /DERIV2/DCDD(200,200,5)
      COMMON /DERIV3/DCDCA(200,200,5)
      COMMON /DERIV4/DTMAX(200,200,5)
      COMMON /DERIV5/DCDC0(200,200,5)
      COMMON /DERIV6/DCDF(200,200,5)
C
C Open input file
C
      OPEN(5,FILE='C.DATA')
C
C Output in file units 6 to 13 is given as a function of time at
C   any number of locations in space.
C
      OPEN(6,FILE='C.CONC')
      OPEN(7,FILE='C.DV')
      OPEN(8,FILE='C.DD')
      OPEN(9,FILE='C.DCA')
      OPEN(10,FILE='C.DCO')
      OPEN(11,FILE='C.DTMAX')
      OPEN(12,FILE='C.DCF')
      OPEN(13,FILE='C.CBAR')
C
C Output in file units 29 and 30 is given as a function of space at
C   any number of points in time.
C
      OPEN(29,FILE='C.CONCX')
      OPEN(30,FILE='C.CBARX')
```

```
C
C Read input parameters
C   NLAY is the number of layers in the medium
C   ILOG, when set to 1, transforms sensitivities to log form.
C
      READ(5,1004) NLAY,ILOG
      READ(5,1003) XI,DX,XM,TI,DT,TM
C
      IMAX=(XM+DX-XI)/DX
      JMAX=(TM+DT-TI)/DT
      DO 90 I=1,IMAX
      DO 90 J=1,JMAX
   90   CBAR(I,J)=0.
C
      DO 100 N=1,NLAY
      READ(5,1003) V,D,R,TMAX,C0,CA
      READ(5,1005) F
C
C
      D=D/R
      V=V/R
      E=0.0
C
C Begin calculations for each point in space (counted by I) and
C   time (counted by J).
C
      DO 4 J=1,JMAX
      TIME(J)=(TI+(J-1)*DT)
      DO 4 I=1,IMAX
      X(I)=XI+(I-1)*DX
      DO 2 M=1,2
      A=0.0
      DV=0.0
      DD=0.0
      DTM=0.0
      T=TIME(J)+((1-M)*TMAX)
      IF(T.LE.0.) GO TO 2
C
C Common terms
C   (Note: overflow in exponents Q,CMA, and CPA will cause
C     program termination.)
C
      CM=(X(I)-V*T)/DSQRT(4.*D*T)
      CMA=DABS(CM)
      CP=(X(I)+V*T)/DSQRT(4.*D*T)
      CPA=DABS(CP)
      Q=V*X(I)/D
      DPI=0.5641895
      API=3.1415926
      B1=1.+Q+(T*(V**2.)/D)
      B2=1./DEXP(CMA**2.)
      B3=.5/DSQRT(D*T)
      B4=DEXP(Q)
```

```
      B5=V/D
      B6=.5*(X(I)-(V*T))/(D*T)
      AB6=DABS(B6)
      B7=DSQRT(T*(V**2.)/(API*D))
      B8=EXF(Q,CP)
      B9=1./DEXP(CPA**2.)
      B10=X(I)/D
      B11=.5*(X(I)+(V*T))/(D*T)
C
C Concentration:  A(x,t)
C
      A=0.5*EXF(E,CM)+V*DSQRT(.3183099*T/D)*EXF(-CM*CM,E)-0.5*(1.+Q+V*V
     1*T/D)*EXF(Q,CP)
C
C Sensitivities
C
      DV=DPI*B2*B3*T
      DV=DV+(1./B7)*(B5*T/API)*B2
      DV=DV+B7*B6*T*B2
      DV=DV-.5*(B10+(2.*B5*T))*B8
      DV=DV-.5*B1*B10*B8
      DV=DV+B1*B4*DPI*T*B3*B9
C
      DD=DPI*B2*B6*T*B3
      DD=DD-.5*(1./B7)*(B5**2.)*(T/API)*B2
      DD=DD+B7*(AB6**2.)*T*B2
      DD=DD+.5*(B5*B10+(B5**2.)*T)*B8
      DD=DD+.5*B1*(Q/D)*B8
      DD=DD+B1*B4*DPI*B9*(.5/D)*(-CP)
C
      DTM=-DPI*B2*(V*B3+D*B6*B3)
      DTM=DTM-.5*(V/API)*B5*B2/B7
      DTM=DTM-B7*B2*(V*B6+D*(AB6**2.))
      DTM=DTM+.5*V*B5*B8
      DTM=DTM+DPI*B1*B4*B9*(-V*B3+D*B11*B3)
C
C Form complete expressions ( TO < TIME < TMAX )
C
      IF(M.EQ.2) GO TO 3
      CONC(I,J,N)=C0+(CA-C0)*A
      DCDV(I,J,N)=(CA-C0)*DV
      DCDD(I,J,N)=(CA-C0)*DD
      DCDCA(I,J,N)=A
      DCDC0(I,J,N)=1.-A
      DTMAX(I,J,N)=0.
    2 CONTINUE
C
C Form complete expressions (TIME > TMAX )
C
    3 CONC(I,J,N)=CONC(I,J,N)-CA*A
      DCDV(I,J,N)=DCDV(I,J,N)-CA*DV
      DCDD(I,J,N)=DCDD(I,J,N)-CA*DD
```

