UNIVERSITY OF CINCINNATI

Date:__04/18/2007_____

I, __Siladitya Ray Chaudhuri _____,

hereby submit this work as part of the requirements for the degree of:

Doctor of Philosophy

in:

Chemical Engineering

It is entitled:

Mathematical Modeling of Percutaneous Absorption Of Volatile Compounds following Transient Liquid-Phase Exposures

This work and its defense approved by:

Chair: **_ Prof. William B. Krantz _____**
_ Prof. Gerald B. Kasting _____
_ Prof. Sun-Tak Hwang _____
_ Dr. Chia Chi Ho _____

MATHEMATICAL MODELING OF PERCUTANEOUS ABSORPTION OF VOLATILE SOLVENTS FOLLOWING TRANSIENT LIQUID-PHASE EXPOSURES

A dissertation submitted to the

Division of Research and Advanced Studies
of the University of Cincinnati

in partial fulfillment of the
requirements for the degree of

DOCTOR OF PHILOSOPHY

in the Division of Chemical Engineering
of the College of Engineering

2007

by

Siladitya Ray Chaudhuri

M.S. in Chemical Engineering
University of Cincinnati, Cincinnati, OH, 2003

Committee Chair: W. B. Krantz, PhD

Abstract

Skin is the largest organ of the human body and due to its immediate proximity to the environment is often exposed to a variety of extraneous chemicals. A proper study of percutaneous absorption is critical to the risk assessment pertaining to variety of exposure scenarios (occupational and environmental) and also for the development of transdermal products (cosmetic and therapeutic). The number of chemicals to which the human population can be exposed is extremely high such that only a small fraction can be studied experimentally. Thus, computational models for dermal absorption used in lieu of animal experiments to estimate absorption of new ingredients are used widely in many industries. The exposure scenario for volatile chemicals is further complicated due to a vigorous evaporation that influences its tendency to penetrate.

This dissertation discusses the development of a mathematical model for predicting skin permeability of volatile compounds using the fundamental principles of transport phenomena and thermodynamics. The key features of the model include treatment of the moving boundary (finite dose volumes), treatment of multilayered problems, convection due to density gradients (binary and multicomponent systems), etc. and ascertaining the system parameters using information obtained from the intricate skin microstructure and mechanisms of permeation. These models are then validated through comparing the respective simulation results with corresponding experimental data on the skin permeation fluxes of pure ethanol and benzene. Optimal values of sensitive parameters are obtained through extensive non-linear regression analysis and compared with the original predictive values. The results show appreciable correlation between the model predictions and the experimental data (fraction of dose absorbed). It was also clear that the disposition characteristics of low-molecular weight volatile liquids are

critically dependent on parameters such as the fractional deposition depth and skin diffusivity. Also, in order to mimic the behavior of penetrating compounds from cosmetic and pharmaceutical formulations, a binary mass-transfer model is developed. This is then validated by comparing the simulation results with experimental skin permeation data on benzyl alcohol (BA) from a dilute ethanolic solution. The fraction absorbed for BA is not independent of the fractional deposition depth but is sensitive to skin diffusivity.

Dedication

I dedicate this thesis to my life in Cincinnati. This city helped me initiate into my truly adult life and prepared me for entering the real world in this country. It has been there for me consistently, when others have either departed or deserted. I would not have been able to write this page if it was not for the beautiful nook that my wife and I had built in Morgens 502 and the heavenly view from its balcony, the few friends with whom I would spend hours talking about the most nonsensical things almost oblivious to the real world that lay ahead, the nice and serene UC campus which transformed right in front of my eyes from a traditional Midwestern taste to a very urban style, for the Langsam Library which provided me with unlimited number of books, movies and documentaries (something that gave tough competition to my wife's cooking as sustenance that I would crave for), the few Indian restaurants where I would go keep going back filling myself up with the exact same spread in the exact same buffet, the aimless long drives through unknown state routes that went right through the greenery of the Buckeye state and its neighbors, the small villages with their cookie shops and antique stores that seemed completely unaware of the presence of a world outside their borders, the beauty of the vast green meadows of Ohio, the endless cornfields of Indiana and the beautiful horse farms in Kentucky with their perfectly painted white fence, the AMC movie theaters and the Barnes & Noble bookstores on the Levee from where I would gaze at the waters of the Ohio river as it glistened during the night as the bright light form the downtown building shun brightly upon it (almost masquerading its sleepy and deserted reality). These are a few of the wonderful things that I have had the pleasure of filling my life with. These helped me ease the ordeal of graduate school, with its various pressures, uncertainties and insecurities. Ever since I opened my eyes in the US, you are the only one whom I have known and I will never forget you and this dissertation is for you.

Ackowledgements

The credit for this dissertation wholly belongs to Prof. Jerry Kasting. He picked me up after my Masters degree when I was wandering with no PhD project in hand and trying to skin my teeth into a suitable and interesting project. He stuck with me during my toughest times that were socially and emotionally challenging. He held his patience against the relentless administrative mess and unfriendly policies put forth that deterred the students form pursuing research wholeheartedly. I sincerely thank you for making my doctoral degree a wonderful learning experience with not only the technical content but also the philosophical aspects. You taught me to be sincere, focused and diligent and equipped me with the right training and expertise that will help me further my. If you had not believed in me from the very beginning, I would not be where I am today.

I would also like to thank Professor William B. Krantz for sticking with me, even though he was not required to. I thank you for believing in me. It was a roller-coaster ride for the both of us and I am glad that we held out until we reached the finish line. I will always get inspired by your enthusiasm towards teaching and research and your analytical ability.

I would also like to thank Prof. Sun-Tak Hwang for his active participation in my research project. You were always there for me as a guide and helped me become a good researcher. Your professionalism and ethics still inspire me. I will always remember our stimulating discussions on mass transfer, thermodynamics, science and its history and philosophy.

I would also like to thank Dr. Chia Chi Ho for agreeing to be on my committee. Your support and kind words will always be remembered.

I would also like to thank my colleague and friend, Dr. Hayong Lee. You and I were soldiers that went to the same hopeless battle and surprisingly emerged victorious. I have learnt a lot from you. I miss the 671 ERC days of spending hours on the board discussing equations and scaling analysis. It was great to have you as a coworker. I would also like to thank another friend and colleague, Mr. Shaun Howard. Shaun, I still miss having you as a colleague and a friend. Our long conversations helped me grow my understanding of the people and their culture in this country. If it were not for you and your impeccable computational support, it would have taken me ages to complete this project. Thanks Hanyong and Shaun!

I would also like to thank Mr. Mathhew Miller for helping me out on several occasions during my research. I would also like to thank Dr. Varsha Bhatt and Rachna Gajjar in the College of Pharmacy for providing valuable experimental data that became the core component of my analysis.

I would like to thank the Department of Chemical Engineering for giving me admission into their graduate program and the University of Cincinnati for approving my admission and financial aid, without which all of this would have been a far-fetched dream. I would also like to thank Graduate Student Governance Association at the University of Cincinnati for providing me with the much needed travel funds. Without the excellent computational support and library facilities, it would have been a lot more difficult and I sincerely appreciate the high quality of these resources at UC.

There were several people who were directly or indirectly involved in contributing towards making my journey smooth and comfortable and I wholeheartedly thank them for their love and support.

Table of Contents

Chapter

List of Tables

List of Figures

CHAPTER 1

INTRODUCTION

1.1 Permeation (Mass Transport) through Human Skin: Relevance and Importance

Although all life forms depend upon membrane–like structures for their existence, nowhere is that more dramatically demonstrated as in the skin (Potts and Francoeur, 1991). This is particularly true in higher animals, in which this complex membrane has evolved to a multifunctional organ. Skin is the largest organ of the human body, covering a surface area of approximately 2 m^2 and receiving about one-third of the blood circulating throughout the body (Singh and Singh, 1993). The skin mainly serves:

- As a protective barrier against chemicals, microorganisms like bacteria, fungi etc., allergens, impact with mechanical objects, UV radiation etc.
- A regulatory role in maintaining homeostasis of the body, especially in terms of composition, heat regulation and temperature, blood pressure control and excretion
- A sensory role in terms in sensing environmental influences such as heat, pressure, pain, and allergen and microorganism entry.
- A regenerative role in keeping itself viable, active and a continual state of destruction, repair and regeneration.

Even though skin is multifunctional its most important role is to act as a barrier against extraneous chemical and biological substances. However, in spite of this excellent resistance, it is by nature still a semi-permeable membrane. Scientists have long identified this and concluded that skin can potentially act as an entry point for a host of drugs and therapeutic substances, which serves as the core concept in transdermal drug delivery. The systemic benefits of transdermal anti–infective and hormonal agents were reported as early as 1940s (?). The main interests in the application of transdermal or topical products are the following:

- Local effects in dermatology (corticosteroids for dermatitis)

- Transport through skin for systemic effects (nicotine and birth–control patches)
- Surface effects (cosmetic products, UV protection creams and lotions)
- Target deeper tissues (products for treating muscle inflammation)

Skin became popular as a site for systemic product delivery as it was believed to prevent or mitigate problems associated with:

- Inactivity due to drastic change of pH from an alkaline condition in the mouth to an extremely acidic condition in the stomach
- Gastrointestinal breakdown due to chemical or enzymatic reactions
- Rejection and removal due to emptying of the stomach and excretion
- Passing through major organs like liver and kidney where it does not have any particular function, but may prove to be potentially harmful
- Uncontrollable presence in the body once administered

Drugs administered transdermally have a much higher assimilation rate, do not become inactive due to drastic changes in pH or due to breakdown owing to chemical, metabolic or enzymatic reactions and gives the user complete control over initiation, dosage and administration rate and termination. These resulted in an aggressive research in this area in the mid 20th century and led to the development of the first transdermal patch in 1979. As a result, the current annual market for transdermal drug delivery is US$ 3 billion. Currently a host of chemical substances such as like scopolamine, nitroglycerine, nicotine, testosterone and also clonidine, fentanyl, estradiol etc. are being successfully administered through skin on a commercial scale, duly approved by the Food and Drug Administration (FDA). However, many of these chemicals have the potential of interacting adversely with the skin or with internal physiological systems. Moreover, owing to its immediate proximity to the environment, skin gets naturally exposed to a variety of substances

on a regular basis, which can also have adverse local or systemic physiological effects. This exposure can be either deliberate (application of cosmetic or topically applied pharmaceuticals) or accidental (exposure to chemical spills, sustained occupational exposure). Also, the nature of exposure can be classified as acute (sudden exposure to high concentrations) or chronic (prolonged exposure to low concentrations). Hence, the disposition of chemical compounds from human skin has evoked substantial interest amongst researchers.

1.2 Microstructure of the Human Skin

Although In order to fully understand the mechanistic details of transdermal disposition of chemical compounds, a systematic study is essential. Such studies are central to the development and successful commercialization of transdermal drugs and topically-applied pharmaceuticals and, at the same time, critical for aspects pertaining to occupational safety and health, toxic exposures and environmental hazards. The latter pertain to a proper understanding of the immunological and toxicological consequences of the application of a particular chemical or a mixture of chemicals to the skin. Thus, there is a pressing need for basic research to develop accurate and reliable mathematical models based on detailed microscopic theories and experiments. This is tied to having a detailed understanding of the anatomy of the skin microstructure and the mechanism of percutaneous penetration. Fig 1.1 is a graphical representation of the human skin (Montagna, 1992). It consists of two distinct layers: epidermis and dermis, as shown in Fig. 1.2 (Bouwstra et al., 2002; Roberts and Walters, 1998; Menon et al., 2002). The lower layer, dermis (forming the bulk of skin), is made up of connective tissue elements like collagen (both type III and type I), elastin, glycosaminoglycans or GAGs (collectively termed as extra cellular matrix or ECM) and fibroblasts. The dermis is highly

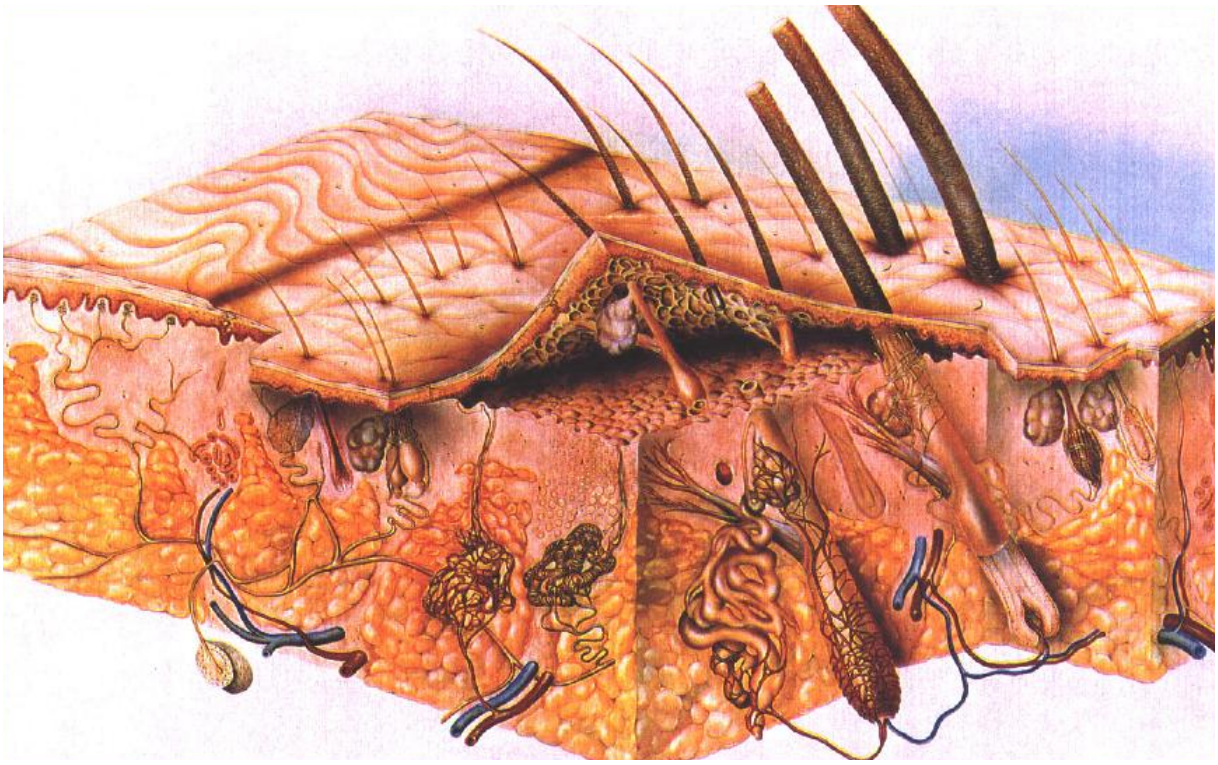


Fig. 1.1 Anatomy of the Human Skin

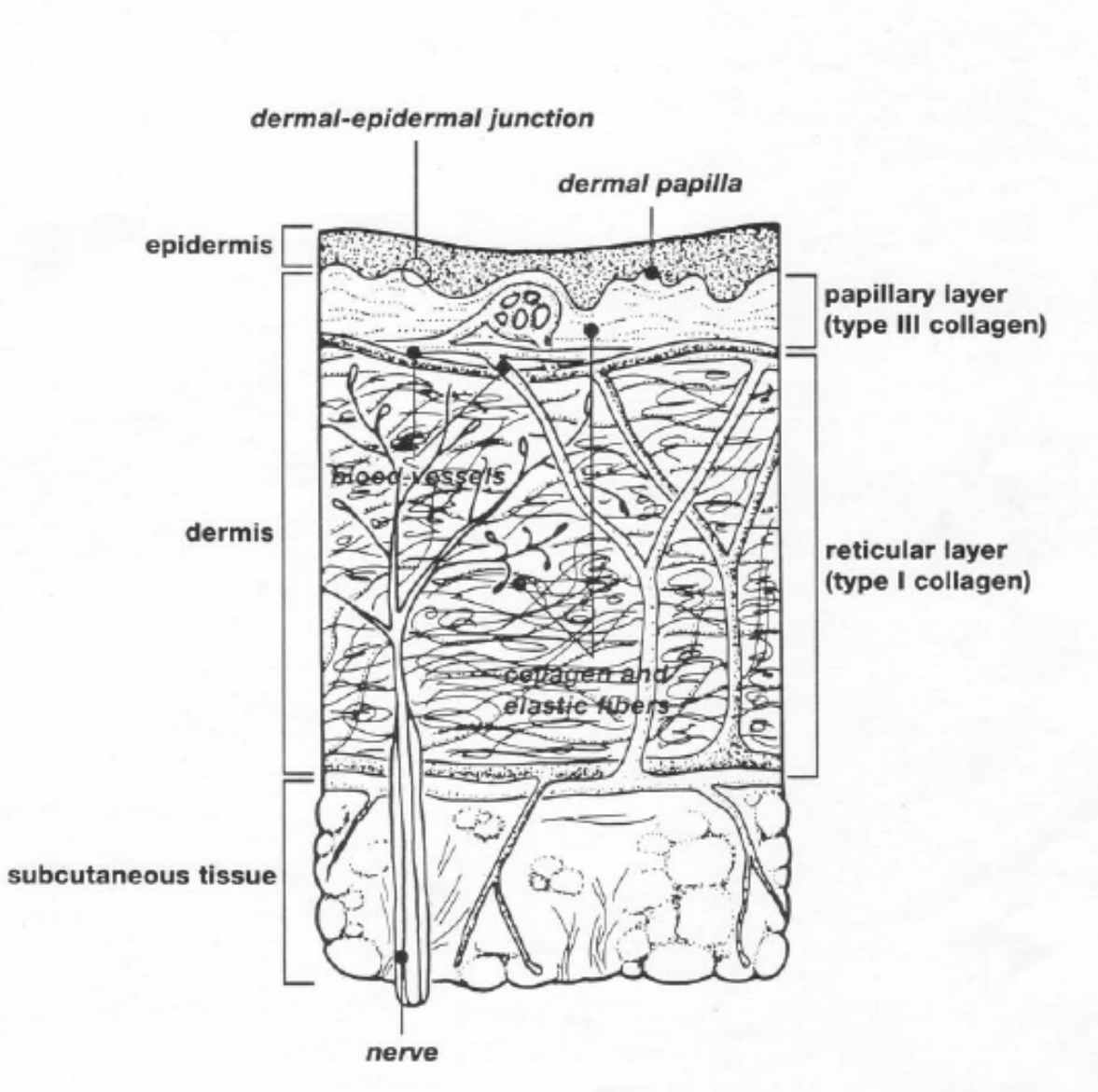


Fig. 1.2 Cross–sectional Representation of Human Skin: Epidermis and Dermis.

vascular, *i.e.* contains active capillaries, and also includes the pilosebaceous units, sweat glands, dermal adipose cells, mast cells, and infiltrating leucocytes. The overlying epidermis is avascular and composed primarily of keratinocytes, and the rest are melanocytes, Langerhans cells, and Merkel cells (mechanoreceptors). The epidermis is between 75 and 150 μm, except on palms and soles where its thickness may reach 0.4 – 0.6 mm. Human epidermis is highly stratified and, in turn, divisible into several distinct layers as:

- *Stratum corneum* (horny layer): topmost or outermost non–viable layer
- *Stratum Lucidum*: which is present only in thick parts of the skin and otherwise absent
- *Stratum Granulosum* (granular layer): darker layer with intracellular granules
- *Stratum Spinosum* (prickle layer): layer with spiny appearance
- *Stratum Basale* (basal layer): innermost layer of the epidermis

These sub-layers are represented in figures 1.3 and 1.4 (Amsden, 1995). Structural details of the individual epidermal layers along with their physiological functions can be found in an excellent review by Menon (2002). In the early part of the 20th century, with the purpose of understanding the interrelationship between the structure and barrier function of skin, Rein proposed that a layer of cells joining the stratum corneum from the surface of the skin to the epidermis posed the major resistance to transdermal transport (Rein, 1924). Blank carried out trans-epidermal water loss or TEWL experiments by removing sequential layers of stratum corneum from the surface of the skin. He showed that the rate of water loss from skin increased dramatically once the stratum corneum was removed and modified Rein's hypothesis. Through their persistent studies, Blank and Scheuplein proved conclusively that the outermost non–viable epidermal sub-layer, the stratum corneum (or SC) is the primary barrier to transdermal transport (Blank, 1964; Scheuplein, 1965; Scheuplein and Blank, 1971).

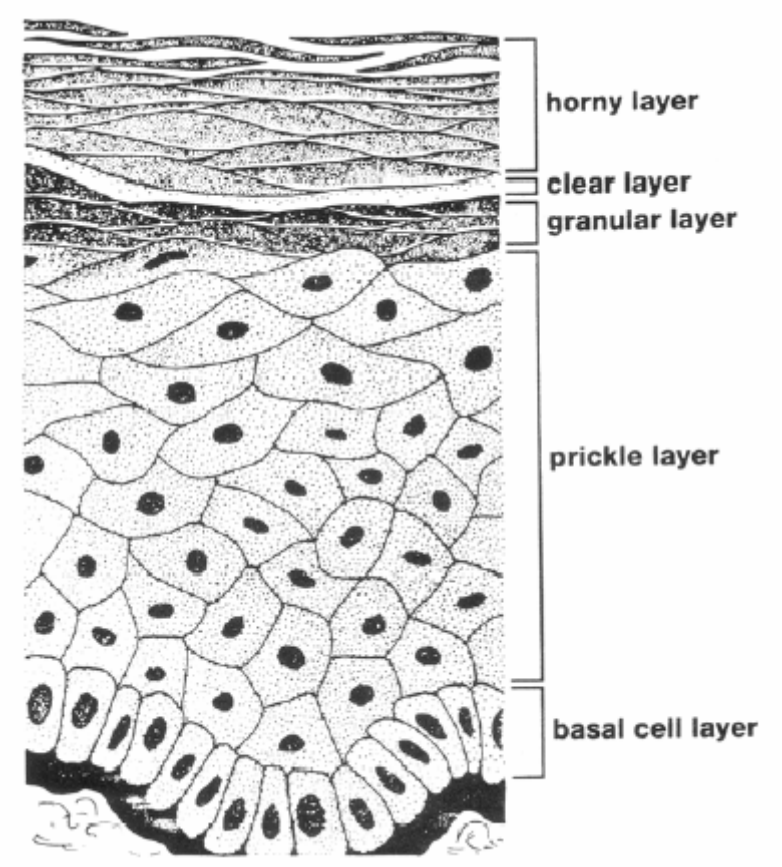


Fig. 1.3 Cross–sectional Representation of Human Epidermis.