```
      DCDCA(I,J,N)=DCDCA(I,J,N)-A
      DCDCO(I,J,N)=DCDCO(I,J,N)
      DTMAX(I,J,N)=-CA*DTM
    4 CONTINUE
C
C Account for fractional contribution (F)
C
      DO 5 I=1,IMAX
      DO 5 J=1,JMAX
        CBAR(I,J)=CBAR(I,J)+F*CONC(I,J,N)
        DCDV(I,J,N)=F*DCDV(I,J,N)
        DCDD(I,J,N)=F*DCDD(I,J,N)
        DCDCA(I,J,N)=F*DCDCA(I,J,N)
        DCDCO(I,J,N)=F*DCDCO(I,J,N)
        DTMAX(I,J,N)=F*DTMAX(I,J,N)
        DCDF(I,J,N)=CONC(I,J,N)
    5 CONTINUE
C
  100 CONTINUE
C
C Optional log transformation
C
      IF (ILOG.NE.1) GO TO 8
      DO 7 I=1,IMAX
      DO 7 J=1,JMAX
      DO 6 N=1,NLAY
      IF (CBAR(I,J).LT.1.D-50) CBAR(I,J)=1.D-50
        DCDV(I,J,N)=DCDV(I,J,N)/CBAR(I,J)
        DCDD(I,J,N)=DCDD(I,J,N)/CBAR(I,J)
        DCDCA(I,J,N)=DCDCA(I,J,N)/CBAR(I,J)
        DCDCO(I,J,N)=DCDCO(I,J,N)/CBAR(I,J)
        DTMAX(I,J,N)=DTMAX(I,J,N)/CBAR(I,J)
        DCDF(I,J,N)=CONC(I,J,N)/CBAR(I,J)
    6   CONTINUE
      CBAR(I,J)=DLOG(CBAR(I,J))
    7 CONTINUE
C
C Print sensitivities and other terms
C
    8 CALL PLOTC(IMAX,JMAX,NLAY)
C
C
 1003 FORMAT(8F10.0)
 1004 FORMAT(4I5)
 1005 FORMAT(F10.0)
      STOP
      END
C
C
C
      SUBROUTINE PLOTC(IMAX,JMAX,NLAY)
      IMPLICIT REAL*8 (A-H,O-Z)
```

```
      COMMON /TXC/TIME(200),X(200)
      COMMON /CONC/CONC(200,200,5),CBAR(200,200)
      COMMON /DERIV1/DCDV(200,200,5)
      COMMON /DERIV2/DCDD(200,200,5)
      COMMON /DERIV3/DCDCA(200,200,5)
      COMMON /DERIV4/DTMAX(200,200,5)
      COMMON /DERIV5/DCDC0(200,200,5)
      COMMON /DERIV6/DCDF(200,200,5)
      DO 10 K=6,13
   10   WRITE(K,1008)
      WRITE(29,1008)
      WRITE(30,1008)
      DO 30 N=1,NLAY
      DO 30 I=1,IMAX
        DO 40 K=6,12
   40   WRITE(K,1010)N,X(I)
        IF (N.EQ.NLAY) WRITE(13,1015)X(I)
        DO 30 J=1,JMAX
          WRITE(6,1020)TIME(J),CONC(I,J,N)
          WRITE(7,1020)TIME(J),DCDV(I,J,N)
          WRITE(8,1020)TIME(J),DCDD(I,J,N)
          WRITE(9,1020)TIME(J),DCDCA(I,J,N)
          WRITE(10,1020)TIME(J),DCDC0(I,J,N)
          WRITE(11,1020)TIME(J),DTMAX(I,J,N)
          WRITE(12,1020)TIME(J),DCDF(I,J,N)
          IF (N.EQ.NLAY) WRITE(13,1020)TIME(J),CBAR(I,J)
   30 CONTINUE
      DO 50 K=6,13
   50   WRITE(K,1009)
C
      DO 60 N=1,NLAY
      DO 60 J=1,JMAX
        WRITE(29,1030)N,TIME(J)
        IF (N.EQ.NLAY) WRITE(30,1040) TIME(J)
        DO 60 I=1,IMAX
          WRITE(29,1020)X(I),CONC(I,J,N)
          IF (N.EQ.NLAY) WRITE(30,1020)X(I),CBAR(I,J)
   60 CONTINUE
      WRITE(29,1009)
      WRITE(30,1009)
C
C
 1008 FORMAT('INPUT DATA.')
 1009 FORMAT('END OF DATA.')
 1010 FORMAT('"LAYER',I2,': DISTANCE=',F6.0,'"')
 1015 FORMAT('"DISTANCE=',F6.0,'"')
 1020 FORMAT(F11.2,1X,E11.3E4)
 1030 FORMAT('"LAYER',I2,': TIME=',F11.0,'"')
 1040 FORMAT('"TIME=',F11.0,'"')
 1050 FORMAT(2(F8.2,2X),E12.6E2)
      RETURN
      END
```

```
C*****************************************************************
C*                                                               *
C*                         Function EXF                          *
C*                                                               *
C*****************************************************************
C
      FUNCTION EXF(A,B)
C
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
```