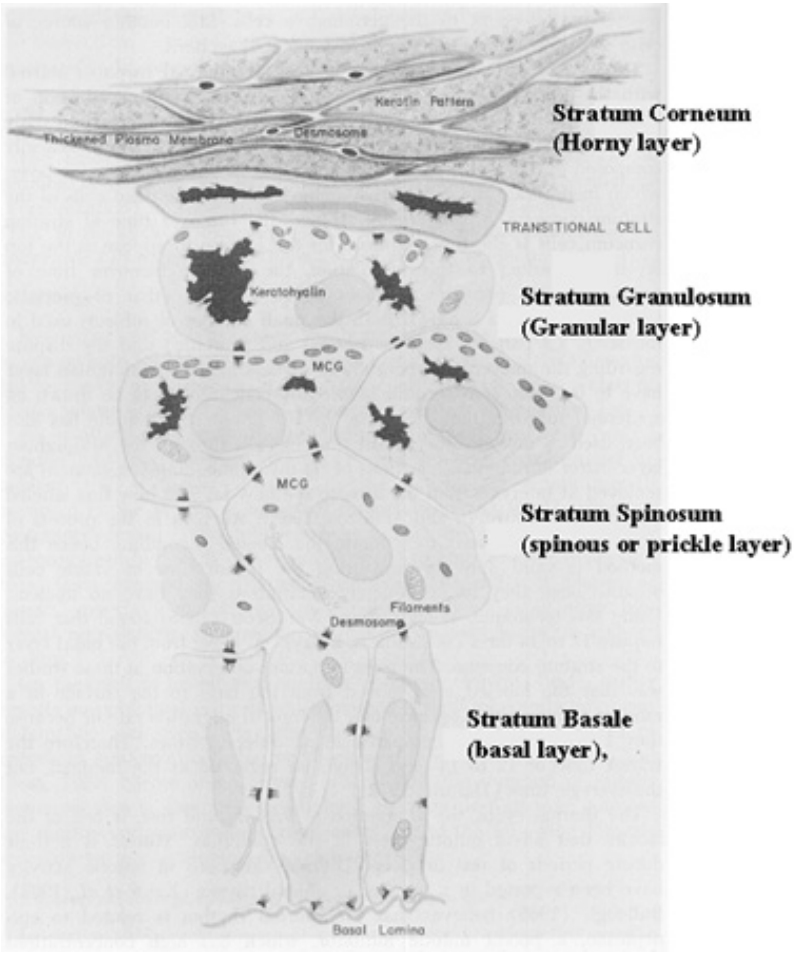


Fig. 1.4 Cross–sectional Representation of the Stratified Epidermis.

This was a significant development as it greatly helped researchers narrow down their attention on the structure–function interrelationship for this particular layer, eliminating the necessity of emphasizing on the rest. However, under certain circumstances, the permeability through the underlying skin sub-layers, namely the viable epidermis and dermis can be important (Cleek and Bunge, 1993; Kretsos, 2006).

1.3 Modeling of Skin Permeation

The use of quantitative relationships or 'models' has played an invaluable role in both understanding the science underlying many processes and in analyzing the engineering that has produced results. Although there are different ways of modeling transport through human skin, they can be broadly classified as being mechanistic or non-mechanistic, and either transient or steady-state. A general review of the steady-state methods along with a critical comparison has been presented by Yamashita and Hashida (2003). The most influential among these steady-state methods was introduced by Potts and Guy (1992), who considered skin as a structured lipid membrane characterized by an octanol-like chemical environment and a strong dependence of diffusivity on the molecular weight of the penetrant. Potts and Guy provided the first quantitative analysis of a large human skin permeability database collected by Flynn (1990). Many variations of this approach have been proposed, often involving sets of chemical descriptors that are more complex than molecular weight or diffusivity. Collectively, they are called Quantitative Structure Activity Relationships (QSARs) or, more precisely Quantitative Structure Permeability Relationships (QSPRs) in such situations. A typical QSAR has three distinct components that form its foundation (Cronin, 1999):

a) Biological Data: This is the most important piece since it serves as the foundation for model building. The applicability and efficacy of QSARs rely heavily upon the source clinical or experimental data.

b) Physico–Chemical and Structural Properties and Descriptors: The underlying assumption of any QSAR is that the biological activity of a penetrant is dependent, in some quantitative manner, on the physico–chemical and/or structural properties of the chemical. There can be a variety of such descriptors employed to accurately model the system.

c) Modeling Techniques: This is the final step linking the biological activity of a series of compounds and their physico–chemical properties. Commonly these techniques range from least squares minimizing algorithms, multiple linear or non-linear regression analyses, neural networks, unconstrained optimization etc. These modeling techniques have been discussed extensively by Livingstone (1995).

There are numerous QSARs that have been developed to model percutaneous penetration. Even though QSARs employ a host of structural and physicochemical descriptors, the most frequently used are the octanol–water partition coefficient of the substance and some measure of molecular size like molecular weight or molecular volume. A brief description of QSARs along with a concise account of the historical development of QSARs has been excellently reviewed by Schultz and coworkers (Schultz et al., 2003). Also, comprehensive critical analyses of QSPRs have been provided by Geinoz et al. (2004); Moss et al. (2002); Moss and Cronin (2002); Patel (2002).

Another approach to steady-state modeling of skin permeation is pharmacokinetic (PK) modeling. These models represent the permeating system through single or multiple layers. These layers are well–stirred and act as reactors or reservoirs of chemicals, with transfer between

the compartments depicted by first-order kinetic expressions. Skin, in these models is usually represented either as a single compartment or by two compartments in an attempt to distinguish between the hydrophilic and the hydrophobic layers of the skin. In reality, however, the skin behaves as a multi–laminate barrier. There is substantial volume of literature that uses PK models to represent transdermal permeation. In most of these works, the rate constants have been estimated through regression analysis of experimental data, as opposed to estimating them from relevant physicochemical properties using mechanistic information. However, this seriously limits the predictive power of such models and restricts them only to the basis set or homologous compounds. A better alternative is using kinetic constants that are derived from system parameters such as diffusivity, partition coefficient, skin thickness, cutaneous blood flow rate etc. The existing PK approaches have been extensively reviewed and analyzed by McCarley and Bunge (2001). In the most common descriptions, skin is represented through one or two compartments although it may be stratified to contain several compartments. In the simplest one-compartment model, in addition to the skin compartment, there is one compartment each for the vehicle and the blood/ receptor solution. A two–compartment skin model is a simple extension of the earlier, where the skin is represented by two distinct layers, the first layer typically representing the SC and the second layer representing viable tissue. The underlying mathematical equations are simply the result of a material balance over each layer. The collection of these material balance equations for all the layers is a system of homogenous or nonhomogenous linear first-order differential equations. The system is fully characterized once the values of 1^{st}-order rate constants are ascertained. There have also been attempts to extend the model beyond two compartments (Anissimov and Roberts, 2002). This approach uses models up to 6 and 8 compartments. Even though inclusion of more compartments makes a PK model more

realistic, the underlying mathematics becomes extremely cumbersome and little is gained in terms of mechanistic information or accuracy. Thus, any PK model should contain an optimal number of layers, representing a trade-off between realistic representation and mathematical compelxity.

On the contrary, transient skin permeation models were pioneered by Scheuplein (1965, 1971) and are comprised of both diffusional models and compartmental pharmacokinetic approaches. Cooper and Berner (1985) reviewed the proposed skin permeation models and suggested that a reliable and realistic state–of–the–art model for skin permeation should include advance features such as finite dose (as opposed to an infinite reservoir or source), evaporation of vehicle (finite vehicle volume), diffusional limitations, complicacies of skin microstructure with particular reference to modes of transport (hydrophobic, hydrophilic etc.), adsorption, surface immobilization resulting in surface reduction, shunt pathways (hair follicles, sweat glands etc.) and, finally, coupled transport. Inclusion of these features may render the model equations analytically unsolvable. Thus, several researchers employed numerical solution techniques to solve model equations describing mass transport through human skin. An excellent summary of early transient mathematical models can be found in the works of McCarley and Bunge (2001). Since then considerable progress has been made by several research groups (Anissimov and Roberts, 1999, 2001, 2004; Barbero and Frasch, 2005; Boderke, 2000; Fernandes et al., 2005; Frasch, 2002; Frasch and Barbero, 2003; George, 2005; George et al., 2004; Gumel, 1998; Kasting, 2001; Kasting and Miller, 2006; Simon and Loney, 2005). It should be noted that most of these approaches have (either theoretically or experimentally) considered disposition of a pure non-volatile compound applied in limited doses to human skin. Studies on pure volatile liquids and binary mixtures are either rare or absent. It is noteworthy that the

disposition characteristics for volatile liquids or mixtures (possibly containing volatile liquids) can be markedly different than that of a pure non-volatile liquid. For volatile liquids the difference is most prominent with reference to high vapor pressure (evaporative flux) and evaporative cooling. For binary mixtures, the biggest difference lies in the permeation enhancement due to the presence of the other component as well as non-ideal behavior of the mixture.

Studies on skin absorption of pure volatile compounds are rare. Water has been studied due to its abundance in the environment and its widespread use in formulations (Blank et al., 1984; Kasting et al., 2003). Organic solvents used in the industry have also been studied, for example N,N-dimethylformamide. The latter is known to cause adverse toxicological effects on the human body and their skin permeation characteristic was measured by Mráz and Nohová (1992). Batterman and Franzblau (1997) reported an experimental study on the penetration of neat methanol in human volunteers. They found the in vivo absorption rates to be significantly higher than the corresponding in vitro absorption rates. Glycol ethers are also an important class of compounds miscible with both aqueous and organic media and have found widespread use in industrial and household applications. However, they can penetrate skin rapidly (Kezic et al. 1997a, 1997b; Larez-Filon et al. 1999) and dermal absorption of these compounds may therefore present a significant health risk. Larez-Filon et al. (1999) studied the human skin permeation characteristics of several glycol ethers in vitro. The compounds were studied in pure form and the kinetics was reported in terms of a permeation profile, lag time and steady-state flux. They found significant percutaneous absorption of these compounds and commented that their air Threshold Limit Values (TLV®s) will be inadequate to predict a potentially harmful exposure. Similar studies on glycol ethers were done by Wilkinson and Williams (2002) who measured the

penetration characteristics of 2-ethoxyethanol, 2-butoxyethanol and 1-methoxy-2-propanol. Ethoxyethanol has demonstrable reproductive, haematological and developmental effects, while haemolytic effects of 2-butoxyethanol have been demonstrated in rats (Wilkinson and Williams, 2002). Propylene glycol ethers such as 1-methoxy-2-propanol are much less toxic and provide as alternatives to ethylene glycol ethers. Among the glycol ethers, 2-butoxyethanol has derived special attention and has also been studied by (Jakasa, et al., 2004; Lockley, et al., 2004). Other than the glycolic ethers, Korinth et al. (2003) studied the percutaneous penetration of ethylene glycol and trimethylbenzene. A related yet slightly different study was performed by Kezic et al. (2000) who experimentally measured the dermal absorption rates of common hydrocarbons in their vaporous states in vivo. The experimental data were then used to validate the permeation models of Cleek & Bunge (1993) and Wilshut & ten Berge (1995). Wilkinson et al. (2006) studied the percutaneous penetration characteristics of pure compounds such as testosterone, caffeine, propoxur etc. However, these articles do not report any quantitative analysis by validating the established (or novel) mechanistic (diffusion) models by utilizing their experimental data.

1.4 Outline of the Thesis

The general goal of this research is to develop a comprehensive mathematical model describing transient permeation of chemicals through human skin. This model is general and can be applied to describe permeation of any chemical compound as applied onto skin in the liquid form (either in pure form or from a binary solution). The model will then be validated against real-time transient skin permeation data for a host of common chemical compounds generated in -house. Chapter 2 describes the development of the theoretical model for skin permeation of a

pure volatile liquid. Chapter 3 describes the application of an approximation technique, scaling analysis on the model equations in order to get a clear idea on the dynamics of the physical system. Chapters 4, 5 and 6 apply the transient mass transport model to percutaneous penetration of ethanol and benzene. Chapter 6 describes the development of a coupled heat and mass-transport model for a binary mixture (two permeants) and applies the model to describe the permeation of a representative substance benzyl alcohol. Chapter 7 makes generalized conclusions about the research and recommendations for future work.

References

Amsden, B. G., Goosen, M. F. A., (1995). Transdermal delivery of peptide and protein drugs: an overview. AIChE Journal, 41 (8), 1972–1997.

Anissimov, Y.G., Roberts, M.S. (1999). Diffusion modeling of percutaneous absorption kinetics. 1. Effects of flow rate, receptor sampling rate, and viable epidermal resistance for a constant donor concentration. Journal of Pharmaceutical Sciences, 88 (11), 1201–1209.

Anissimov, Y. G., Roberts, M. S. (2002). A compartmental model of hepatic disposition kinetics: 1. model development and application to linear kinetics. Journal of Pharmacokinetics and Pharmacodynamics, 29, 131–156.

Anissimov, Y.G., Roberts, M.S. (2001). Diffusion modeling of percutaneous absorption kinetics: 2. finite vehicle volume and solvent deposited solids. Journal of Pharmaceutical Sciences, 90 (4), 504–520.

Anissimov, Y.G., Roberts, M.S. (2004). Diffusion modeling of percutaneous absorption kinetics: 3. variable diffusion and partition coefficients, consequences for stratum corneum depth profiles and desorption kinetics. Journal of Pharmaceutical Sciences, 93 (2), 470–487.

Barbero, A.M., Frasch, H.F. (2005). Modeling of diffusion with partitioning in stratum corneum using a finite element model. Annals of Biomedical Engineering, 33, 1281-1292.

Batterman, S.A., Franzblau, A. (1997). Time-resolved cutaneous absorption and permeation rates of methanol in human volunteers. International Archives of Occupational and Environmental Health, 70, 341-351.

Blank, I. H., (1964). Penetration of low–molecular–weight alcohols into skin. I. effect of concentration of alcohol and type of vehicle. Journal of Investigative Dermatology, 43, 415-420.

Blank, I. H. (1984). The diffusion of water across the stratum corneum as a function of its water content. Journal of Investigative Dermatology, 82, 188-194.

Boderke, P., Schittkowski, K., Wolf, M., Merkle, H.P., (2000). Modeling of diffusion and concurrent metabolism in cutaneous tissue. Journal of Theoretical Biology, 204, 393–407.

Bouwstra, J. A., Honeywell-Nguyen, P.L., Gooris, G.S., Ponec, M., (2002). Structure of the skin barrier and its modulation by vesicular formulations. Progress in Lipid Research, 42, 1–36.

Cleek, R.L., Bunge, A.L., (1993). A new method for estimating dermal absorption from chemical exposure. 1. General approach. Pharmaceutical Research, 10, 497–506.

Cooper, E. R., Berner, B. (1985). Skin Permeability, in: Skerrow, D. & Skerrow, C. J. (Eds), Methods of Skin Research, John Wiley & Sons Ltd., New York.

Cronin, M.T.D., Dearden, J.C., Moss, G.P., Murray-Dickson, G., (1999). Investigation of the mechanism of flux across human skin in vitro by quantitative structure–permeability relationships. European Journal of Pharmaceutical Sciences, 7, 325–330.

Fernandes, M., Simon, L., Loney, N.W., (2005). Mathematical modeling of transdermal drug-delivery systems: Analysis and applications. Journal of Membrane Science, 256, 184-192.

Flynn, G.L., (1990). Physicochemical determinants of skin absorption, in: Gerritv, T.R. & Henry, C.J. (Eds.), Principles of Route-to-Route Extrapolation for Risk Assessment, Elsevier, New York, pp. 93–127.

Frasch, H.F. (2002). A random walk model of skin permeation. Risk Analysis, 22, 265-276.

Frasch, H.F., Barbero, A.M. (2003). Steady-state flux and lag time in the stratum corneum lipid pathway: results from finite element models. Journal of Pharmaceutical Sciences, 92, 2196-2207.

Geinoz, S., Guy, R.H., Testa, B., Carrupt, P.-A., (2004). Quantitative structure–permeability relationships (QSPeRs) to predict skin permeation: a critical evaluation. Pharmaceutical Research, 21 (1), 83–92.

George (2005). A two-dimensional mathematical model of non-linear dual-sorption of percutaneous drug absorption. Biomedical Engineering Online, 4 (40), available online at http://www.biomedical-engineering-online.com/content/4/1/40 [accessed 5/16/2007]

George, S., Kubota, K., Twizell, E.H., (2004). A two-dimensional mathematical model of percutaneous drug absorption. Biomedical Engineering Online, 3 (18), available online at http://www.biomedical-engineering-online.com/content/3/1/18 [accessed 5/16/2007]

Gumel, A.B., Kubota, K., Twizell, E.H., (1998). A sequential algorithm for the non-linear dual-sorption model of percutaneous drug absorption. Mathematical Biosciences, 152, 87-103.

Jakasa, I., Mohammadi, N., Kruse, J., Kezic, S.. (2004). Percutaneous absorption of neat and aqueous solutions of 2-butoxyethanol in volunteers. International Archives of Occupational and Environmental Health, 77, 79-84.

Kasting, G.B. (2001). Kinetics of finite dose absorption through skin 1. vanillylnonanamide. Journal of Pharmaceutical Sciences, 90, 202-212.

Kasting, G.B., Miller, M.A. (2006). Kinetics of finite dose absorption through skin 2. Volatile compounds. Journal of Pharmaceutical Science, 95, 268–280.

Kasting, G.B., Barai, N.D., Wang, T.F., Nitsche, J.M., (2003). Mobility of water in human stratum corneum. Journal of Pharmaceutical Sciencess, 92 (11), 2326-2340.

Kezic, S., Mahieu, K., Monster, A.C., de Wolff, F.A., (1997a). Dermal absorption of vaporous and liquid 2-methoxyethanol and 2 ethoxyethanol in volunteers. Occupational and Environmental Medicine, 54, 38–43.

Kezic, S., Dermal exposure to vaporous and liquid organic solvents in volunteers. Fundamental and Applied Toxicology, 36, 190.

Kezic, S., Monster, A.C., Kruse, J., Verberk, M.M., (2000). Skin absorption of some vaporous solvents in volunteers. International Archives of Occupational and Environmental Health, 73, 415-422.

Korinth, G., Geh, S., Schaller, K.H., Drexler, H., (2003). In vitro evaluation of the efficacy of skin barrier creams and protective gloves on percutaneous absorption of industrial solvents. International Archives of Occupational and Environmental Health, 76, 382-386.

Larese-Filon, F., Fiorito, A., Adami, G., Barbieri, P., Coceani, N., Bussani, R., Reisenhofer, E., (1999). Skin absorption in vitro of glycol ethers. International Archives of Occupational and Environmental Health, 72, 480-484.

Livingstone, D.J. (1995). Data Analysis for Chemists: Applications to QSAR and Chemical Product Design. Oxford University Press, Oxford.

Lockley, D.J., Howes, D., Williams, F.M., (2004). Percutaneous penetration and metabolism of 2-butoxyethanol. Archives of Toxicology, 78, 617-628.

McCarley, K. D., Bunge, A. L., (2001). Pharmacokinetic models of dermal absorption," Journal of Pharmaceutical Sciences, 90, 1699–1719.

Menon, G. K., (2002). New insights into skin structure: scratching the surface. Advance Drug Delivery Reviews, 54 (Suppl. 1), S3–S17.

Montagna, W., Kligman, A.M., Carlisle, K.S., (1992). Atlas of Normal Human Skin. Springer–Verlag, Berlin and Heidelberg.

Moss, G. P., Cronin, M. T. D., (2002). Quantitative structure–permeability relationships for percutaneous absorption: re–analysis of steroid data. International Journal of Pharmaceutics, 238, 105–109.

Moss, G. P., Cronin, M. T. D., (2002). Quantitative structure–permeability relationships for percutaneous absorption: re–analysis of steroid data. International Journal of Pharmaceutics, 238, 105–109.

Mráz, J., Nohová, H. (1992). Percutaneous absorption of N,N-dimethylformamide in humans. International Archives of Occupational and Environmental Health, 64, 79-83.

Patel, H., ten Berge, W., Cronin, M.T., (2002). Quantitative structure–activity relationships (QSARs) for the prediction of skin permeation of exogenous chemicals. Chemosphere, 48, 603–613.

Potts, R. O., Francoeur, M. L., (1991). The influence of stratum comeum morphology on water permeability. Journal of Investigative Dermatology, 96, 495-499.

Rein, H., (1924). Experimental electroendosmotic studies on living human skin. Z. Biol., 81, 124.

Roberts, M. S., Walters, K. A., (1998). The relationship between structure and barrier function of skin, in: Roberts, M. S. & Walters, K. A. (Eds.), Dermal Absorption and Toxicity Assessment, Marcel Dekker Inc., New York.

Scheuplein, R. J., (1965). Mechanism of percutaneous absorption I. routes of penetration and the influence of solubility. Journal of Investigative Dermatology, 45, 334-345.

Scheuplein, R. J., Blank, I. H., (1971). Permeability of the skin. Physiological Reviews, 51, 702-747.

Schultz, T. W., Cronin, M.T.D., netzeva, T.I., (2003). The present status of QSAR in toxicology. Journal of Molecular Structure (Theochem), 622, 23–28.

Simon, L., Loney, N. W., (2005). An analytical solution for percutaneous drug absorption: Application and removal of the vehicle. Mathematical Biosciences, 13, 119–139.

Singh, S., Singh, J., (1993). Transdermal drug delivery by passive diffusion and iontophoresis: a review. Medicinal Research Reviews, 13, 569–621.

Wilkinson, S.C. & Williams, F.M. (2002). Effects of experimental conditions on absorption of glycol ethers through human skin in vitro. International Archives of Occupational and Environmental Health, 75, 519-527.

Wilkinson, S.C., Maas, W.J., Nielsen, J.B., Greaves, L.C., van de Sandt, J.J., Williams, F.M., (2006). Interactions of skin thickness and physicochemical properties of test compounds in percutaneous penetration studies. International Archives of Occupational and Environmental Health, 79, 405-413.

Wilschut, A. Ten Berge, W.F. (1995). Two mathematical skin permeation model for vapors. in: Brain, K.R., James, V.J. & Walters, K.A. (Eds.), Prediction of percutaneous penetration, Vol 4a. 3M Medica, La Grande Motte.

Yamashita, F., Hashida, M., (2003). Mechanistic and empirical modeling of skin permeation of drugs. Advanced Drug Delivery Reviews. 55, 1185–1199.

CHAPTER 2

PURE COMPONENT MODEL DEVELOPMENT

2.1 Modeling Considerations

This section elaborates the considerations that are necessary in order to develop a comprehensive mathematical model describing mass transport through human skin layers. The model is based on the fundamental principles of material balance, with as few limiting assumptions as possible. The process initiates with the application of a known amount of pure volatile liquid on the top of the skin, which immediately starts disposing off the skin surface through simultaneous evaporation and absorption. Thus, the describing equations include both the evaporation and absorption dynamics. Limiting assumptions pertaining to the skin layers such as 'well–mixed' or 'well–stirred' (no spatial variation of composition); an infinite source of components; constant composition (no time variation of composition), and constant thickness have been avoided. The vehicle layer has been treated as finite and its shrinking thickness with time has been appropriately incorporated in the model. The evaporation takes place into the surrounding gaseous (air) phase, considered to be a single component for all practical purposes. Evaporation of the volatile solvent may result in slight decrease in the temperature of the liquid remaining on top of the skin. However, for the purposes of this work, it has been assumed to be isothermal.

2.2 Development of the Model Equations

The dual evaporation-absorption process is shown schematically in Fig. 2.1. It involves the application of a known amount of pure volatile liquid on the top of the skin forming an explicit vehicle layer (VH) of initial thickness L_0. Now, the human skin is a highly stratified barrier with complex structural and physiological aspects (Bouwstra, 2002; Menon et al., 2002;

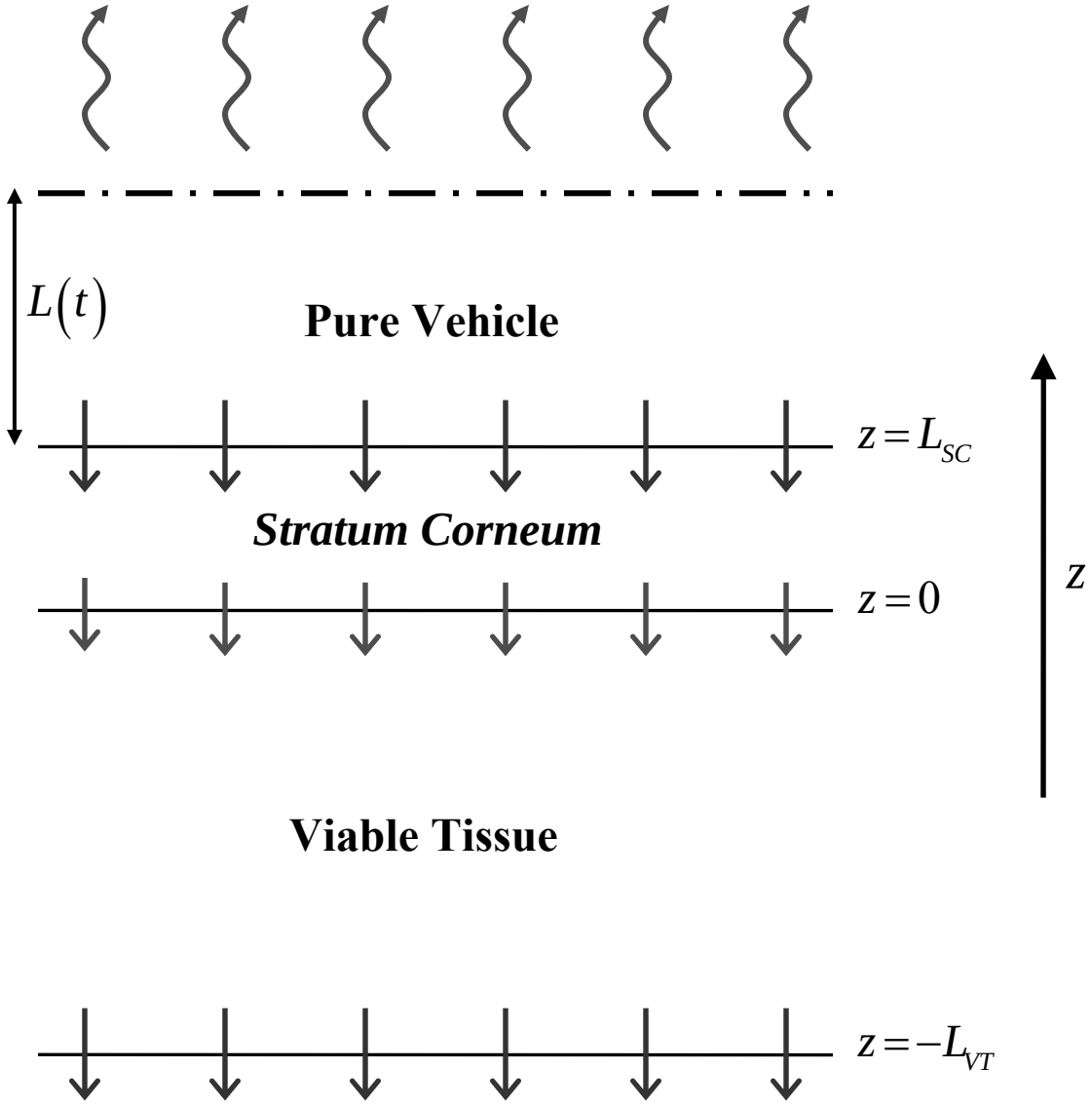


Fig. 2.1 Schematic representation of simultaneous evaporation and absorption of a volatile liquid from the surface of the skin.

Montagna et al., 1992; Roberts et al., 1998). It consists of two main layers: epidermis and dermis. The lower layer, dermis, forms the bulk of skin and is highly vascular, *i.e.* contains active capillaries. The top layer, epidermis, is devoid of blood vessels and can in turn be subdivided into several distinct layers. The structure and function of the individual epidermal layers has been extensively discussed by Menon (2002). Most importantly, it has been established that the outermost epidermal layer, the non–viable SC is the primary barrier to transdermal transport for most compounds (Blank, 1964; Scheuplein, 1965; Scheuplein and Blank, 1971). Accordingly, the current model considers skin to be a two-layered membrane. The top layer is the more resistive SC (of thickness L_{SC}) while the bottom layer, henceforth referred to as viable tissue or VT and having thickness L_{VT}, represents viable epidermis and a fraction of the unperfused dermis. In the absence of an appropriate model describing intermolecular interaction in such biological systems, the proposed model does not consider either swelling or shrinking and the skin layer thicknesses remain constant throughout the permeation process. The origin of the spatial coordinate is at the SC-VT interface. While a VH layer is present, the volatile compound dissipates from this layer via two simultaneous mechanisms. At the liquid-gas interface, the compound is lost by evaporation, while at the VH-SC interface it penetrates via diffusion (and convection) into the SC. This results in a thinning of the VH film whose thickness is given by $L = L(t)$. When the vehicle layer completely disappears, the system is reduced to a simpler two-layer model. The compound now desorbs from the SC and VT layers, respectively, through evaporation at the newly formed SC-air interface and clearance at the lower boundary of VT. We adopt the concept of a deposition layer as described in Kasting and Miller (2006). This solvent deposition is assumed to be instantaneous; a reasonable assumption for volatile solvents. The amount of compound deposited into SC is given by a product of the saturation concentration

of the compound in SC and the depth of the solvent-deposited region (henceforth referred to as the deposition depth). The net amount of liquid left in the VH is the difference between the initial amount applied and the amount deposited into the SC. The entire system is assumed to be isothermal and at 32^0 C.

The general conservation equation for mass transport through an arbitrary barrier layer LY can be written as:

$$\frac{\partial \rho_P^{LY}}{\partial t} = -\frac{\partial n_P^{LY}}{\partial z} \tag{1}$$

where t is the time, z is the spatial coordinate, ρ_P^{LY} and n_P^{LY} respectively denote the mass concentration and total mass flux of the penetrant in a particular layer (LY = SC, VT). By definition, mass concentration is a product of mass fraction (ω_P^{LY}) and the total solution (layer) density. It can be written as:

$$\rho_P^{LY} = \omega_P^{LY} \rho^{LY} \tag{2}$$

Since SC and VT are immobile phases, the total mass flux of a penetrant through these layers can be written as (Bird, et al., 1960):

$$n_P^{LY} = \frac{j_P^{LY}}{1 - \omega_P^{LY}} \tag{3}$$

where j_P^{LY} is the diffusive flux of the penetrant. The diffusive flux can be expressed in terms of Fick's 1st law of diffusion (Bird et al., 1960) as:

$$j_P^{LY} = -\rho^{LY} D_{PLY} \frac{\partial \omega_P^{LY}}{\partial z} \tag{4}$$

where D_{PLY} is the diffusivity of the penetrant inside LY. Thus, the governing equations describing mass transport of the penetrant in SC and VT, respectively, are given by:

$$\frac{\partial \omega_P^{SC}}{\partial t} = \frac{\partial}{\partial z}\left[\frac{D_{PSC}}{\left(1-\omega_P^{SC}\right)}\frac{\partial \omega_P^{SC}}{\partial z}\right] \tag{5}$$

$$\frac{\partial \omega_P^{VT}}{\partial t} = \frac{\partial}{\partial z}\left[\frac{D_{PVT}}{\left(1-\omega_P^{VT}\right)}\frac{\partial \omega_P^{VT}}{\partial z}\right] \tag{6}$$

Eqs. (5) and (6) are non-linear second-order PDEs, each requiring one initial condition (IC) and two boundary conditions (BCs) for a particular solution. The IC for the PDEs is given by a known initial concentration of permeant in these layers. For the VT, this is equal to zero:

$$\omega_P^{VT}\Big|_{z,0} = 0 \tag{7}$$

However, for the SC, the presence of the solvent-deposited region means that a fraction of the SC will have a non-zero concentration in the beginning. This IC can be written as:

$$\omega_P^{SC}\Big|_{z,0} = 0 \text{ for } 0 \le z < \left(1-f_{dep}\right)L_{SC} \tag{8}$$

$$\omega_P^{SC}\Big|_{z,0} = \omega_{sat}^0 \text{ for } \left(1-f_{dep}\right)L_{SC} \le z \le L_{SC} \tag{9}$$

where f_{dep} is the fractional deposition depth and ω_{sat}^0 is the saturation composition. The bottom of the viable tissue has a sink condition for permeant concentration. This is representative of in vivo situations in which the penetrant is completely cleared by a network of blood capillaries or the in vitro situation, where it is flushed away by circulating buffer solution. This condition is written as:

$$\omega_P^{VT}\Big|_{-L_{VE},t} = 0 \tag{10}$$

The BC at the top of the SC will depend on the presence or absence of the vehicle layer (as discussed above) and is given by:

$$\omega_P^{SC}\Big|_{L_{SC},t} = K_{SC/VH}^{P}\frac{\rho_P^{\circ}}{\rho^{SC}} \text{ for } 0 \le t \le t_{depl} \tag{11}$$

$$n_P^{SC}\Big|_{L_{SC},t} = n_P^G\Big|_{L_{SC},t} \text{ for } t > t_{depl} \tag{12}$$

where $K_{SC/VH}^P$ and ρ_P° denote the SC/VH partition coefficient and the pure component density of the penetrant, respectively. n_P^G is the mass flux of the penetrant into the surrounding air phase and t_{depl} is the time required for the vehicle layer to completely disappear. The boundary conditions at the SC-VT interface remain the same at all times and are given by a partition relation and continuity of flux.

$$\omega_P^{SC}\Big|_{0,t} = \frac{\rho^{VT}}{K_{VT/SC}^P \rho^{SC}} \omega_P^{VT}\Big|_{0,t} \tag{13}$$

$$n_P^{VT}\Big|_{0,t} = n_P^{SC}\Big|_{0,t} \tag{14}$$

Finally, since the VH layer consists of a single component, an instantaneous material balance on this layer results in an ODE (instead of a PDE) and is written as:

$$\rho_P^\circ \frac{dL}{dt} = -n_P^G + n_P^{SC}\Big|_{z=L_{SC}} \tag{15}$$

where $\frac{dL}{dt}$ is the velocity of the liquid-gas interface. Eq. (15) is coupled with Eqs (5) and (6) and needs an initial condition. This is given by the initial thickness of the VH:

$$L(0) = L_0 \tag{16}$$

In order to complete the description of the model, an expression for the gas-phase flux of the permeant from the VH is required. The simplest description is through the use of a lumped parameter or mass-transfer coefficient. The expressions for the gas-phase flux from the vehicle and SC are given by Eqs. (17) and (18), respectively.

$$n_P^G = k_{evap} \rho_P^\circ \tag{17}$$

$$n_P^G\Big|_{L_{SC},t} = k_{evap}\rho_P^{\circ}\frac{\rho_P^{SC}\Big|_{L_{SC},t}}{\rho_{sat}^{SC}} \tag{18}$$

where k_{evap} is the evaporative mass transfer coefficient describing penetrant flux into the ambient gas phase and ρ_{sat}^{SC} is the saturation concentration of the penetrant inside the SC. Using the expression given in Eq. (17), Eq. (15) can be rewritten as:

$$\rho_P^{\circ}\frac{dL}{dt} = -k_{evap}\rho_P^{\circ} + \frac{-\rho^{sc}D_{Psc}}{1-\omega_P^{SC}}\frac{\partial\omega_P^{sc}}{\partial z}\Bigg|_{z=L_{SC}} \tag{19}$$

2.3 Solution Methodology

Eqs. (1-19) were solved numerically using a finite difference/finite element based subroutine D03PPF from the NAG® (Numerical Algorithms Group) mathematical library (NAG®, 2006). The solution is obtained using the method of lines. The PDEs are discretized in space, and the resulting ODEs are integrated in time using the Gear method, which is suited for problems in which the gradients can become very large. Trial simulations were performed with increasing mesh sizes. A mesh size of 201 nodes was determined to have sufficient numerical accuracy and was used to generate the simulation results reported in this paper.

Although D03PPF can solve multiple PDEs, an important restriction lies in the fact that spatial location of the boundaries (on which boundary conditions are imposed) for all the individual PDEs in the system must be the same. They are represented in the manual as $x = a$ and $x = b$ (NAG®, 2006). This condition must be satisfied irrespective of the system of equations, even if they are distributed over several layers of different layer thicknesses. In the proposed model the transport equations are distributed over the participating layers VH, SC and VT, each having a different thickness. Hence, the model equations cannot be solved in their

original form. Eqs. (5-19) must first be rewritten through a process of compression and normalization of all the skin layers. This can be achieved by converting the spatial coordinate (z) from a uniform variable to a layer-specific variable, using the following set of transformation equations:

$$z^{SC} = \frac{z}{L_{SC}} \tag{20}$$

$$z^{VT} = -\frac{z}{L_{VT}} \tag{21}$$

Eqs. (20) and (21) can be used to develop specific equations relating the expressions for partial derivatives between the old and the new coordinate systems. A detailed derivation and exact expressions for these relationships can be found in Appendix A. Using these expressions, the model equations can be re-written for the transformed coordinate system. These equations have been given in Table 2.1.

2.4 Representative Results

A detailed validation of the proposed model requires real-time experimental data on skin permeation of pure volatile liquids. The data-acquisition technique that has been employed for studying the current percutaneous absorption process is described later in Chapters 4 & 5 of this dissertation. In this section, the model predictions and their implications are presented through representative simulations. The system behavior of the proposed model is similar to the KM model (Kasting and Miller, 2006) as well as related modeling efforts from other groups (Anissimov and Roberts, 1999, 2001; Fernandes et al., 2005). Hence, this section will present key simulation results in order to highlight the system behavior and will not revisit the model implications already laid down in the KM model (Kasting and Miller, 2006).

Table 2.1. Model Equations for Transformed Coordinate System

$$\frac{\partial \omega_P^{SC}}{\partial t} = \frac{1}{L_{SC}^2} \frac{\partial}{\partial z^{SC}} \left[\frac{D_{PSC}}{\left(1-\omega_P^{SC}\right)} \frac{\partial \omega_P^{SC}}{\partial z^{SC}} \right] \tag{29}$$

$$\frac{\partial \omega_P^{VT}}{\partial t} = \frac{1}{L_{VT}^2} \frac{\partial}{\partial z^{VT}} \left[\frac{D_{PVT}}{\left(1-\omega_P^{VT}\right)} \frac{\partial \omega_P^{VT}}{\partial z^{VT}} \right] \tag{29}$$

$$\omega_P^{VT}\Big|_{z^{VE},0} = 0 \tag{29}$$

$$\omega_P^{SC}\Big|_{z^{SC},0} = 0 \text{ for } 0 \le z^{SC} < 1 - f_{dep} \tag{29}$$

$$\omega_P^{SC}\Big|_{z^{SC},0} = \omega_{sat}^0 \text{ for } 1 - f_{dep} \le z \le 1 \tag{29}$$

$$\omega_P^{VE}\Big|_{1,t} = 0 \tag{10}$$

$$\omega_P^{SC}\Big|_{1,t} = K_{SC/VH}^P \frac{\rho_P^\circ}{\rho^{SC}} \text{ for } 0 \le t \le t_{depl} \tag{11}$$

$$\left. \frac{\rho^{SC} D_{PSC}}{L_{SC}\left(1-\omega_P^{SC}\right)} \frac{\partial \omega_P^{SC}}{\partial z^{SC}} \right|_{1,t} = k_{evap} \rho_P^\circ \left(\omega_P^{SC}\Big|_{1,t} \right) \text{ for } t \ge t_{depl} \tag{29}$$

$$\left. \frac{\rho^{VT} D_{PVT}}{L_{VT}\left(1-\omega_P^{VT}\right)} \frac{\partial \omega_P^{VT}}{\partial z^{VT}} \right|_{0,t} = \left. \frac{\rho^{SC} D_{PSC}}{L_{SC}\left(1-\omega_P^{SC}\right)} \frac{\partial \omega_P^{SC}}{\partial z^{SC}} \right|_{0,t} \tag{29}$$

$$\rho_P^\circ \frac{dL}{dt} = -k_{evap} \rho_P^\circ + \left. \frac{\rho^{SC} D_{PSC}}{L_{SC}\left(1-\omega_P^{SC}\right)} \frac{\partial \omega_P^{SC}}{\partial z^{SC}} \right|_{z^{SC}=1} \tag{29}$$

The skin permeation model developed here is capable of predicting the concentration profiles inside the SC and VT, the instantaneous VH thickness, instantaneous evaporation and absorption fluxes, and cumulative amounts (representing the fraction of the original dose) of evaporation and absorption flux, as well as other variables that are useful for interpreting the process. Each of these parameters has special significance depending on the particular field of study. The most relevant and important indicator for this study on occupational safety and health is the total amount of liquid penetrated through the skin, as given by the absorbed dose fraction. For studies pertaining to pharmacokinetic bioequivalence, several other parameters such as maximum absorption flux of the permeant (peak flux; $J_{\max}$) and the time required to reach this peak flux ($t_{\max}$) are important (Anissimov and Roberts, 2001). Furthermore, for toxicological studies pertaining to allergic contact dermatitis (ACD), the epidermal concentrations are critically important (more than the absorption flux or amount penetrated). Also, for studies pertaining to efficacy and product development of volatile ingredients in cosmetic formulations and insect repellants, the instantaneous evaporation flux is the most relevant parameter. Properties of the system, such as instantaneous thickness of the VH and concentration profiles of penetrant inside the SC and VT are extremely difficult to measure experimentally and results pertaining to these quantities are absent in the literature. Thus, it is nearly impossible to validate the model predictions for these quantities.

As discussed earlier, the absorbed fraction depends only on the ratio of the evaporative mass transfer coefficient and the overall permeability of the skin sub-layers. However, a change in the absolute value of the latter will change both $J_{\max}$ and $t_{\max}$. For a given permeant, the experimental parameters that can be varied freely are the initial amount of the liquid (also referred to as dose), the ambient airflow velocity and the system temperature. This article

restricts the representative simulations to variations in ambient and experimental conditions, but for a particular compound, ethanol. Ethanol was the diluent used in earlier studies from this laboratory of the less volatile compounds DEET (Santhanam et al., 2005) and benzyl alcohol (Miller et al., 2006). In order to interpret the binary system the binary system behavior in these studies, the kinetics of both solute and the solvent must be understood. A follow-up article (Ray Chaudhuri et al., 2007) provides details of the estimation of the thermodynamic and transport properties for the ethanol-skin system. The numerical values used herein (Table 2) were chosen to accurately represent the situation of ethanol permeation through human skin. The influence of the fractional deposition depth (f_{dep}) will be examined in some detail, as this parameter is relatively new concept and has not been extensively evaluated. The usefulness of f_{dep} will become evident in the accompanying analysis of ethanol skin permeation (Chapters 4 & 5).

2.4.1 Dose Dependence

The equations given in Table 2.1 reveal that the fraction of dose absorbed should depend on the evaporative tendency of the system in relation to its permeable tendency. In particular, when $f_{dep} = 0$, it should not depend on the amount applied (dose) but only on the relative values of the evaporative mass-transfer coefficient and the SC diffusivity of the penetrant, a ratio that is similar to the popular dimensionless group Biot number (*Bi*) for mass-transfer. In order to test this prediction, simulations were performed using the parameters in Table 2.2 and doses of 5, 50 and 500 μL/cm^2, which corresponds to an initial VH thicknesses of 50, 500 and 5000 μm. Fig. 2.2 contains the results. It shows that even though the shapes of the profiles are different, the fraction of dose absorbed was the same for all doses.

Table 2.2. Numerical Values of Physical and Transport Properties for the Ethanol/SC/VT System

Parameter	Value (units)
Pure ethanol density	0.7808 (g/cm^3)
Molecular weight of ethanol	46.07 (g/gmole)
Pure SC density	1.1936 (g/cm^3)
Thickness of the SC ($f_{dep} = 0$)	13.4 (μm)
Diffusivity of ethanol in SC	4.6455e-10 (cm^2/s)
Pure VT density	1.0190 (g/cm^3)
Thickness of the VT	500 (μm)
Diffusivity of ethanol in VT	5.7100e-6 (cm^2/s)
SC/VH partition coefficient of ethanol	0.3029
VT/SC partition coefficient of ethanol	2.3308
Evaporation mass-transfer coefficient for ethanol	0.3827 (cm/h)
Constant system temperature	32 (deg. C)

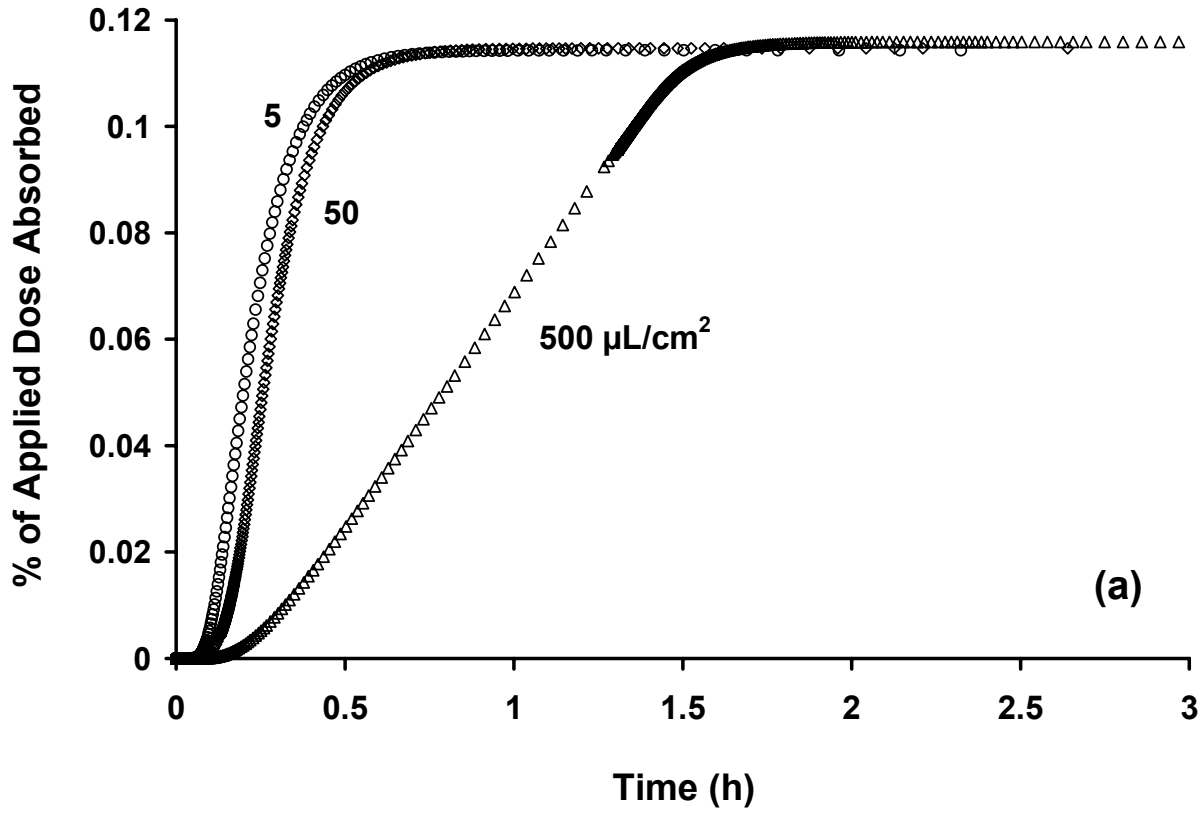


Fig. 2.2(a) Absorption of penetrant with varying doses (5, 50 and 500 µL/cm^2): fraction absorbed. f_{dep} was held constant at 0 for all runs.

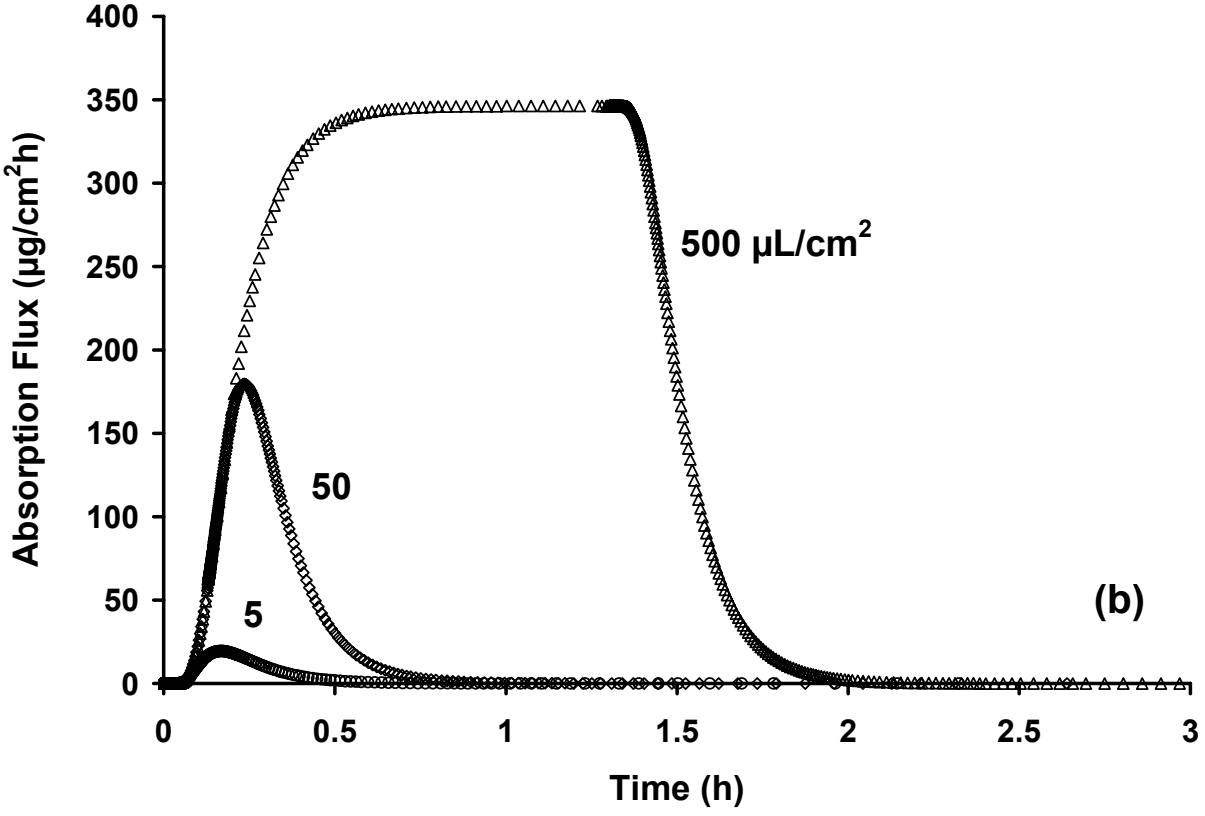


Fig. 2.2(b) Absorption of penetrant with varying doses (5, 50 and 500 µL/cm^2): instantaneous absorption flux. f_{dep} was held constant at 0 for all runs.

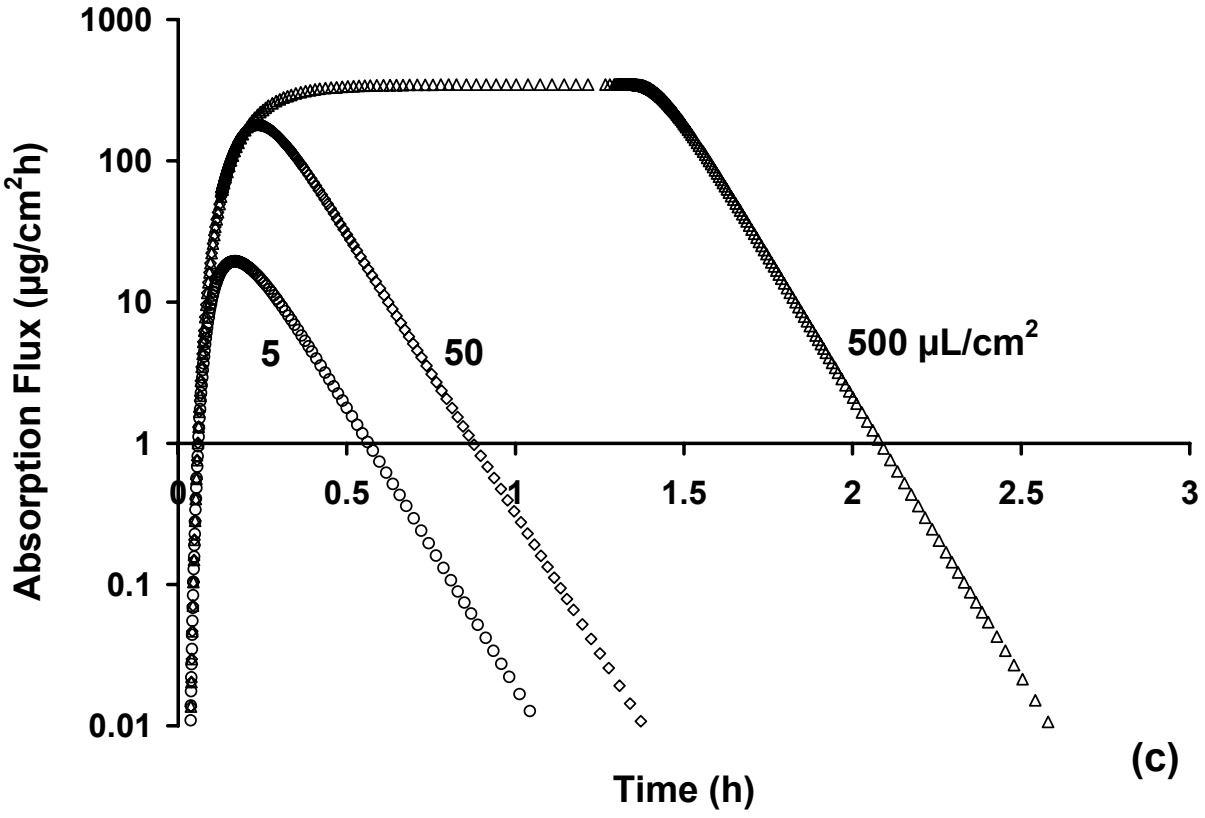


Fig. 2.2(c) Absorption of penetrant with varying doses (5, 50 and 500 µL/cm^2): instantaneous absorption flux plotted on a logarithmic scale. f_{dep} was held constant at 0 for all runs.

On the other hand, $J_{\max}$ and $t_{\max}$ change with varying doses. This has been shown in Fig. 2.2b, which plots the instantaneous absorption flux as it changes with time.

For large doses, the system may reach a steady-state, as shown in this figure for the 500 μL/cm^2 dose. Similar results have been obtained in prior modeling studies (Anissimov and Roberts, 1999, 2001; Kasting and Miller, 2006). Also, at long times the absorption flux should decay exponentially (Anissimov and Roberts, 1999, 2001). This behavior can be clearly seen in Fig 2.2c. As noted by Anissimov and Roberts (2001), the slope of the linear portion of such curves is a function only of the intrinsic permeabilities of the SC and VT. For a given permeant, the slope remains constant, even if the dose is varied. Fig 2.2c confirms this prediction.

When f_{dep} is non-zero, the fraction absorbed depends on dose, as shown in Figure 2.3. This result stems from the fact that small doses deposited into the upper SC evaporate more slowly than solvent residing in the VH layer. For very large doses, (greater than approximately 100 μL/cm^2), the system is insensitive towards the numerical value of f_{dep}. This reflects the fact the VH layer persists for an extended time under these conditions, so that the initial deposition of a small fraction of the solvent into the upper SC is inconsequential.

2.4.2 Deposition Depth Dependence

Fig. 2.4 shows the result for varying f_{dep} for a small applied dose (5 μL/cm^2). Although the value of *Bi* ($=k_{evap}/D_{PSC}$) was the same for all cases, the total absorbed fraction is not constant. The above simulation was repeated using a substantially larger dose (500 μL/cm^2). Figure 2.5 shows the results for both the small and large doses. Similar to finding in Fig. 2.3, f_{dep} does not have a major impact on the total absorbed fraction for the large dose.

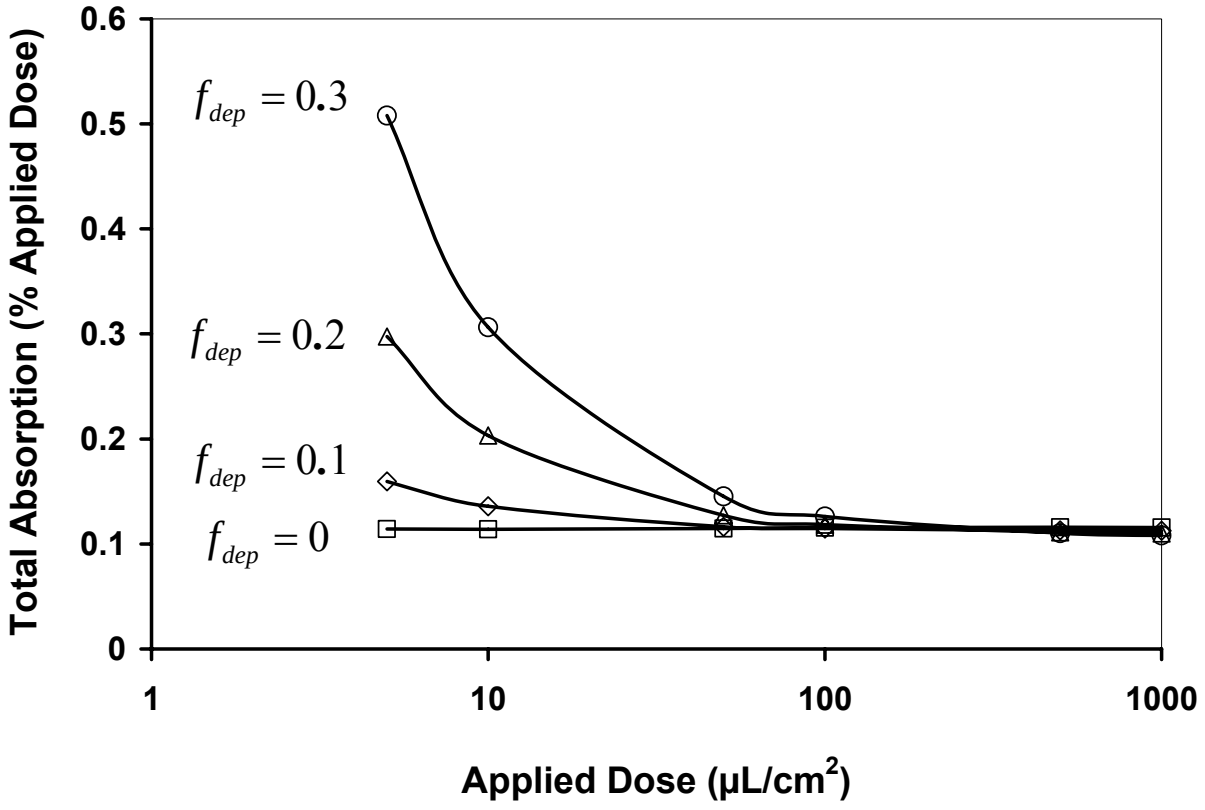


Fig. 2.3. Variation of total absorbed fraction with applied dose for varying values of deposition depth (f_{dep}). The applied dose values were 5, 50, 500 and 5000 µL/cm^2.

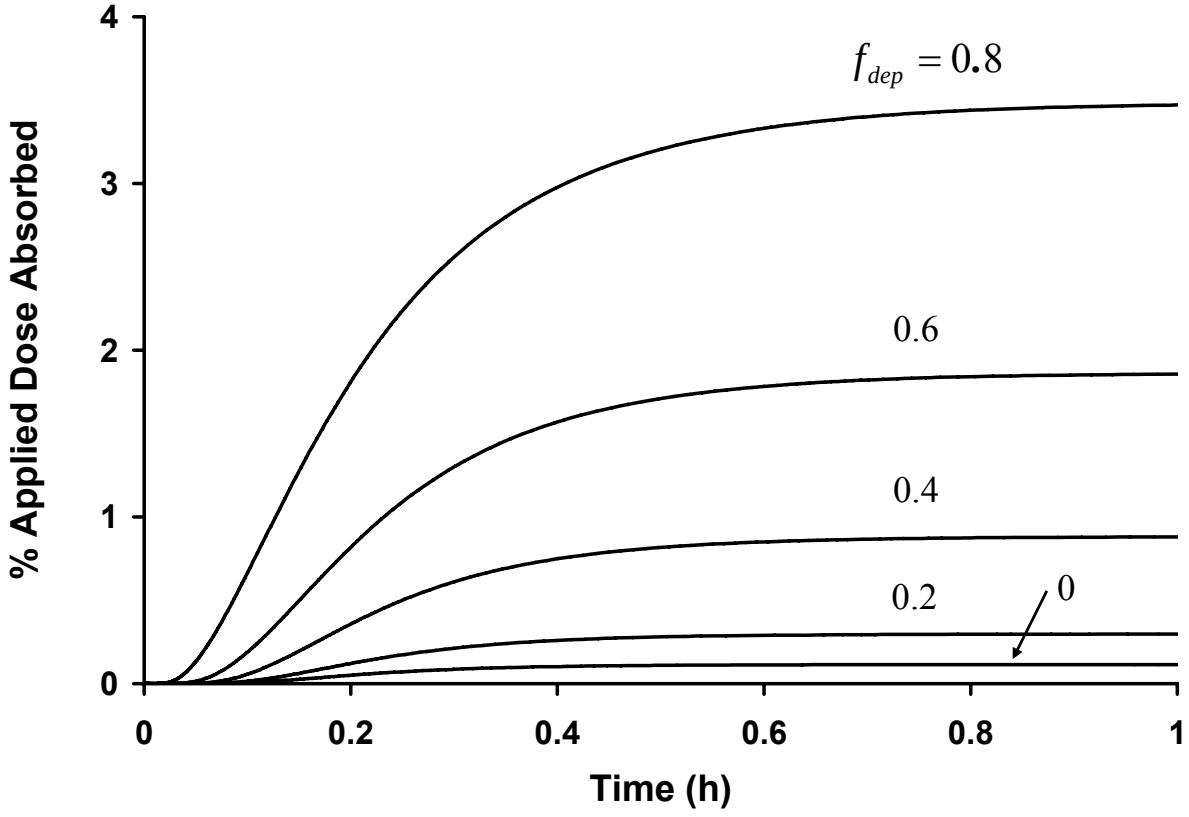


Fig. 2.4. Profiles for fraction dose absorbed with varying f_{dep} (5 μL/cm^2 dose)

Fig. 2.5 shows the maximum absorption of 5.5 % of applied dose for the 5 μL/cm^2 is obtained when f_{dep} = 1. This maximum value is clearly dose dependent.

2.4.3 Airflow Dependence

For a particular permeant and a constant temperature, the evaporation mass transfer coefficient (k_{evap}) is a function only of airflow velocity. The default value of k_{evap} representing laboratory conditions is 0.38 cm/h (as given in Table 2.1), corresponding to an airflow velocity of 0.74 m/s. This particular choice of airflow velocity has been discussed in detail in Chapter 4 of this dissertation. In these simulations, k_{evap} was varied to assume 0.1, 1 and 10 times the default value. The applied dose for these simulations was 50 μL/cm^2. Initially the value f_{dep} = 0 was chosen, which meant that an increase in k_{evap} should yield a proportional decrease in the total absorbed fraction. In other words, if the total absorbed fraction is plotted against the reciprocal of *Bi*, (both quantities on logarithmic scale) the result should be a straight line with slope = 1. This result is evident from the solid squares in Fig. 2.6. However, this proportional change is observed only when f_{dep} is zero. For non-zero f_{dep}, the relative absorption flux increase is less for a given value of $1/Bi$. This effect is more prominent for higher deposition depths.

2.5 Conclusion

A mathematical model representing transient mass-transfer pertaining to skin permeation of pure volatile liquids is presented in this paper. For small doses, the model is sensitive to a

parameter defined as the fractional deposition depth of permeant in the SC (f_{dep}), which mathematically appears in the initial condition of the problem. The system behavior is found to

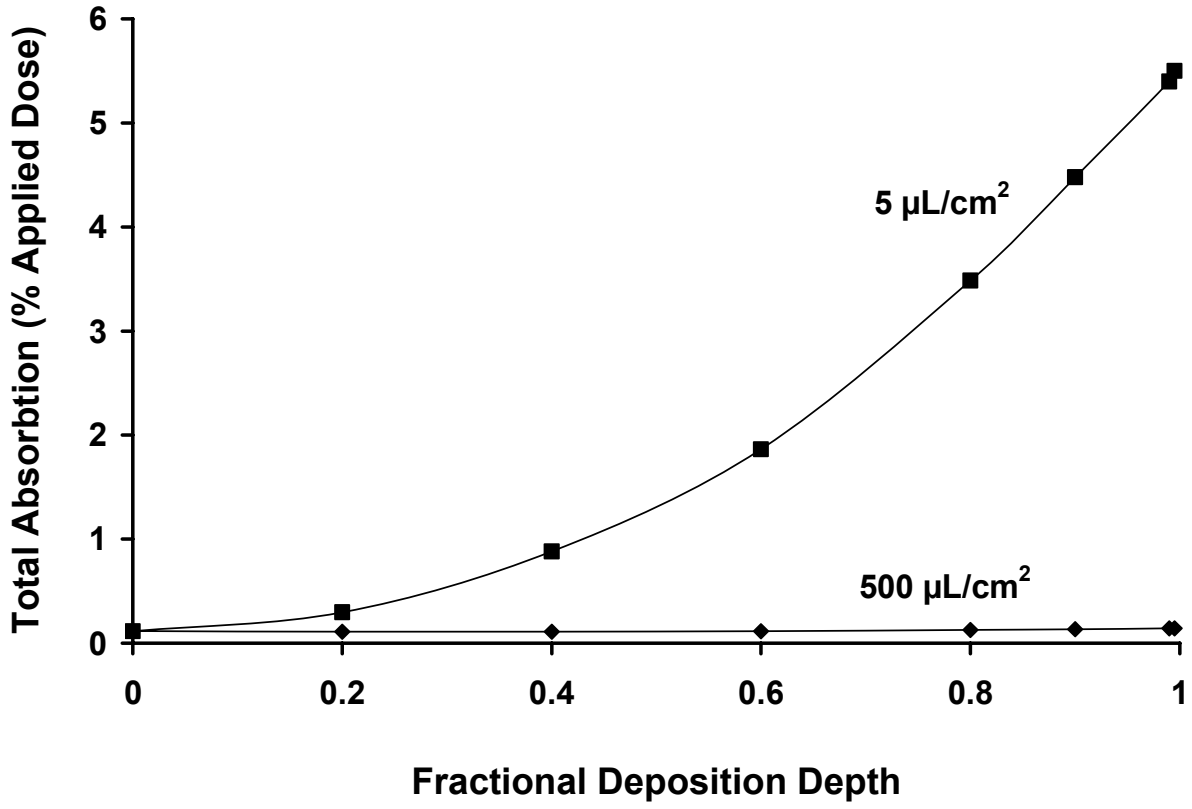


Fig. 2.5. Total absorbed fraction as a function of f_{dep} for small (5 µL/cm^2) and large doses (500µL/cm^2)

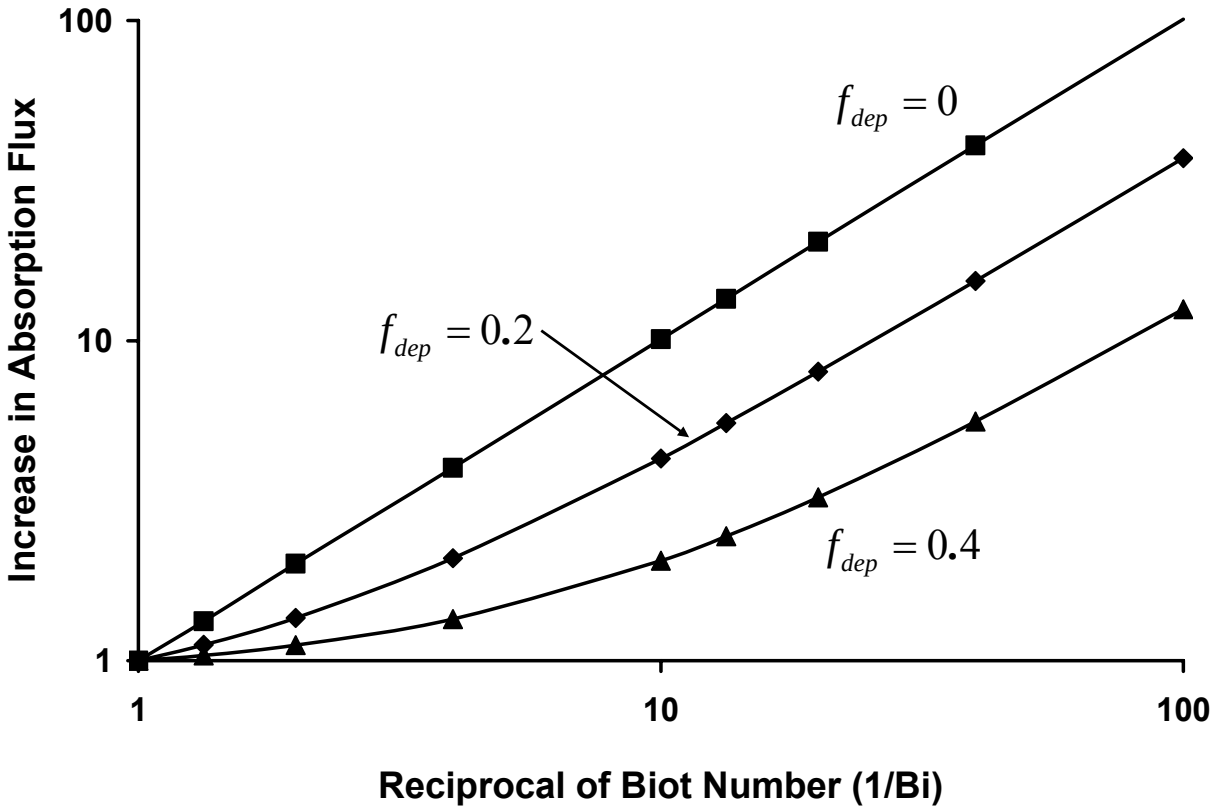


Fig. 2.6. Variation in absorption flux with *Bi* for different values of f_{dep} (50 μL/cm^2) dose).

be in accordance with the predictions of scaling analysis as well as with several analytical expressions available in the literature. The simulation results indicated that the total amount of a topically applied compound permeating through the skin depends on the temperature, the vapor pressure of the permeant, the ambient airflow velocity, the permeability of the SC and VT and on f_{dep}. For a given permeant, the fraction of dose absorbed depends only on system temperature, airflow velocity, dose and f_{dep}. The proposed model can be used to predict percutaneous disposition characteristics for pure volatile liquids and can also be extended to non-volatile liquids.

Alsoy, S., Duda, J.L. (2002). Influence of swelling and diffusion-induced convection on polymer sorption processes. AIChE Journal, 48 (9), 1849-1855

Anissimov, Y.G., Roberts, M.S. (1999). Diffusion modeling of percutaneous absorption kinetics. 1. Effects of flow rate, receptor sampling rate, and viable epidermal resistance for a constant donor concentration. Journal of Pharmaceutical Sciences, 88 (11), 1201–1209.

Anissimov, Y.G., Roberts, M.S. (2001). Diffusion modeling of percutaneous absorption kinetics: 2. finite vehicle volume and solvent deposited solids. Journal of Pharmaceutical Sciences, 90 (4), 504–520.

Anissimov, Y.G., Roberts, M.S. (2004). Diffusion modeling of percutaneous absorption kinetics: 3. variable diffusion and partition coefficients, consequences for stratum corneum depth profiles and desorption kinetics. Journal of Pharmaceutical Sciences, 93 (2), 470–487.

Barbero, A.M., Frasch, H.F. (2005). Modeling of diffusion with partitioning in stratum corneum using a finite element model. Annals of Biomedical Engineering, 33, 1281-1292.

Batterman, S.A., Franzblau, A. (1997). Time-resolved cutaneous absorption and permeation rates of methanol in human volunteers. International Archives of Occupational and Environmental Health, 70, 341-351.

Bear, J, Bachmat, Y. (1990). Introduction to Modeling of Transport Phenomena in Porous media, Kluwer Academic Publishers, Boston.

Bird, B.R., Stewar,t W.E., Lightfoot, E.N., (1960). Transport Phenomena, John Wiley and Sons Inc., New York.

Blank, I. H., (1964). Penetration of low–molecular–weight alcohols into skin. I. effect of concentration of alcohol and type of vehicle. Journal of Investigative Dermatology, 43, 415-420.

Bouwstra, J. A., Honeywell-Nguyen, P.L., Gooris, G.S., Ponec, M., (2002). Structure of the skin barrier and its modulation by vesicular formulations. Progress in Lipid Research, 42, 1–36.

Cleek, R.L., Bunge, A.L., (1993). A new method for estimating dermal absorption from chemical exposure. 1. General approach. Pharmaceutical Research, 10, 497–506.

Deen, W.M., (1998). Analysis of Transport Phenomena, Oxford University Press, New York.

Del-Valle, V., Almenar, E., Hernández-Muñoz, P., Lagarón, J. M., Catala, R., Gavara, (2004). Volatile organic compound permeation through porous polymeric films for modified atmosphere packaging of foods. Journal of the Science of Food and Agriculture, 84, 937-942.

Fernandes, M., Simon, L., Loney, N.W., (2005). Mathematical modeling of transdermal drug-delivery systems: Analysis and applications. Journal of Membrane Science, 256, 184-192.

Flynn, G.L., (1990). Physicochemical determinants of skin absorption, in: Gerritv, T.R. & Henry, C.J. (Eds.), Principles of Route-to-Route Extrapolation for Risk Assessment, Elsevier, New York, pp. 93–127.

Frasch, H.F. (2002). A random walk model of skin permeation. Risk Analysis, 22, 265-276.

Frasch, H.F., Barbero, A.M. (2003). Steady-state flux and lag time in the stratum corneum lipid pathway: results from finite element models. Journal of Pharmaceutical Sciences, 92, 2196-2207.

Geinoz, S., Guy, R.H., Testa, B., Carrupt, P.-A., (2004). Quantitative structure–permeability relationships (QSPeRs) to predict skin permeation: a critical evaluation. Pharmaceutical Research, 21 (1), 83–92.

Jakasa, I., Mohammadi, N., Kruse, J., Kezic, S.. (2004). Percutaneous absorption of neat and aqueous solutions of 2-butoxyethanol in volunteers. International Archives of Occupational and Environmental Health, 77, 79-84.

Kasting, G.B. (2001). Kinetics of finite dose absorption through skin 1. vanillylnonanamide. Journal of Pharmaceutical Sciences, 90, 202-212.

Kasting, G.B., Miller, M.A. (2006). Kinetics of finite dose absorption through skin 2. Volatile compounds. Journal of Pharmaceutical Science, 95, 268–280.

Kezic, S., Monster, A.C., Kruse, J., Verberk, M.M., (2000). Skin absorption of some vaporous solvents in volunteers. International Archives of Occupational and Environmental Health, 73, 415-422.

Korinth, G., Geh, S., Schaller, K.H., Drexler, H., (2003). In vitro evaluation of the efficacy of skin barrier creams and protective gloves on percutaneous absorption of industrial solvents. International Archives of Occupational and Environmental Health, 76, 382-386.

Lee, H, Ray Chaudhuri, S., Krantz, W.B., Hwang, S.-T., (2006). A model for evaporative casting of polymeric membranes incorporating convection due to density changes. Journal of Membranes Science, 284, 161-172.

Lockley, D.J., Howes, D., Williams, F.M., (2004). Percutaneous penetration and metabolism of 2-butoxyethanol. Archives of Toxicology, 78, 617-628.

McCarley, K.D., Bunge, A.L. (2001). Pharmacokinetic models of dermal absorption. Journal of Pharmaceutical Sciences, 90 (11), 1699–1719.

Menon, G. K., (2002). New insights into skin structure: scratching the surface. Advance Drug Delivery Reviews, 54 (Suppl. 1), S3–S17.

Miller, M.A., Bhatt, V., Kasting, G.B., (2006). Dose and airflow dependence of benzyl alcohol disposition on skin. Journal of Pharmaceutical Sciences, 95 (2), 281–291.

Montagna, W., Kligman, A.M., Carlisle, K.S., (1992). Atlas of Normal Human Skin. Springer–Verlag, Berlin and Heidelberg.

Moss, G.P., Dearden, J.C., Patel, H., Cronin, M.T., (2002). Quantitative structure–permeability relationships (QSPRs) for percutaneous absorption. Toxicology in Vitro, 16, 299–317.

Mráz, J., Nohová, H., (1992). Percutaneous absorption of N,N-dimethylformamide in humans. International Archives of Occupational and Environmental Health, 64, 79-83.

Potts, R.O., Guy, R.H., (1992). Predicting skin permeability. Pharmaceutical Research, 9 (5), 663–669.

Potts, R. O., Francoeur, M. L., 1991. The influence of stratum comeum morphology on water permeability. Journal of Investigative Dermatology, 96, 495-499.

Roberts, M. S., Walters, K.A. (1998). The relationship between structure and barrier function of skin, in: Roberts, M.S. & Walters, K.A. (Eds.), Marcel Dekker Inc., New York.

Santhanam, A., Miller, M.A., Kasting, G.B., (2004). Absorption and evaporation of N, N-diethyl-m-toluamide from human skin in vitro. Toxicology and Applied Pharmacology, 204, 81-90.

Scheuplein (1965). Mechanism of percutaneous drug absorption. I: routes of penetration and the influence of solubility, Journal of. Investigative Dermatology, 45, 334–345.

Scheuplein, R.J., Blank, I.H. (1971). Permeability of the skin, Physiological Reviews, 51, pg 702.

Semple, S., Brouwer, D.H., Dick, F., Cherrie, J.W., (2001). A dermal model for spray painters. part II: estimating the deposition and uptake of solvents. Annals of Occupational Hygiene, 45 (1), 25-33.

Singh, S., Singh, J., (1993). Transdermal drug delivery by passive diffusion and iontophoresis: a review. Medicinal Research Reviews, 13, 569–621.

Stinchcomb, A.L., Pirot, F., Touraille, G.D., Bunge, A.L., Guy, R.H., (1999). Chemical uptake into human stratum corneum in vivo from volatile and non-volatile solvents. Pharmaceutical Research, 16 (8), 1288-1293.

Wilkinson, S.C., Williams, F.M. (2002). Effects of experimental conditions on absorption of glycol ethers through human skin in vitro. International Archives of Occupational and Environmental Health, 75, 519-527.

Yamashita, F., Hashida, M., (2003). Mechanistic and empirical modeling of skin permeation of drugs. Advanced Drug Delivery Reviews. 55, 1185–1199.

CHAPTER 3

SCALING (APPROXIMATION) ANALYSIS OF TRANSIENT MASS TRANSPORT THROUGH MULTILAMELLAR MEMBRANE

3.1 Background

3.1.1 Modeling Membrane Permeation

The principal property of membranes is the ability to control the permeation of different species. Two distinct theoretical approaches are used to describe this permeation process (Wijmans and Baker, 1995). The first is the pore-flow model, in which permeants are separated by pressure-driven flow through tiny pores. The second is the solution-diffusion model consisting of three different steps. At first, the permeating component is absorbed into the membrane, followed by its diffusion through the membrane and, finally, desorption from the membrane (Dijkstra et al., 2006). These models achieve separation between different permeants based on different properties. For the pore flow model, the differences in size of the permeants cause separation, while the differences in solubility and diffusivity inside the membrane causes separation according to the solution-diffusion model. Both models were proposed in the nineteenth century, but the solution-diffusion model has been established as an appropriate description of transport through dense membranes (Wijmans and Baker, 1995). Membranes can simply serve as semi-permeable or selectively permeable barriers when separation is not desired. Even then, the permeating species follows the solution-diffusion mechanism. Such membrane barriers can either be artificial or natural. Other than the commercially manufactured barriers, since almost all life forms depend upon membrane–like structures for their existence, there are several examples of natural occurring barriers such as skin, gastrointestinal tract and lipid bilayer membranes. Considering the different kinds of membranes (natural and artificial), there are literally thousands of chemical and biological compounds for which the relevant barrier functions can be important. It is impossible to experimentally measure the permeation characteristics of each substance through individual barriers. Thus, researchers have attempted to

develop predictive mathematical models that have the ability to estimate the transient and steady-state permeation characteristics for an arbitrary compound through an arbitrary membrane. The accuracy and predictive power of such models is critically dependent on a judicious choice of system parameters, which requires a thorough understanding of the physical properties and chemical structure of the permeant and that of the membrane. Even though these parameters are widely different for different systems (class of compound and/ or nature of membrane), the fundamental mathematical equations describing solution and diffusion through a barrier remain invariant. These equations can be obtained from the basic principles of conservation of mass without the use of adjustable parameters or process specific factors. The generalized equation representing transient mass transport is given by the conservation-of-species equation. It can be written as (Bird et al., 2001):

$$\rho\left(\frac{\partial \omega_i}{\partial t} + W\frac{\partial \omega_i}{\partial z}\right) = -\frac{\partial j_i}{\partial z} \pm r_i \quad \text{for } i = 1,2,...,m \tag{30}$$

In Eq. (30), t represents time and z the spatial coordinate, ρ is the total solution density, ω_i and j_i are the mass fraction and diffusive mass flux of species i, respectively. The first term in Eq. (30) represents unsteady-state accumulation while the second term represents convective mass flux, expressed through the mass-average velocity (W). r_i represents the volumetric generation or consumption rate of species i in terms of mass units. Under normal circumstances, membrane permeation does not involve a generation or consumption term. However, for specialized situations such as transport through membrane reactors or permeation through human skin it can be quite relevant. Particularly, for skin permeation, a network of blood capillaries in the dermal layer of skin systematically clears the permeant and this can be equivalently expressed through a consumption rate in the mass transport equation. Under steady–state conditions and in the

absence of convection or generation or consumption, the above partial differential equation (PDE) reduces to a simple ordinary differential equation (ODE):

$$\frac{\partial j_i}{\partial z} = 0 \tag{31}$$

For appropriate boundary conditions, Eq. (31) can be easily integrated by a host of analytical methods (Bird et al., 2001; Crank, 1975; Welty et al., 2001) resulting in a closed-form expression for the steady-state permeation flux according to the solution-diffusion model. However, for unsteady-state processes that are represented by Eq. (30), the possibilities of analytical solutions are limited. Even if a closed-form solution is possible, it requires the use of complex mathematical techniques such as integral transforms. A large repository of analytical solutions to a host of representative PDEs can be found in the works of Crank (1975), Carslaw and Jaeger (1986) and Mikhailov and Ozisik (1984). In the absence of an analytical solution, Eq. (30) can be integrated numerically. Most numerical algorithms are expressed in terms of computer programs and with the general accessibility of high–speed digital computers it is now possible to obtain rapid and accurate solutions to several complex problems.

3.1.2 Approximation Analysis

Irrespective of method the solution (analytical or numerical), an approximation analysis of the model differential equation(s) (representing barrier permeation) is important. Such an analysis will result in a (possibly) simplified form of the mathematical equations whose representation of a physical process will be equivalent to that of the non–simplified comprehensive equations, without any significant loss of accuracy. A simplified set of equations would result in a quicker and more efficient numerical solution. Also, when the comprehensive equations owing to their complexity, preclude the possibility of any analytical solution, the

corresponding simplified differential equations may lead to an analytical solution. The merit of approximation analysis on model equations has been identified by several researchers. Bear and Bear and Bachmat (1991) referred to this process as "deletion of non–dominant effects", while others have used terminology such as 'scaling' (Tosun, 2002) or 'scaling analysis' (Krantz, 1970). However, they have not been extensively used in the literature.

One of the first steps in an approximation analysis involves non–dimensionalization of the describing equations. It should be noted that dimensionless (and semi–dimensionless) representation of transport equations is quite common in the engineering textbooks (Bird et al., 2001; Crank, 1975; Mikhailov and Ozisik , 1984; Welty et al., 2001), even when an approximation analysis is not explicitly desired. Welty et al. (2002) attempted to develop simple analytical solutions to transport problems by converting only the dependent variables to a dimensionless form. They also performed dimensional analysis by looking at geometric and kinematic similarities between different systems, for the purpose of scaling up bench-scale processes to industrial scale. However, they did not perform any systematic approximation analysis. Bird et al. (2001) expressed the transport equations in pure dimensionless form. This process of non-dimensionalization led to the formation of dimensionless groups or numbers. It will be clear from this section that these dimensionless groups play an important role in the understanding of the physical significance of each aspect of transport processes. If the process of non–dimensionalization is performed in a systematic way it can also be used to simplify transport equations by scientifically justifying different approximations or assumptions. This would require a judicious choice of characteristic variables that are used to non-dimensionalize the process variables. Conventionally, the process of selecting characteristic variables is arbitrary and rarely any mechanistic explanation is given for such choices. Moreover, even though the use

of dimensionless groups or numbers is quite popular in engineering analysis, their systematic development through the process of non–dimensionalization is rarely encountered.

The first systematic method of scaling a problem to determine when a simplified set of equations may describe a process had been described in an article given by Hellums and Churchill (1964). Although their section is primarily concerned with using this method to determine similarity variables for PDEs, this tool was not applied globally to simplifying describing differential equations. Krantz (1970) built on the ideas put forward in this section to devise a systematic method for the non–dimensionalization of mathematical equations describing transport (and chemical reaction) processes, eventually leading up to an approximation analysis. This work employed an o(1) scaling analysis, a unique method of generating a dimensionless system of equations that comprise of a minimum parametric representation of the physical process. This means that the solution for any quantity obtained from these equations will be at most a function of the dimensionless independent variables and the dimensionless groups generated by the scaling analysis. These dimensionless groups (or numbers) that are used frequently in chemical engineering analyses (mainly in correlations) play a critical role in this process. o(1) scaling analysis has several other potential applications. It identifies the dimensionless variables and groups that dictate the inherent physics of the process, thereby supplying valuable information on the system dynamics and parametric sensitivity. It can be used to correlate data from either laboratory or numerical experiments (*i.e.*, computer simulations). The resulting dimensionless groups can also be used for scale–up or scale–down analyses by invoking the principles of geometric and dynamic similarity. It is also useful for developing perturbation expansion solutions to the describing equations. Also, in determining the scales and expansion parameters in perturbation analyses, it can be used in assessing potential problems that

can occur in solving a system of describing equations numerically. Also, scaling analysis serves as a valuable tool both for educators as well as students in communicating the fundamental similarity for different transport processes. Krantz (1970) treated a variety of transport problems and highlighted on the strength of the systematic process of scaling. The evolution of different dimensionless groups and their physical significance was also discussed in detail. The purpose of this section was mainly pedagogical, aimed at chemical engineering graduate students for a better understanding of transport processes. It should be noted that the original work of Krantz (1970) was purely a theoretical analysis and the conclusions made through scaling analysis were not substantiated through either experimental measurements or computer simulations. It is also true that the state–of–the on scaling or similar approximation analysis, including the work of Krantz (1970), focus mainly on fluid flow or heat transfer problems and rarely on mass transport equations. Mass transfer is unique and different from heat or momentum transfer and this uniqueness is most prominent in multicomponent (binary and higher) systems. For instance, in mass transport problems, each chemical species has a separate identity (in having its own concentration), whereas in heat transfer, individual species have the same temperature, which is equal to the uniform system temperature. This difference between transport of mass and other entities has been identified for quite some time and has been mentioned in (Tosun, 2002). Thus, the process of approximation analysis needs to be extended from the usual fluid flow and heat transfer problems and applied rigorously to mass–transfer processes.

3.1.3 Objectives

The main objective of this section is to apply o(1) scaling analysis to a practical mass-transfer problem pertaining to absorption and transport of pure liquids through a multi-lamellar

membrane. Practical instances of such problems are common, but the most relevant example can be found in human skin. Skin is the largest organ of the human body, covering a surface area of approximately 2 m^2 and receiving about one-third of the total circulating blood (Singh and Singh, 1993). Owing to its immediate proximity to the environment, semi-permeable human skin gets naturally exposed to a variety of chemical compounds. Thus, it can serve as a possible entry point for a host of therapeutic substances, which serves as the core concept in transdermal drug delivery. However, many of these substances have the potential of interacting adversely with the skin or with internal physiological systems. Hence, the estimation of skin penetration rates and systemic absorption of compounds following incidental or intentional application to the skin is an important aspect of occupational risk assessment and toxicological studies. The degree of risk associated with the exposure depends on the amount and the nature of the chemical substance(s) involved. Some of them are relatively benign (ethanol, acetone etc.), some are potentially hazardous (pesticides, herbicides, fertilizers etc.) while some are extremely toxic and are lethal (chemical warfare agents). For compounds having high exposure levels and/or high potential toxicity, extensive experimental studies of dermal absorption, metabolism, and toxicity can be justified. However, as mentioned above, the number of chemicals to which the population is exposed on a daily basis is so high that only a small fraction can be studied experimentally (Basketter et al., 1996; Robinson et al., 2000; Robinson, 1998). Hence, computational models for dermal absorption are increasingly used in lieu of experimental studies to estimate absorption of new ingredients and such a predictive approach is widely used in many industries (Robinson et al., 2000; Bunge et al., 1994;Gerbrick and Robinson, 2000). Thus, in an attempt to apply scaling analysis to the transient solution-diffusion problem in a multi-lamellar membrane, this section will consider the example of penetration of pure liquids through human skin. However, this does

not restrict the scope of the ensuing scaling analysis and it can be applied to any system involving absorption and diffusion through multi-lamellar barriers that can be modeled using similar mathematical equations.

Another principal objective of this section involves validating the conclusions emanating from scaling analysis. This can be done by comparing the predictions of scaling analysis for this situation against the detailed numerical simulation of the comprehensive model equations (devoid of any assumptions). Collectively, this process of applying o(1) scaling analysis to the membrane transport problem and validating the predictions made by scaling analysis through numerical simulations of the comprehensive model is the novelty of this section.

3.2 o(1) Scaling Analysis of Model Equations

3.2.1 Background of o(1) Scaling

The stepwise procedure for systematic ○(1) scaling analysis is given in Table 3.1. An in–depth discussion of the significance and relative importance of each step can be found in (Krantz, 2007). In general, there is no unique set of dimensionless variables and groups for a given system of equations. However, one can scale a system of equations in a unique way to ensure that the relevant dependent and independent variables and their derivatives are bounded of order one. Order one in the present context means that the magnitude of the particular dimensionless variable or their derivatives are bounded between 0 and 1. This is the most critical aspect of ○(1) scaling analysis. Since ○(1) scaling ensures that the dependent and independent variables and their derivatives are bounded between 0 and 1, it means that one can assess the importance of various terms in the resulting dimensionless describing equations on the basis of the value of the

Table 3.1. Stepwise Procedure for o(1) Scaling Analysis

1. Write the dimensional describing equations and their initial and boundary conditions appropriate to the transport or reaction process being considered.
2. Define unspecified scale factors for each dependent and independent variable as well as appropriate derivatives appearing explicitly in the describing equations and their initial, boundary, and auxiliary conditions.
3. Define unspecified reference factors for each dependent and independent variable that is not referenced to zero in the initial, boundary, and auxiliary conditions.
4. Form dimensionless variables by introducing the unspecified scale factors and reference factors for the dependent and independent variables and the appropriate derivatives.
5. Introduce these dimensionless variables into the describing equations and their initial, boundary, and auxiliary conditions.
6. Divide through by the dimensional coefficient of one term (preferably one that will be retained) in each of the describing equations and their initial, boundary, and auxiliary conditions.
7. Determine the scale and reference factors by insuring that the principal terms in the describing equations and initial, boundary, and auxiliary conditions are ○(1); *i.e.*, they are bounded between zero and of order one.
8. The preceding steps result in the minimum parametric representation of the problem (*i.e.*, in terms of the minimum number of dimensionless groups); appropriate simplification of the describing equations now can be explored.

dimensionless groups that are attached to each of these terms. If a dimensionless group is of order 0.01 or less, the term attached to it (representing a particular aspect of the transport process) can be ignored for the particular process while incurring only a very small (~1%) error, which is safely within most of the acceptable engineering standards. However, if the magnitude of a particular dimensionless group is ~ 1, then the terms attached to it are important and cannot be neglected. Thus, by using ○(1) scaling one can appropriately simplify the describing equations for a process.

3.2.2 Non-dimensionalization of Model Equations

For the given problem, the independent and dependent variables are divided by unknown scale parameters to convert them into dimensionless forms. They are collectively written as:

$$t^* = \frac{t}{t_s},\ z^*_{SC} = \frac{z}{z_{SCs}},\ z^*_{VT} = \frac{z}{z_{VTs}},\ \omega^*_{PSC} = \frac{\omega_P^{SC}}{\omega_{PSCs}},\ \omega^*_{PVT} = \frac{\omega_P^{VT}}{\omega_{PVTs}} \text{ and } L^* = \frac{L}{L_s} \tag{32}$$

Even though mass fraction is a dimensionless quantity, in order to restrict its value between 0 and 1, a separate scale factor (for each layer) has been introduced. Moreover, the interfacial displacement velocity should be scaled with an independent scale factor (and not as the thickness scale divided by time scale). This scale factor will be determined by the dominant mechanism between absorption and evaporation that is mainly responsible for the shrinking of the LQ layer. Thus:

$$\left(\frac{dL}{dt}\right)^* = \frac{1}{V_s}\frac{dL}{dt} \tag{33}$$

The dimensionless variables are then substituted in Eqs. (5 & 6) to obtain:

$$\left(\frac{z_{SCs}^2}{t_s D_{PSC}}\right)\frac{\partial \omega^*_{PSC}}{\partial t^*} - \frac{\omega_{PSCs}}{\left(1-\omega_{PSCs}\omega^*_{PSC}\right)}\left(\frac{\partial \omega^*_{PSC}}{\partial z^*_{SC}}\right)^2 = \frac{\partial^2 \omega^*_{PSC}}{\partial {z^*_{SC}}^2} \tag{34}$$

$$\underbrace{\left(\frac{z_{VTs}^2}{t_s D_{PVT}}\right)\frac{\partial \omega_{PVT}^*}{\partial t^*}}_{\text{unsteady-state}} - \underbrace{\frac{\omega_{PVTs}}{\left(1-\omega_{PVTs}\omega_{PVT}^*\right)}\left(\frac{\partial \omega_{PVT}^*}{\partial z_{VT}^*}\right)^2}_{\text{convection}} = \underbrace{\frac{\partial^2 \omega_{PVT}^*}{\partial z_{VT}^{*\,2}}}_{\text{diffusion}} \tag{35}$$

The ICs and BCs are non-dimensionalized as

$$\text{At } t^* = 0,\ \omega_{PSC}^* = \omega_{PVT}^* = 0 \tag{36 \& 37}$$

$$\text{At } z^* = \frac{L_{sc}}{z_{SCs}}, \text{ for } 0 \le t^* \le \frac{t_{depl}}{t_s}, \text{ the condition is } \omega_{PSC}^*\Big|_{\frac{L_{sc}}{z_{SCs}},\frac{t}{t_s}} = \frac{K_{SC/LQ}^P \rho_P^\circ}{\omega_{PSCs}\rho^{SC}} \tag{38}$$

$$\text{and for } t^* \ge \frac{t_{depl}}{t_s}, \text{ it is } \frac{1}{\left(1-\omega_P^{SC}\right)^*}\frac{\partial \omega_{PSC}^*}{\partial z_{SC}^*} = \left(\frac{k_{evap}\rho_P^\circ z_{SCs}}{\rho^{SC} D_{PSC}\omega_{PSCs}}\right)\omega_{PSC}^*\Big|_{\frac{L_{sc}}{z_{SCs}},\frac{t}{t_s}} \tag{39}$$

$$\text{At } z_{SC}^* = z_{VT}^* = 0,\ \omega_{PSC}^*\Big|_{0,\frac{t}{t_s}} = \left(\frac{\rho^{VT}\omega_{PVTs}}{K_{VT/SC}^P \rho^{SC}\omega_{PSCs}}\right)\omega_{PVT}^*\Big|_{0,\frac{t}{t_s}} \tag{40}$$

$$\text{and } \frac{\rho^{SC} D_{PSC}\omega_{PSCs}}{z_{SCs}\left(1-\omega_P^{SC}\right)^*}\frac{\partial \omega_{PSC}^*}{\partial z_{SC}^*} = \frac{\rho^{VE} D_{PVT}\omega_{PVTs}}{z_{VTs}\left(1-\omega_P^{VE}\right)^*}\frac{\partial \omega_{PVT}^*}{\partial z_{VT}^*} \text{ or}$$

$$\frac{1}{\left(1-\omega_P^{SC}\right)^*}\frac{\partial \omega_{PSC}^*}{\partial z_{SC}^*} = \left(\frac{\dfrac{\rho^{VE} D_{PVT}\omega_{PVTs}}{z_{VTs}}}{\dfrac{\rho^{SC} D_{PSC}\omega_{PSCs}}{z_{SCs}}}\right)\frac{1}{\left(1-\omega_P^{VT}\right)^*}\frac{\partial \omega_{PVT}^*}{\partial z_{VT}^*} \tag{41}$$

$$\text{At } z_{VT}^* = -\frac{L_{VT}}{z_{VTs}},\ \omega_{PVT}^*\Big|_{-\frac{L_{VT}}{z_{VTs}},\frac{t}{t_s}} = 0 \tag{42}$$

Finally, the coupled ODE is non-dimensionalized as:

$$\left(\frac{dL}{dt}\right)^* = -\left(\frac{k_{evap}}{V_s}\right) + \left(\frac{\rho^{SC} D_{PSC}\omega_{PSCs}}{z_{SCs}\rho_P^\circ V_s}\right)\left[\frac{1}{\left(1-\omega_P^{SC}\right)^*}\frac{\partial \omega_{PSC}^*}{\partial z_{SC}^*}\right]_{z^*=\frac{L_{SC}}{z_{SCs}}} \tag{43}$$

with the corresponding dimensionless initial condition:

$$\text{at } t^* = 0,\ L^* = \frac{L_0}{L_s} \tag{44}$$

3.2.3 Determination of Scale Factors

For the current problem, the following considerations dictate the determination of the unknown scale factors. In order to restrict the dimensionless spatial coordinate and film thickness to be of ○(1) we obtain $z_{SCs} = L_{SC}$, $z_{VTs} = L_{VT}$ and $L_s = L_0$. The time scale is the observation time; that is $t_s = t_\circ$. The scale factor for penetrant mass fraction in the SC can be obtained from Eq. (38) and written as:

$$\omega_{PSCs} = \frac{K^P_{SC/LQ}\rho^\circ_P}{\rho^{SC}} = \omega^{\max}_{SC} \text{ (say)} \tag{45}$$

Similarly, the scale factor for penetrant mass fraction in VT is obtainable from Eq. (40) and can be written as:

$$\omega_{PVTs} = \frac{K^P_{VT/SC}\rho^{SC}\omega_{PSCs}}{\rho^{VT}} = \frac{K^P_{VT/SC}K^P_{SC/LQ}\rho^\circ_P}{\rho^{VT}} = \omega^{\max}_{VT} \text{ (say)} \tag{46}$$

The scale factor for V_s can be obtained from Eq. (44). If the permeating compound is volatile in nature or the ambient airflow velocity is substantially high, then evaporation will be mainly responsible for the thinning of the LQ layer and $V_s = k_{evap}$. On the other hand, if its volatility is very low, then absorption will dictate film thinning. In this case, we have:

$$V_s = \frac{\rho^{SC}D_{PSC}\omega_{PSCs}}{z_{SCs}\rho^\circ_P} = \frac{D_{PSC}K^P_{SC/LQ}}{L_{SC}} = P_{SC} \tag{47}$$

In Eq. (47), P_{SC} is the permeability of the compound through the membrane and this quantity is frequently encountered in the skin and membrane permeation literature. Similar to k_{evap}, it has the units of velocity (length/time). For the purposes of this section, we will assume that the compound has appreciable volatility such that $V_s = k_{evap}$. The results of this analysis can easily be

converted to that for a relatively nonvolatile compound either by considering k_{evap} to be a very small number or by rescaling the problem, as discussed below.

These scale factors then result in the following set of dimensionless describing equations, along with the appropriate dimensionless groups that will permit us to assess when certain approximations can be assumed and justified:

$$\left(\frac{L_{SC}^2}{t_\circ D_{PSC}}\right)\frac{\partial \omega_{PSC}^*}{\partial t^*} - \frac{\omega_{SC}^{\max}}{\left(1-\omega_{SC}^{\max}\omega_{PSC}^*\right)}\left(\frac{\partial \omega_{PSC}^*}{\partial z_{SC}^*}\right)^2 = \frac{\partial^2 \omega_{PSC}^*}{\partial z_{SC}^{*\,2}} \tag{48}$$

$$\left(\frac{L_{VT}^2}{t_\circ D_{PVT}}\right)\frac{\partial \omega_{PVT}^*}{\partial t^*} - \frac{\omega_{VT}^{\max}}{\left(1-\omega_{VT}^{\max}\omega_{PVT}^*\right)}\left(\frac{\partial \omega_{PVT}^*}{\partial z_{VT}^*}\right)^2 = \frac{\partial^2 \omega_{PVT}^*}{\partial z_{VT}^{*\,2}} \tag{49}$$

The initial and boundary conditions are written as

At $t^* = 0$, $\omega_{PSC}^* = \omega_{PVT}^* = 0$ (36 & 37)

At $z^* = 1$, for $0 \le t^* \le \dfrac{t_{depl}}{t_o}$, we have $\omega_{PSC}^*\big|_{1,t^*} = 1$ (50)

and for $t^* \ge \dfrac{t_{depl}}{t_s}$, we have $\dfrac{1}{\left(1-\omega_P^{SC}\right)^*}\dfrac{\partial \omega_{PSC}^*}{\partial z_{SC}^*} = \left(\Big/ \dfrac{k_{evap}}{\dfrac{D_{PSC}K_{SC/LQ}^P}{L_{SC}}}\right)\omega_{PSC}^*\Big|_{\frac{L_{sc}}{z_{SCs}},t^*}$ or

$$\frac{1}{\left(1-\omega_P^{SC}\right)^*}\frac{\partial \omega_{PSC}^*}{\partial z_{SC}^*} = \left(\frac{k_{evap}}{P_{SC}}\right)\omega_{PSC}^*\Big|_{\frac{L_{sc}}{z_{SCs}},t^*} \tag{51}$$

where $P_{SC} = \dfrac{D_{PSC}K_{SC/LQ}^P}{L_{SC}}$ represents permeability of the penetrant in the SC.

At $z^* = 0$, $\omega_{PSC}^*\big|_{0,t^*} = \omega_{PVT}^*\big|_{0,t^*}$ (52)

And $\dfrac{1}{\left(1-\omega_P^{VT}\right)^*}\dfrac{\partial \omega_{PVT}^*}{\partial z_{VT}^*} = \left(\dfrac{P_{SC}}{P_{VT}}\right)\dfrac{1}{\left(1-\omega_P^{SC}\right)^*}\dfrac{\partial \omega_{PSC}^*}{\partial z_{SC}^*}$ (53)

where $P_{VT} = \dfrac{D_{PVE} K^{P}_{VT/SC} K^{P}_{SC/LQ}}{L_{VT}}$ is similarly the permeability of penetrant in the VT phase.

At $z^* = 1$, $\omega^*_{PSC}\big|_{1,t^*} = \omega^*_{PVE}\big|_{1,t^*} = 0$ (54 & 55)

Finally, the coupled ODE is written as:

$$\left(\frac{dL}{dt}\right)^* = -1 + \left(\frac{P_{SC}}{k_{evap}}\right)\left[\frac{1}{\left(1-\omega_P^{SC}\right)^*}\frac{\partial \omega^*_{PSC}}{\partial z^*_{SC}}\right]\Bigg|_{z^*_{SC}=1} \tag{56}$$

with the corresponding dimensionless initial condition:

at $t^* = 0$, $L^* = 1$ (57)

3.2.4 Significance of Dimensionless Groups and Conditions for Approximations

Scaling analysis of the describing equations clearly indicates that the system dynamics is a combined interplay of specific dimensionless groups, which can be estimated through measurable physical quantities. Most importantly, these physical quantities are known *a priori*, which means that knowledge of the dimensionless groups is available even before any experimental or theoretical work is carried out. The coefficient of the unsteady-state (1st) terms in Eqs. (48) and (49) is the reciprocal of the respective Fourier numbers and is given as

$$Fo_{SC} = \frac{t_\circ D_{PSC}}{L^2_{SC}} \text{ and } Fo_{VT} = \frac{t_\circ D_{PVT}}{L^2_{VT}} \tag{58 \& 59}$$

Eqs. (48) and (49) indicate that if the Fourier numbers are inherently large for example for thick barrier layers, the system will quickly reach steady-state. Also at long times, the system will automatically reach steady-state. If the Fourier numbers are greater than 100, the unsteady-state term can be dropped from the analysis and a (pseudo) steady-state analysis is sufficient to accurately describe the process. In such cases, the PDEs can be replaced by ODEs. Even if these

ODEs are non-linear (due to the presence of the convection term) they will be much simpler to integrate. Eqs. (48) and (49) also indicate that the magnitude of the convection terms depend on $\omega_{SC}^{\max}$ and $\omega_{VT}^{\max}$. If $\omega_{SC}^{\max}$ and $\omega_{VT}^{\max}$ are both less than 0.01 then convective mass flux can be safely neglected. This means that if the maximum solubility (saturation concentration) of the permeant in the skin layers is small, then convection is negligible compared to diffusion. This is commonly referred to as the 'dilute solution approximation' in the chemical engineering literature. However, if $\omega_{SC}^{\max}$ and $\omega_{VT}^{\max}$ are large, then convection must be incorporated in the describing equations.

It is obvious that for such a combined evaporation-absorption process, the system response (particularly, the individual fractions of the applied liquid that evaporate and absorb) will depend on the evaporative mass-transfer coefficient and the permeabilities of SC and VT. The scaling analysis above reveals that this dependence is not random. In fact, the system behavior is not dictated merely by the absolute values of system parameters, but their values relative to each other, arranged specifically in terms of dimensionless groups. Eqs. (48-57) show that for a constant P_{VT}, the ratio of evaporated and absorbed fractions of the applied amount of liquid (henceforth referred to as disposition ratio) should depend only on the ratio of k_{evap} and P_{SC}. This ratio appears frequently in Eqs. (48-57) and in accordance with the definition given above, may be called the disposition Biot number (*Bi*), representing the relative tendency of the compound to evaporate into the surrounding gaseous phase as compared to its tendency to absorb and penetrate through the SC. Thus, we have

$$Bi = \frac{k_{evap}}{P_{SC}} \tag{60}$$

It should be noted that a dimensionless group (χ) conceptually similar to this Biot number and representing the relative evaporation-absorption tendency, had been introduced in the skin permeation model proposed by Kasting and Miller (2006).

In the current system, the SC and VT offer resistance to permeation in series and the equivalent permeability (P_{EQ}) of the system is given as:

$$\frac{1}{P_{EQ}} = \frac{1}{P_{SC}} + \frac{1}{P_{VT}} \tag{61}$$

Eq. (61) can be rewritten as:

$$\frac{P_{SC}}{P_{EQ}} = 1 + \frac{1}{P_R} \tag{62}$$

where the relative permeability (P_R) is given as:

$$P_R = \frac{P_{VT}}{P_{SC}} \tag{63}$$

Also, this dimensionless group P_R is similar to the parameter B introduced by Cleek and Bunge representing the relative permeability of SC and Epidermis (1993). It is clear that for substantially large values of P_R, P_{EQ} will be nearly equal to P_{SC}. ○(1) scaling analysis predicts that for P_R values greater than 100, the presence of the VT will be virtually undetectable. In that case, a monolayer skin model (with only SC) is sufficient to describe the permeation characteristics for that particular permeant. On the other hand, if P_R is less than 100, it is important and should be included in the analysis. Also, Eq. (56) shows that the interfacial displacement velocity (slope of the curve depicting changing LQ layer thickness with time) will depend on *Bi*. For *Bi* values greater than or equal to 100, the second term in the right-hand side of Eq. (56) will be negligibly small. In this case, evaporation dominates the absorption process

and the change of LQ thickness due to permeation of penetrant into the SC is negligible compared to evaporation at the free surface. In such cases, the normalized slope is equal to -1 at all times. However, for *Bi* values less than 100, this term cannot be neglected. In such cases, the slope is significantly higher than -1 and its magnitude increases progressively with decreasing *Bi*. If cases where k_{evap} is very small Eq. (56) is not valid. In this case the second term in Eq. (42) is much greater than 1 and violates the underlying principles of ○(1) scaling analysis. In such cases, absorption of the compound into SC is the dominant mechanism responsible for LQ film thinning and Eq. (42) can be rewritten as:

$$\left(\frac{dL}{dt}\right)^* = -Bi + \left[\frac{1}{\left(1-\omega_P^{SC}\right)^*}\frac{\partial \omega_{PSC}^*}{\partial z_{SC}^*}\right]\Bigg|_{z_{SC}^*=1} \tag{64}$$

similar to Eq. (42), other equations involving *Bi* should be changed accordingly for this situation.

In contrast to the factors discussed above that affect the system behavior, there are factors that have no effect, even though intuitively they seem capable of exerting an influence. Eqs. (48-57) suggest that the disposition ratio should be independent of the density of the penetrating substance as well as the density of the barrier layers. These equations also suggest that the evaporated and absorbed fractions will be independent of the initial dose (amount of liquid) applied onto skin.

3.3 Solution Methodology

Similar to the representative simulations in Chapter 2 of this dissertation, Eqs. (5-17) were solved numerically using a finite difference/finite element based subroutine D03PPF from the NAG® (Numerical Algorithms Group) mathematical library (NAG, 2007). The solution is obtained using the method of lines. The PDEs are discretized in space, and the resulting ODEs

are integrated in time using the Gear method which is suited for problems in which the gradients can become very large. Trial simulations were performed with increasing mesh sizes. A mesh size of 201 nodes was determined to have sufficient numerical accuracy and was used to generate the simulation results reported in this section. Also, similar to that in Chapter 2, the mode equations have been converted from multilayer to monolayer through a coordinate transformation that has been discussed in Appendix A. The transformed equations are given in Table 3.2.

3.4 Validation of Scaling Analysis through Representative Simulations

The predictions and conclusions made by scaling above can be validated by numerically solving the comprehensive model equations (without any non-dimensionalization or approximation). If there is appreciable correlation or similarity, then it can be safely concluded that (a) the numerical algorithm and the computer program and accurate and (b) the $\circ(1)$ scaling analysis is correct and the approximations suggested by the analysis are justified. Several representative simulations have been considered in this section, where the system parameters are systematically varied, in order to test the effect of these parameters on the behavior of the system. The latter has been measured through the disposition ratio and the profile of cumulative evaporation and absorption fluxes, as they vary with time. Table 3.3 lists the default value of the system parameters considered for all the simulations in this section.

Table 3.2. Model Equations for Transformed Coordinate System

$$L_{SC}^2 \frac{\partial \omega_P^{SC}}{\partial t} - \frac{D_{PSC}}{\left(1-\omega_P^{SC}\right)}\left(\frac{\partial \omega_P^{SC}}{\partial z^{SC}}\right)^2 = D_{PSC}\frac{\partial^2 \omega_P^{SC}}{\partial z^{SC^2}} \tag{53}$$

$$L_{VT}^2 \frac{\partial \omega_P^{VT}}{\partial t} - \frac{D_{PVT}}{\left(1-\omega_P^{VT}\right)}\left(\frac{\partial \omega_P^{VT}}{\partial z^{VT}}\right)^2 = D_{PVT}\frac{\partial^2 \omega_P^{VT}}{\partial z^{VT^2}} \tag{54}$$

$$\omega_P^{SC}\Big|_{z,0} = \omega_P^{VT}\Big|_{z,0} = 0 \tag{7}$$

$$\omega_P^{VT}\Big|_{-L_{VT},t} = 0 \tag{8}$$

$$\omega_P^{SC}\Big|_{L_{SC},t} = K_{SC/LQ}^P \frac{\rho_P^\circ}{\rho^{SC}} \text{ for } 0 \le t \le t_{depl} \tag{9}$$

$$\frac{\rho^{SC} D_{PSC}}{L_{SC}\left(1-\omega_P^{SC}\right)}\frac{\partial \omega_P^{SC}}{\partial z^{SC}}\bigg|_{1,t} = k_{evap}\rho_P^0\left(\omega_P^{SC}\Big|_{1,t}\right) \text{ for } t > t_{depl} \tag{10}$$

$$\omega_P^{SC}\Big|_{0,t} = \frac{\rho^{VT}}{K_{VT/SC}^P \rho^{SC}}\omega_P^{VT}\Big|_{0,t} \tag{11}$$

$$\frac{\rho^{VE} D_{PVT}}{L_{VT}\left(1-\omega_P^{VT}\right)}\frac{\partial \omega_P^{VT}}{\partial z^{VT}}\bigg|_{0,t} = \frac{\rho^{SC} D_{PSC}}{L_{SC}\left(1-\omega_P^{SC}\right)}\frac{\partial \omega_P^{SC}}{\partial z^{SC}}\bigg|_{0,t} \tag{55}$$

$$\rho_P^\circ \frac{dL}{dt} = -k_{evap}\rho_P^\circ + \frac{\rho^{SC} D_{PSC}}{L_{SC}\left(1-\omega_P^{SC}\right)}\frac{\partial \omega_P^{SC}}{\partial z^{SC}}\bigg|_{z^{SC}=1} \tag{56}$$

$$t^* = 0,\ L^* = 1 \tag{57}$$

Table 3.3. Parameter Values kept Constant for Permeability Dependent Simulations

Parameter (symbol)	**Value (unit)**
Uniform System Temperature	303.15 (K)
Initial VH Thickness (L_0)	0.1000
Density of Penetrant (ρ_p^0)	0.7813
Density of the SC Phase (ρ^{SC})	1.250
Density of the VT Phase (ρ^{VE})	1.000
Thickness of the VT Phase (L_{VE})	0.0100
Diffusivity of A in VT	1.093*e*-3
SC/VH Partition Coefficient of A	0.3410
VT/SC Partition Coefficient of A	0.2683
Relative Tolerance (RTOL)	1.000*e*-8
Absolute Tolerance (ATOL)	1.000*e*-8
Time Increment Step Size	1.000*e*-3

3.4.1 Permeability Dependent Studies - Interdependence of k_{evap}, P_{SC} and P_{VT}

As discussed earlier, Eqs. (48-57) suggest that for a constant P_{VT} the disposition ratio should only depend on *Bi*. Three different systems of varying values k_{evap} and P_{SC} are considered. In particular, for systems 1 and 2, the individual values of k_{evap} and P_{SC} are different, but their ratio (*Bi*) is equal to 1. For systems 1 and 3, the values of k_{evap}, P_{SC} and *Bi* are the same, but the parameters that make up P_{SC} such as diffusivity, partition coefficient and layer thickness have been varied randomly. The exact values of the varied parameters are given in Table 3.4. The comparative simulation results have been given in Fig. 3.1. This clearly shows that in spite of the difference in the absolute values for systems 1 and 2, the disposition ratio is identical since their *Bi* values are same. The shape of the cumulative flux profiles in Fig 3.1 suggests that systems 1 and 2 have different evaporation and absorption kinetics, since the values of k_{evap} and P_{SC} are different. For systems 1 and 3, even though the individual values of diffusivity, layer thickness etc. are different, since the values of k_{evap}, P_{SC} and *Bi* are same, the kinetics for these systems are identical. In this case, not only are the evaporated and absorbed fractions equal, but the shape of the respective profiles are superimposable.

Table 3.4. Parameter Values Varied during Permeability Dependent Study – Constant *Bi*

	System 1	System 2	System 3
P_{VT}	1.000	1.000	1.000
D_{PSC}	2.933*e*-3	5.865*e*-3	5.8652*e*-3
$K^{P}_{SC/LQ}$	0.3410	0.3410	0.3410
L_{SC}	1.000*e*-3	2.000*e*-3	2.000*e*-3
P_{SC}	1.000	2.000	1.000
k_{evap}	1.000	2.000	1.000
Bi	**1.000**	**1.000**	**1.000**

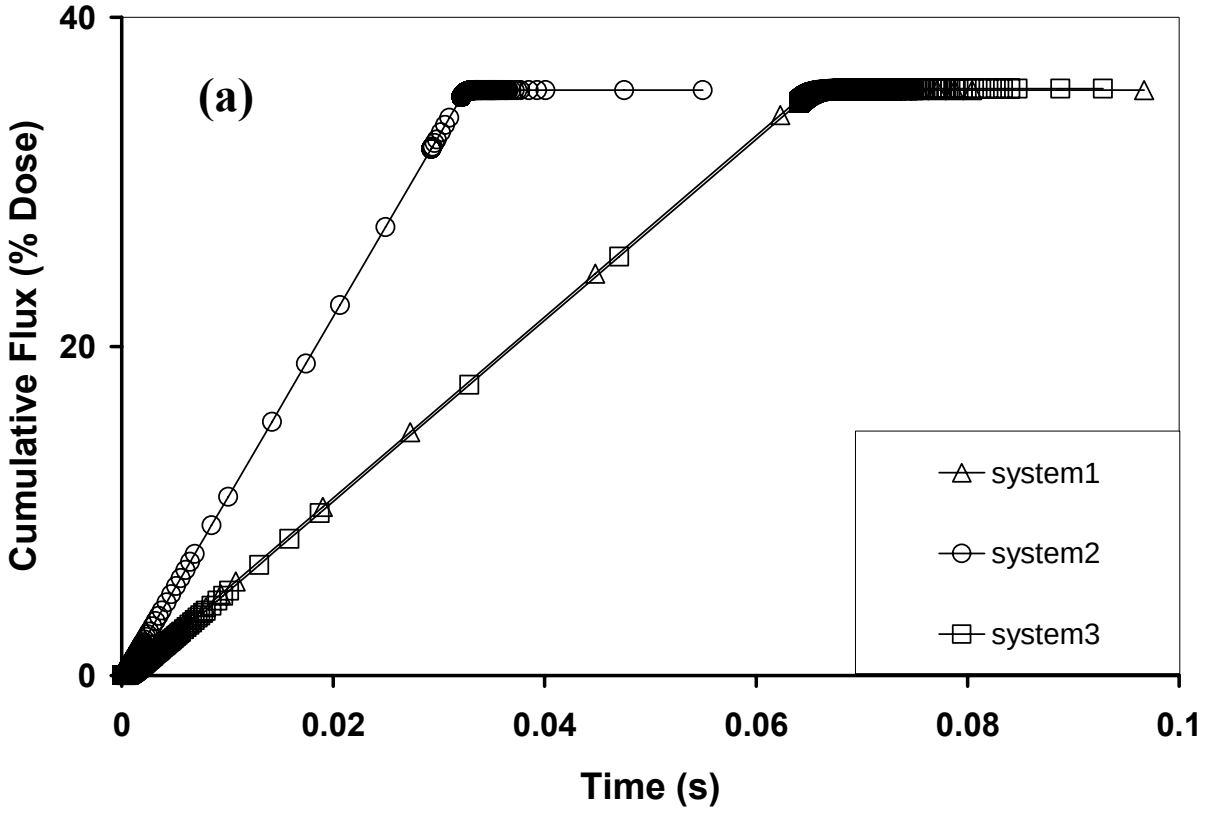


Fig. 3.1(a). Absorption Profiles for Constant *Bi*

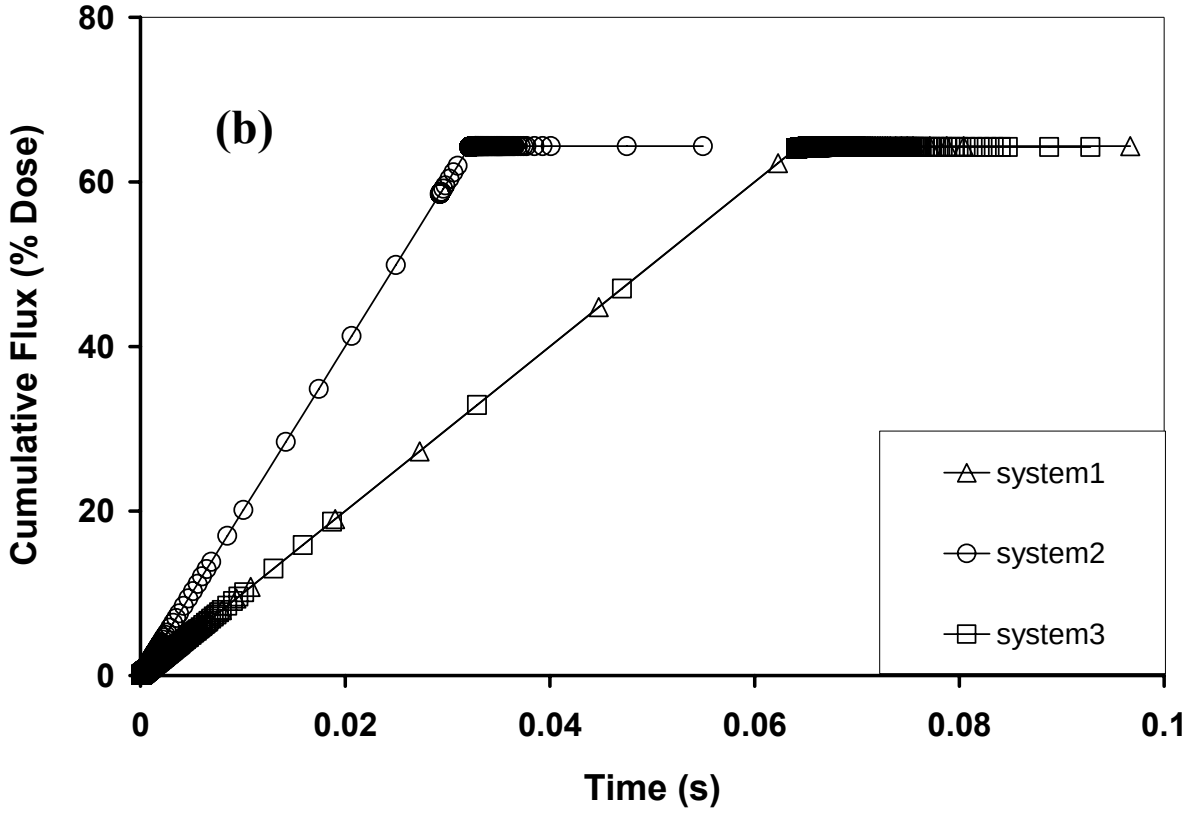


Fig. 3.1(b). Evaporation Profiles for Constant *Bi*

In an attempt to compare the effect of varying relative permeability of the SC and VT, P_{VT} is varied from a very low value (0.001 m/s) to a very high value (1000 m/s), while keeping k_{evap} and P_{SC} constant at 1 m/s. The exact values of the varied parameters are given in Table 3.5. Fig 3.2 plots the fraction of applied dose evaporated and absorbed with varying P_R. When $P_R = 1$, then the fractions evaporated and absorbed are very close to the theoretical predictions of 0.67 and 0.33, respectively. For P_R values lower than 1, the VT is rate-limiting and with a progressive decrease in value of P_R, the fraction absorbed gradually approaches zero. Scaling also suggests that when *Bi* = 1, for very high values of P_R, the evaporated and absorbed fractions of the dose will be equal (50% each). This has been shown in Fig. 3, although the final values are slightly different than the predicted value (50%). This is due to the minor approximations incorporated in the model equations and will be discussed below.

Fig. 3.3 depicts the variation of (normalized) interfacial displacement velocity with time for varying *Bi*. Since the scale factor for this quantity, k_{evap}, in this case is equal to 1, the magnitude of the normalized and the true interfacial displacement velocity are the same. As expected, for *Bi* greater than 100 the velocity (or slope) is nearly equal to -1. However, for *Bi* less than 100 the slope is significantly higher than -1 and the initial and the steady-state slope increases progressively with increasing *Bi*. Table 3.6 lists the exact values of the parameters that were varied in these simulations.

Table 3.5. Parameter values varied during permeability dependent study - variation of P_{VT} or P_R

System Parameter	**Numerical Value used during Simulation**
Thickness of SC Phase	1.000*e*-3
Diffusivity of A in SC	2.933*e*-3
Evaporative Mass Transfer Coefficient (k_{evap})	1.000
Permeability in SC (P_{SC})	1.000
Permeability in VT (P_{VT})	1.000*e*-3 – 1.000*e*3
Equivalent Permeability P_{EQ}	9.9900*e*-4 – 0.9990
Bi	1.000
Ratio of k_{evap} and P_{VT}	1.000*e*-3 – 1.000*e*3
Ratio of k_{evap} and P_{EQ}	1.001 – 1.001*e*3

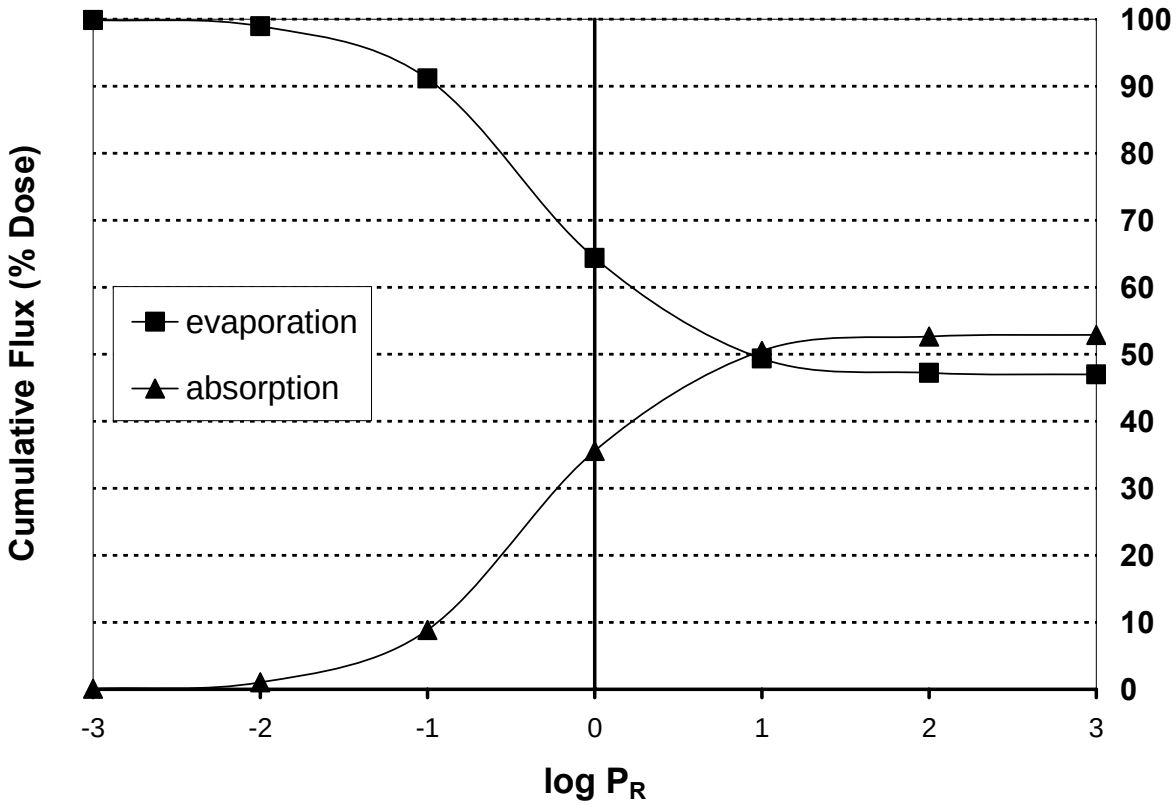


Fig. 3.2. Fractions of Applied Dose Evaporated and Absorbed with Varying P_R

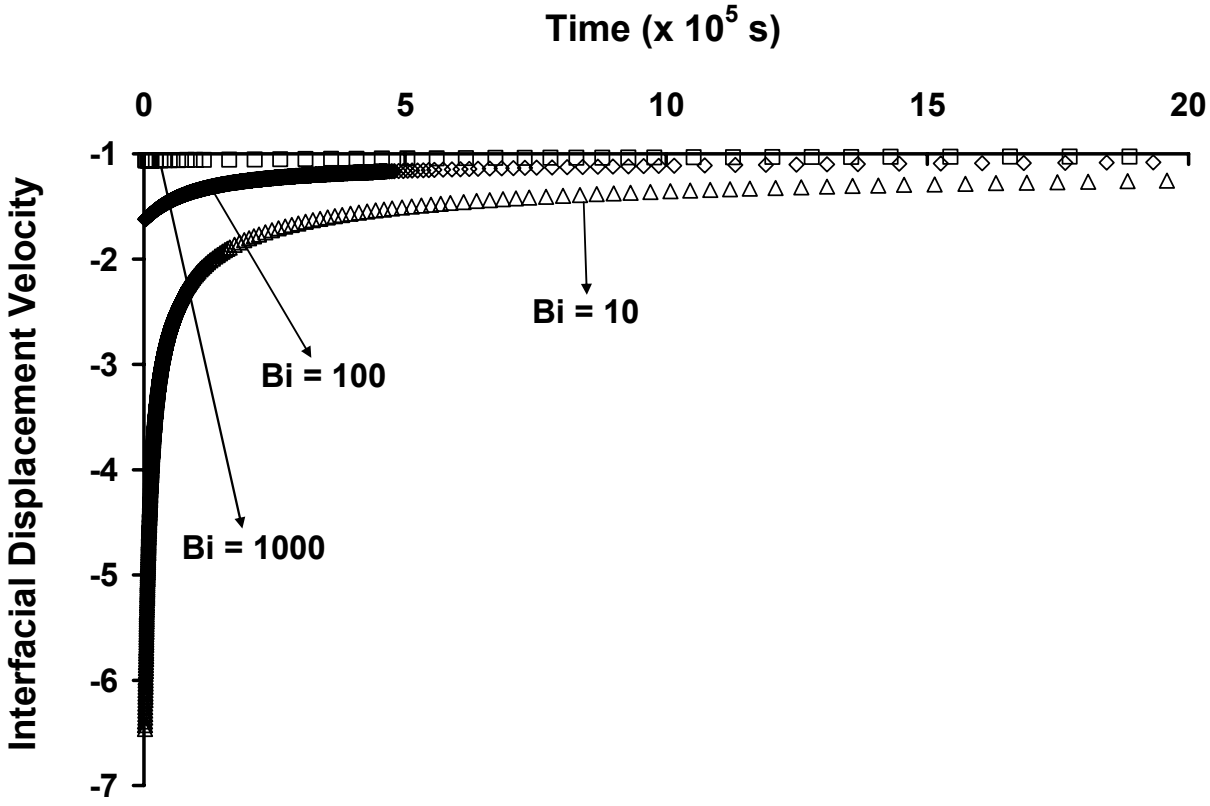


Fig. 3.3. Variation of Interfacial Displacement Velocity with Varying *Bi*

3.4.2 Dose and Density Dependent Studies

As mentioned above, Eqs (48-57) suggest that the disposition ratio should be independent of the density of the penetrating substance as well as that of the barrier layers. Several representative simulations are performed by separately varying these density values. Table 7 lists the parameters values as well as the simulation results depicting the effect of varying pure component density of the penetrant, while the SC and VT density are kept constant. In this case, SC has been converted to a rate-limiting resistance by choosing P_{VT} = 100 m/s, while k_{evap} and P_{SC} remain unchanged at 1 m/s. In such cases, according to scaling, both the evaporated and absorbed fractions of the dose should be equal to 0.5. This is clearly shown in Table 3.6, although the final values are slightly different than the predicted values. A closer look at the numerical values in Table 3.6 indicate that this deviation or offset increases with decreasing pure component density. On the other hand, Table 3.8 shows the simulation results when the density of the SC is varied, while the density of the LQ and VT are kept constant. Similar to the earlier situation (with variable LQ density), the fractions evaporated and absorbed are very close to the expected value of 0.5, but not exactly same and the offset increases with decreasing SC density.

Surprisingly, this error or offset observed for density dependent studies in Tables 3.7 and 3.8 has also been observed in Figs. 3.1 and 3.2. It is a direct manifestation of two main approximations that have been incorporated in the proposed model. Firstly, when a finite amount of penetrant

Table 3.6. Parameter values varied during permeability dependent study - variation of interfacial displacement velocity with varying *Bi*

	System 1	System 2	System 3
P_{VE}	1.000		
D_{PSC}	2.933*e*-6	2.933*e*-5	2.933*e*-4
$K_{SC/VH}^{P}$	0.3410		
L_{SC}	1.000*e*-3	2.000*e*-3	2.000*e*-3
P_{SC}	1.000*e*-3	1.000*e*-2	1.000*e*-1
k_{evap}	1.000		
Bi	**1000**	**100**	**10**

Table 3.7. Parameter values and results for density dependent simulations – variation of penetrant density

	System 1	System 2	System 3
P_{VE}	0.0100		
P_{SC}	1.000		
k_{evap}	1.000		
SC Density (ρ^{SC})	1.250		
VT Density (ρ^{VE})	1.000		
Penetrant Density (ρ_P^0)	0.3907	0.7813	1.5626
Final Evaporated Fraction (% of Applied Dose)	**50.02**	**49.85**	**49.50**
% Offset (Evaporated Fraction)	**0.0470**	**0.2955**	**0.9927**
Final Absorbed Fraction (% of Applied Dose)	**49.97**	**50.13**	**50.47**
% Offset (Absorbed Fraction)	**0.0611**	**0.2674**	**0.9356**

Table 3.8. Parameter values and results for density dependent simulations – variation of SC density

	System 1	System 2	System 3
P_{VE}	0.0100		
P_{SC}	1.000		
k_{evap}	1.000		
Penetrant Density (ρ_P^0)	1.000		
VT Density (ρ^{VE})	1.000		
SC Density (ρ^{SC})	1.000	5.000	10.00
Final Evaporated Fraction (% of Applied Dose)	**45.10**	**49.30**	**49.76**
% Offset (Evaporated Fraction)	**9.793**	**1.390**	**0.4890**
Final Absorbed Fraction (% of Applied Dose)	**54.67**	**50.66**	**50.23**
% Offset (Absorbed Fraction)	**9.333**	**1.318**	**0.4529**

enters the SC (and eventually the VT), it should initially increase its thicknesses. Since, the amount of liquid absorbed into the membranes is small, this change in thickness has been neglected, thereby incorporating a small error in the mass conservation of the system and in the describing equations. This assumption can be removed by converting the barrier thicknesses to be a function of time, depending on the magnitude of interfacial absorption fluxes. This would increase the complexity of the describing model by incorporating additional coupled ODEs. Secondly, even though a finite amount of penetrant enters the SC and VT, the density for these skin layers is assumed to be constant and equal to their respective intrinsic densities. The latter approximation is valid only when the density of the SC and/or VT is much greater than that of the permeating species. Thus, these approximations have led to offsets between the simulated values and the values predicted by scaling analysis. It is not surprising that these offsets decrease with increasing relative density of the membranes with respect to the density of the permeant.

Eqs (48-57) also suggest that the disposition ratio will be independent of the amount of applied dose (the initial thickness of the LQ layer). Three values of applied dose were chosen: 100, 1000 and 10000 μL/cm^2. The relevant permeability values and k_{evap} were kept constant and equal to 1. The simulation results are given in Fig. 3.4, which clearly shows that the disposition ratio is independent of the dose and the final evaporated and absorbed fractions for all cases are identical. Moreover, as expected, even though the final values are identical for any dose, the profiles for cumulative evaporation and absorption vary due to variation in the amount of applied dose.

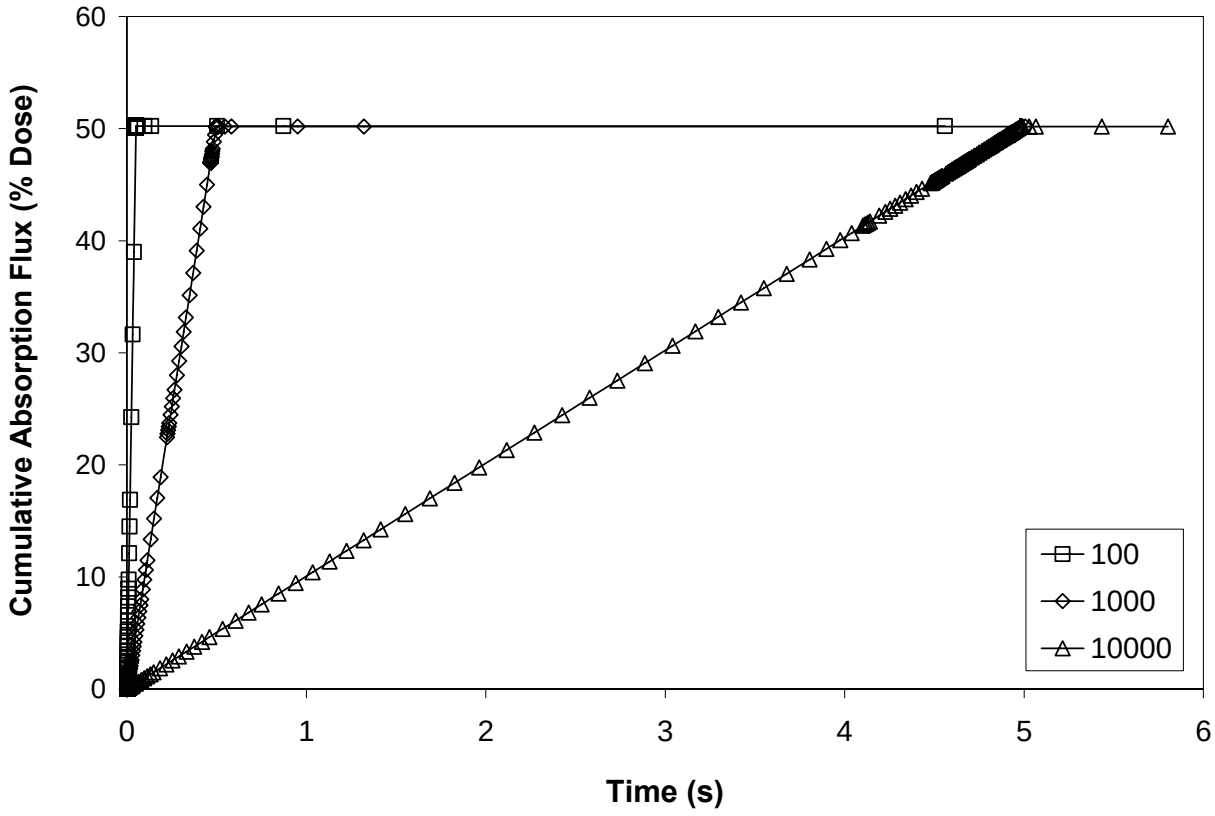


Fig. 3.4. Variation of Fraction Absorption with Amount of Applied Dose

3.5 *Conclusion*

The describing differential equations for transient solution-diffusion models for barrier permeation can be sufficiently complex and an appropriate approximation analysis is strongly warranted, even before an analytical or numerical solution is attempted. This section applies ○(1) scaling analysis (Krantz, 1970) to a transient mass transfer through a multi-lamellar membrane. An in-depth scaling analysis reveals valuable information about the system dynamics and its dependence on different system parameters. It also identifies the factors that exert no influence on the system characteristics. Scaling analysis also reveals conditions under which certain assumptions can be made in order to appreciably simplify the mathematical equations describing the process. The scaling analysis done in this section can be applied to any arbitrary system involving permeation through multi-lamellar membrane having a rectangular geometry. The results can also be used simply for transient permeation through a single membrane. The methodology can also be successfully extended to transient permeation through multi-lamellar membranes in cylindrical or spherical coordinate systems.

References

Basketter, D.A., Gerberick, G.F., Robinson, M.K., (1996). Risk assessment in toxicology of contact hypersensitivity, in I. Kimber and T. Maurer (Eds.), Taylor & Francis, London, pp.152-164.

Bear, J, Bachmat, Y., (1990). Introduction to Modeling of Transport Phenomena in Porous media, Kluwer Academic Publishers, Boston.

Bird, B.R., Stewar,t W.E., Lightfoot, E.N., (1960). Transport Phenomena, John Wiley and Sons Inc., New York.

Bunge, A., Flynn, G.L., Guy, R.H., Predictive model for dermal exposure assessment, in R.G.M. Wang (Ed.), Water Contamination and Health: Integration of Exposure Assessment, Toxicology, and Risk Assessment, Marcel Dekker, New York.

Carslaw, H.S., Jaeger, J.C., (1986). Conduction of Heat in Solids, Oxford University Press, New York.

Cleek, R.L., Bunge, A.L., (1993). A new method for estimating dermal absorption from chemical exposure. 1. General approach. Pharmaceutical Research, 10, 497–506.

Crank, J., (1975). Mathematics of Diffusion, 2nd ed., Oxford University Press, New York.

Dijkstra, M.F.J., Bach, S., Ebert, K., (2006). A transport model for organophilic nanofiltration, Journal of Membrane Science, 286 (2006) 60-68.

Gerberick, G.F., Robinson, M.K., (2000). A skin sensitization risk assessment approach for evaluation of new ingredients and products, American Journal of Contact Dermatitis, 11, 65-73.

Hellums, J.D., Churchill, S.W., (1964). Simplification of the mathematical description of boundary and initial value problems, AIChE Journal, 10, 110-114.

Kasting, G.B., Miller, M.A., (2006). Kinetics of finite dose absorption through skin 2. volatile compounds, Journal of Pharmaceutical Sciences, 95, 268–280.

Krantz, W.B., (1970). Scaling Initial and Boundary Value Problems as a Teaching Tool for a Course in Transport Phenomena, Chemical Engineering Education, 4, 145-152.

Krantz, W.B., (2007). Scaling Analysis in Modeling Transport and Reaction Processes: A Systematic Approach to Model Building and the Art of Approximation, Wiley-Interscience, New York.

Menon, G.K., (2002). New insights into skin structure: scratching the surface, Advance Drug Delivery Reviews, 54 (Suppl. 1), S3–S17.

Mikhailov, M.D., Özişik, M.N., (1984). Unified Analysis and Solutions of Heat and Mass Diffusion, Wiley Interscience, New York.

Montagna, W., Kligman, A.M., Carlisle, K.S., (1992). Atlas of Normal Human Skin. Springer–Verlag, Berlin and Heidelberg.

Robinson, M.K., Gerberick, G.F., Ryan, C.A., McNamee, P., White, I.R., Basketter, D.A., (2000). The importance of exposure estimation in the assessment of skin sensitization risk. Contact Dermatitis, 42, 251–259.

Robinson, P.J., (1998). Prediction: Simple risk models and overview of dermal risk assessment, in M.S. Roberts and K.A. Walters (Eds.), Dermal Absorption and Toxicity Assessment, Marcel Dekker, New York, pp.203-244.

Scheuplein, R.J., Blank, I.H., (1971). Permeability of the skin, Physiologiical Reviews, 51, 702.

Singh, S., Singh, J., (1993). Transdermal drug delivery by passive diffusion and iontophoresis: a review, Med. Res. Rev., 13, 569–621.

Tosun, I.., (2002). Modeling in Transport Phenomena: A Conceptual Approach, Elsevier, Amsterdam, pg. 302.

Welty, J.R., Wicks, C.E., Wilson, R.E., Rorrer, G.L., (2001). Fundamentals of Momentum, Heat and Mass Transfer, Wiley, New York.

Wijmans, J.G., Baker, R.W., (1995). The solution-diffusion model: a review, Journal of Membrane Science, 107, 1-21.

CHAPTER 4

PERCUTANEOUS PERMEATION OF ETHANOL

4.1 Background

Ethanol is a volatile liquid that is commonly used in the chemical industry as well as cosmetic and pharmaceutical formulations. The skin's permeability to ethanol and – conversely – the impact of ethanol on skin permeability have been widely studied, for a number of reasons. Ethanol and other *n*-alkanols have frequently served as test compounds for determination of structure-permeability relationships in the stratum corneum (SC) (Scheuplein, 1965; Scheuplein, 1967; Scheuplein and Blank, 1973; Durrheim et al., 1980; Ackermann et al., 1987; Cross et al., 2003) and other layers of the skin (Scheuplein and Blank, 1973). Ethanol has been widely used to increase the skin's permeability to drugs delivered transdermally (Williams and Barry, 2004), in part because of its excellent skin tolerance. In this respect its synergistic behavior with water as an enhancer system for transdermal nitroglycerin has been particularly well studied (Berner et al., 1989a, 1989b). In rats even an oral dose of ethanol can increase skin permeability to toxic chemicals (Brand, 2004; Brand et al., 2006). Its "inside-out" kinetics has recently been modeled (Anderson and Hlastala, 2006) in an attempt to understand the operation of new devices that measure the concentration of blood alcohol transdermally. Unlike more hazardous solvents, ethanol dermal absorption from the vapor phase does not appear to have been studied, and reports of its absorption from unoccluded systems are isolated (Williams et al., 1994).

The objective of this chapter is to characterize the skin disposition (evaporation from and absorption into the human skin) under unoccluded conditions of a highly volatile solvent in terms of a mass transport model described in Chapter 3 of this dissertation. Ethanol was chosen as a model permeant due to its known skin permeation characteristics and its noncorrosive nature – it does not permanently alter skin permeability or extract lipids from the SC (Williams and Barry, 2004). Furthermore, its kinetics as a pure compound are of interest in interpreting the skin

disposition of other compounds deposited on skin from ethanolic formulations (solutions) (Santhanam *et al*, 2005; Miller *et al*, 2006). Such solvent-deposition techniques for delivering compounds topically are widely employed in the drug and cosmetic industries; for example, fine fragrances are usually formulated in ethanol or phenoxyethanol (Vuilleumier et al., 1995). Although, ethanol has been chosen to be the representative permeant, the diffusion model can be applied to any pure compound in its liquid state, as can be seen in subsequent chapters. In this study, the permeation of pure ethanol through human skin in vitro has been experimentally measured and the absorption data have been analyzed using the mathematical model described in Chapter 2 of this dissertation.

4.2 Materials and Methods

4.2.1 Chemicals

[^{14}C]-Ethanol at specific activity 2.8 mCi/mmol was purchased from Sigma-Aldrich (St. Louis, MO). The radiochemical purity was stated by the supplier to be 100%. Unlabeled neat ethanol was purchased from Aaper (Shelbyville, KY) and calcium-free Dulbecco's phosphate-buffered saline was purchased from Sigma-Aldrich (St. Louis, MO). Ultima Gold™ scintillation fluid and Soluene-350™ were purchased from Perkin-Elmer Biosciences. All other chemicals used were reagent grade.

4.2.2 Permeation Methodology

Tissue preparation and handling have been previously described (Kasting and Bowman, 1990). Split-thickness human cadaver skin either from back, abdomen, or thigh, obtained from the U.S. Tissue and Cell and stored in the foil packs at −80°C, was thawed in 35-40°C distilled

water and then rinsed with Dulbecco's phosphate-buffered saline, pH 7.4, preserved with 0.02% sodium azide (PBS) solution to remove the glycerol used in the preservation process. The skin was cut into ~1.5 cm × 1.5 cm squares using a 22 blade scalpel and mounted onto modified Franz diffusion cells (0.79 cm^2) (Merritt and Cooper, 1984) with the SC facing the donor compartment and the dermis contacting the receptor compartment. Non-occluded glass tops, which were open to the atmosphere and extended 4 mm above the skin surface, were placed on top of the skin and firmly clamped in place. The receptor compartments were filled with PBS, taking care to remove any air bubbles trapped between the skin and receptor solution. The cells were placed into aluminum blocks maintained at 37°C in thermostatted heating/stirring modules. The temperature of the skin surface was approximately 32°C. In order to mimic outdoor wind conditions (1.5 m/s) (Miller et al., 2006), the modules were placed in a fume hood with the sash height at 18 inches. A tissue integrity test was performed using ^3H_2O and the samples were randomized over treatments based on ^3H_2O permeation results (Franz and Lehman, 1990; Kasting et al., 1994), after which the skin samples were allowed to equilibrate overnight. After a final receptor fluid exchange to remove residual ^3H_2O, aliquots (5-40 μL) of ^{14}C-labeled ethanol was applied to the skin samples, n = 4-5/dose, and allowed to freely evaporate from and penetrate into the skin in the fume hood environment. The receptor solutions were exchanged with fresh PBS at predetermined time points ranging from 5 min to 24 h post-dose, and the collected samples were analyzed by liquid scintillation counting (LSC). After 24 h the skin samples were removed from the diffusion cells, dissolved in 2 mL of Soluene and analyzed by LSC. Skin permeation results were obtained from three donors, with n = 4-5 per donor per dose. All data were first averaged by dose for each donor and then averaged across donors to obtain the reported results.

4.2.3 Evaporation rate measurements

In a separate set of experiments, evaporation rates of ethanol (and benzene) in the Franz cell environment were determined gravimetrically by applying a large (80 μL) dose of non-radiolabeled solvent to skin mounted in the diffusion cells. The cells were thermostatted and maintained in the fume hood environment as they had been in the permeation studies. The cells were periodically removed from the hood, weighed and replaced into their positions. The temperature of the skin surface was monitored by infrared thermometry throughout the course of each study. Evaporation rate and the associated mass transfer coefficient k_{evap} were determined from the slopes of weight versus time plots as described later. An additional experiment was conducted in which the thermostatted Franz cells were placed on a laboratory benchtop rather than inside the fume hood. This experiment yielded a lower evaporation rate than did the fume hood experiments, consistent with expectations for an indoor environment.

4.2.4 Estimation of transport and thermodynamic parameters

The transport model described in Chapter 3 of this dissertation requires as inputs a number of vehicle and tissue dimensions, physicochemical properties of the permeants and their diffusivities and partition coefficients in the two skin layers, designated SC (stratum corneum) and VT (viable tissues). In this representation the viable epidermis and residual dermis in the split-thickness skin samples are treated as a single layer with the properties of unperfused dermis (Kretsos et al., 2007). Support for this approach stems from previous experience in our laboratory (Kretsos et al., 2004) and others (Cleek and Bunge, 1993). It is particularly appropriate in cases where accurate estimates of skin concentrations are not required and the

lower skin layers are not rate-determining for absorption. Such is the case for ethanol, as will be shown.

Relevant physicochemical properties for ethanol transport in skin are readily obtained from the literature and are listed in Table 4.1. These properties were used as input parameters for the previously developed correlations for transport parameters and partition coefficients summarized in Table 4.2. Parameters for which the present analysis departs from the literature values are discussed below.

4.2.4.1 Layer thicknesses

The initial thickness of the vehicle layer deposited on the skin surface was calculated from the volume of ethanol applied, its density at 32°C and the diffusion cell cross-sectional area. A correction was made for the amount of ethanol assumed to be rapidly deposited into the SC deposition layer h_{dep} (Kasting and Miller, 2006). The unswollen SC thickness was taken to be 13.4 μm (Johnson et al., 1997; Wang et al., 2006); this value was adjusted upward in a concentration-dependent manner to account for swelling by ethanol following application of the dose. A zero volume of mixing assumption resulting in a linear density correction was used for this adjustment. The thickness of the VT layer was taken to be 500 μm. This sample thickness for in vitro studies typically results when cadaver skin dermatomed to a nominal thickness of 300 μm swells in a saline buffer (Khalil et al., 2006).

Table 4.1. Physicochemical properties of ethanol for skin disposition estimation

Property	Symbol	Units	Value	Ref.
Molecular weight	MW	g/mol	46.07	Yaws (1999)
Density	ρ	g/cm^3	0.7807	Yaws (1999)
Vapor pressure	P_{vp}	torr	87.73	Yaws (1999)
Log (octanol/water partititon coefficient	$\log K_{oct}$		-0.3100	Yaws (1999)
Water solubility	S_w	g/L	789 (miscible)	Yaws (1999)
Fraction unbound to albumin	f_u		0.700	Yamazaki & Kanaoka (2004)
Fraction nonionized	f_{non}		1	Martin (1993)

Table 4.2. Thermodynamic and transport properties for estimation of ethanol skin disposition

Property	Units	Value	Correlation	Reference
			Vehicle layer (VH)	
u	m/s	1.50	Non-linear regression	Miller et al. (2006)
k_{evap}	cm/h	0.667	$k_g = 6320\,u^{0.78}/MW^{1/3}$ $k_{evap}\rho = k_g P_{vp} MW/\left(7.6\times10^{-5} RT\right)$	Peress (2003), N-Dri-Stempfer & Bunge (2005), Kasting & Miller (2006)
			Stratum corneum (SC)	
L_{SC}	μm	13.40		Johnson et al. (1996), Wang et al. (2006)
ρ_{SC}	g/cm^3	1.194	$\rho_{SC} = (1+v)/\left(\omega_{lip}/\rho_{lip} + \omega_{pro}/\rho_{pro} + v/\rho_w\right)$	Nitsche et al. (2006)
D_{SC}	cm^2/s	9.291×10^{-10}	WKN correlation	Wang et al. (2007)
$K_{SC/VH}$		0.3029	Experimental data	Berner et al. (1989)
			Viable tissue (VT)	
L_{VT}	μm	500.0		Khalil et al. (2006)
ρ_{VT}	g/cm^3	1.019	$\rho_{VT}\left(g/cm^3\right) = \left(\omega_{H_2O} + 0.649\omega_{pro} + 1.227\omega_{lip}\right)^{-1}$	Anderson et al. (2000)
D_{free}	cm^2/s	5.761×10^{-6}	$\log D_{free} = -4.15 - 0.655\log MW$	unpublished result
D_{VT}	cm^2/s	5.710×10^{-6}	$D_{VT} = D_{free}/\left(0.68 + 0.32/f_u + 0.001 f_{non} K_{oct}\right)$	unpublished result
$K_{VT/SC}$		0.4290		Cross et al. (2003)

4.2.4.2 Evaporation mass transfer coefficient (k_{evap})

When an explicit vehicle layer (VH) is present, this coefficient is defined by the relationship (Kasting and Miller, 2006)

$$dM_{evap}/dt = k_{evap}\rho A \tag{65}$$

where M_{evap} is the mass of evaporating solvent, ρ is its density and *A* the surface area for evaporation. The mass of solvent remaining at time *t* following a dose M_0 is:

$$M(t) = M_0 - M_{evap} = M_0 - \int_0^t \left(k_{evap}\rho A\right)dt \,. \tag{66}$$

Since k_{evap}, ρ and *A* remain constant with time, Eq. (2) is simply:

$$M(t) = M_0 - k_{evap}\rho t \tag{67}$$

$$\text{or } \frac{M(t)}{M_0} = 1 - \frac{k_{evap}}{L_0}t \tag{68}$$

where $M_0 = L_0\rho A$ and L_0 is the initial thickness of the vehicle layer. Thus, the value of k_{evap} was estimated from plots of $M(t)/M_0$ versus time as determined in the evaporation rate experiments. Experimental values were compared with *a priori* estimates derived from the correlation discussed in (Kasting and Miller, 2006) (Table 4.2).

4.2.4.3 SC/VH *partition coefficient ($K_{SC/VH}$) and permeant solubility in* SC *(C_{sat})*

The SC/vehicle partition coefficient was estimated from equilibrium sorption of ethanol from ethanol/water mixtures into isolated human SC as measured by (Berner et al., 1989). The details of this calculation are given in Appendix B and the estimated value of $K_{SC/VH}$ is given in

Table 4.2. The saturation concentration of ethanol in SC containing 30% w/w water was then calculated as

$$C_{sat} = K_{SC/VH}\rho \tag{69}$$

where ρ is ethanol density. This procedure differs from the method described in (Kasting and Miller, 2006) in which an SC/water partition coefficient $K_{SC/w}$ is first calculated from a correlation described by (Nitsche et al., 2006) and C_{sat} is then calculated as $K_{SC/VH}S_w$. The latter method is appropriate for compounds having low water solubility S_w, but fails for highly soluble compounds. The method described in Appendix B based on equilibrium ethanol sorption data (Berner et al., 1989) provides a plausible alternative, at the expense of requiring experimental data.

4.2.4.4 Diffusivity in stratum corneum (D_{SC})

The diffusivity of ethanol in SC was estimated from the skin permeation data by a least squares fitting procedure. Its value was compared to the estimate $D_{WKN} = 9.291 \times 10^{-10}$ cm^2/s (Table 4.2) derived from the correlation for partially hydrated skin proposed by (Wang et al., 2007). In one version of the fitting procedure, a concentration-dependent diffusivity of the form proposed by (Miller et al., 2006) was tested, that is,

$$D_{SC} = D_0 + \frac{D_{sat} - D_0}{1 + \boldsymbol{exp}\left[m\left(1 - C / C_{\boldsymbol{inf}}\right)\right]}. \tag{70}$$

In Eq. (6), C is the instantaneous local concentration of permeant within the SC, D_0 is the intrinsic SC diffusivity, D_{sat} is the maximum or saturation diffusivity, $C_{\boldsymbol{inf}}$ represents the concentration corresponding to the inflection point of the sigmoidal curve and m describes its

slope. The values of C_{inf} and m were taken to be $C_{sat}/2$ and 5.0 as in (Miller et al., 2006), whereas the values of D_0 and D_{sat} were determined from the data.

4.2.4.5 Fractional deposition depth in SC *(f_{dep})*

This parameter was defined by (Kasting and Miller, 2006) as the ratio of the thickness of a highly permeable upper portion of the SC to its full thickness. The value f_{dep} = 0.1 was proposed by (Kasting and Miller, 2006) and supported experimentally for benzyl alcohol (Miller et al., 2006) and DEET (Santhanam et al., 2005). Larger values of f_{dep} lead to increased absorption of volatile compounds, especially in the limit of small doses (Kasting and Miller, 2006; Ray Chaudhuri et al., 2007). In the present study it was of interest to determine whether the absorption of a small, highly volatile solvent (ethanol) would be effectively described by the same value of f_{dep} as the larger molecules, benzyl alcohol and DEET.

4.2.5 Regression analysis

The model parameters in Table 4.2 were used without modification to calculate the time course of absorption and evaporation of ethanol from skin according to the described model (Ray Chaudhuri et al., 2007), with three exceptions: k_{evap}, D_{SC} and f_{dep}. These parameters were varied to provide the best fits of the model to the experimental data. They were selected because their values were believed to be the least accurately predicted by the developed correlations (Table 4.2) and each plays a key role in determining the rate and extent of ethanol absorption. The value of k_{evap} was determined from the evaporation rate experiments, after which the values of D_{sc} and f_{dep} were determined from the absorption data.

Specifically, values of k_{evap} for each evaporation trial were determined by linear regression according to Eq. (4). Gas phase mass-transfer coefficients k_g were then calculated for each trial by decomposing k_{evap} according to the relationship (Miller et al., 2006)

$$k_{evap}\rho = k_g \frac{P_{vp} MW}{\left(7.6\times10^{-5}\right) RT} \tag{71}$$

where the parameters have the units shown in Tables 4.1 and 4.2 and the gas constant R is expressed as 0.0821 L·atm·deg^{-1}mol^{-1}. The constant in Eq. (71) differs from that in (Miller et al., 2006) by a factor of 10^6 to reflect the fact that ρ in this report has units of g/cm^3 rather than μg/cm^3. The advantage of extracting k_g from k_{evap} arises because the temperature and, hence, P_{vp} and T, varied slightly between the evaporation trials. The value of k_g may also be compared with published correlations developed for other applications, e.g., the EPA correlation for chemical spills risk assessment (Peress, 2003; N-Dri-Stempfer and Bunge, 2005; Kasting and Miller, 2006)

$$k_g = 6320\, u^{0.78} / MW^{1/3} \tag{72}$$

In Eq. (72), u is wind velocity in m/s and k_g is expressed in cm/h.

Values of D_{SC} and f_{dep} were determined by nonlinear regression by minimizing the normalized sum of squared residuals χ_v^2, defined as (Bevington, 1969)

$$\chi_v^2 = \frac{1}{n-p}\sum_{i=1}^{n} w_i \left[y_i^{obs} - y_i^{fit} \right]^2 \tag{73}$$

Here n is the number of observations, p is the number of adjustable parameters, w_i is the weight, y_i^{obs} and y_i^{fit}, respectively are the observed and calculated values of cumulative absorption flux expressed as percent of applied dose. The denominator $v = n - p$ in Eq. (73) is the number of

degrees of freedom in the fit. The significance of improvements in the fit with added parameters was ascertained by comparing the ratio $\chi^2_{v2} / \chi^2_{v1}$ with the value of $F_{v1,v2}$ at the desired level of significance (Bevington, 1969). Values of $F_{v1,v2}$ were obtained from (Abramowitz and Stegun, 1972). Equal weighting, $w_i = 1$, was used for all calculations. This choice emphasized the data at longer times (higher percents absorbed) and resulted in optimized parameters which best reconstruct the cumulative absorption values rather than the transient absorption rates.

The model-fitting procedure entailed repetitively running the numerical model with varying values of the adjustable parameters D_{SC} and f_{dep} and calculating the residuals for each trial according to Eq. (73). Parameters were optimized by carefully integrating Bevington's (1969) grid search (GRID SEARCH) and parabolic expansion (CHIFIT) methods with the chosen PDE-solving numerical algorithm of NAG® (Ray Chaudhuri et al., 2007).

4.3 Results and Discussion

4.3.1 Evaporation rates of ethanol and benzene from Franz cells

The results of the evaporation rate studies for ethanol and benzene from the diffusion cells are shown in Figs. 4.1-4.2. Linear weight loss profiles were observed in both the fume hood and benchtop environments, indicative of constant evaporation rates and correspondingly constant values of the mass-transfer coefficient k_{evap} in each trial. The evaporation rates were positively correlated with the surface temperature of the evaporating film, as expected from Eq. (71) and the strong temperature-dependence of P_{vp}. Values of k_{evap} and the air flow velocity calculated from these data according to Eq. (71) are shown in Table 4.3.

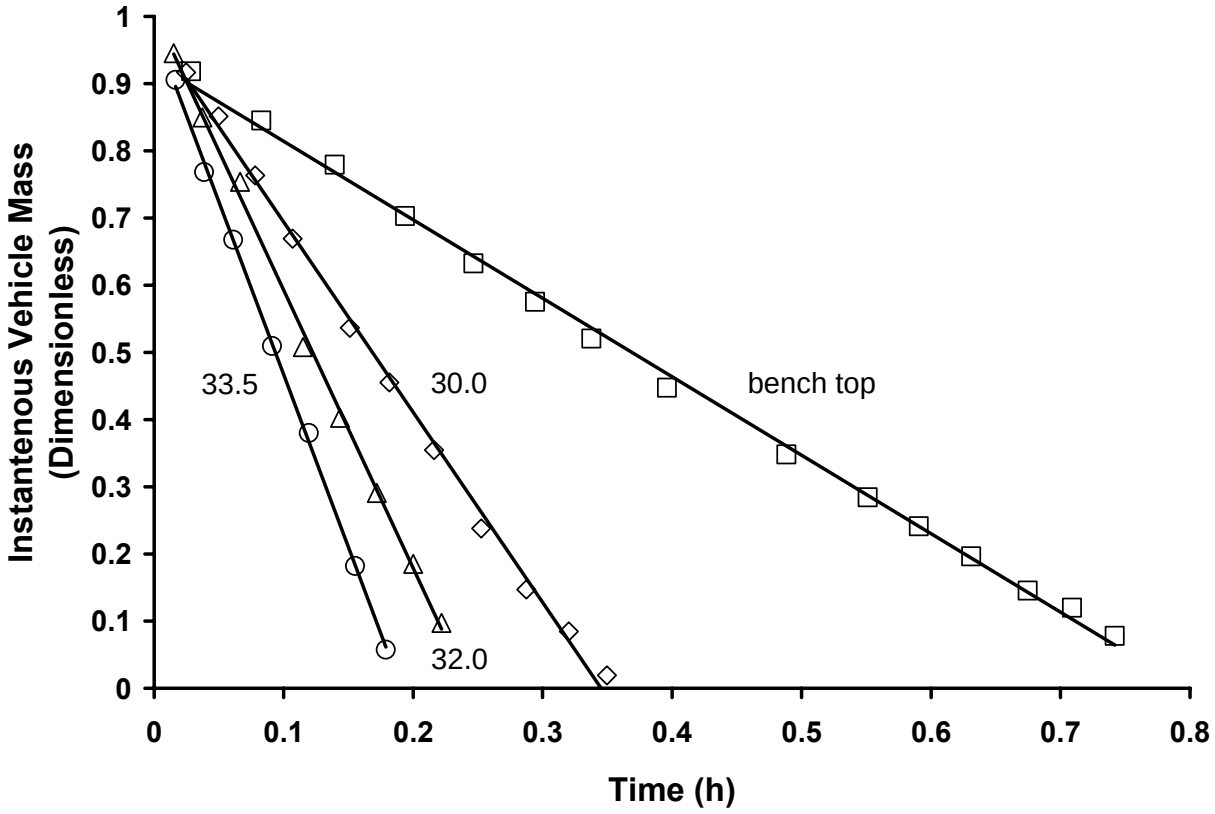


Fig. 4.1. Experimental evaporation rates of ethanol under fume-hood and bench-top conditions

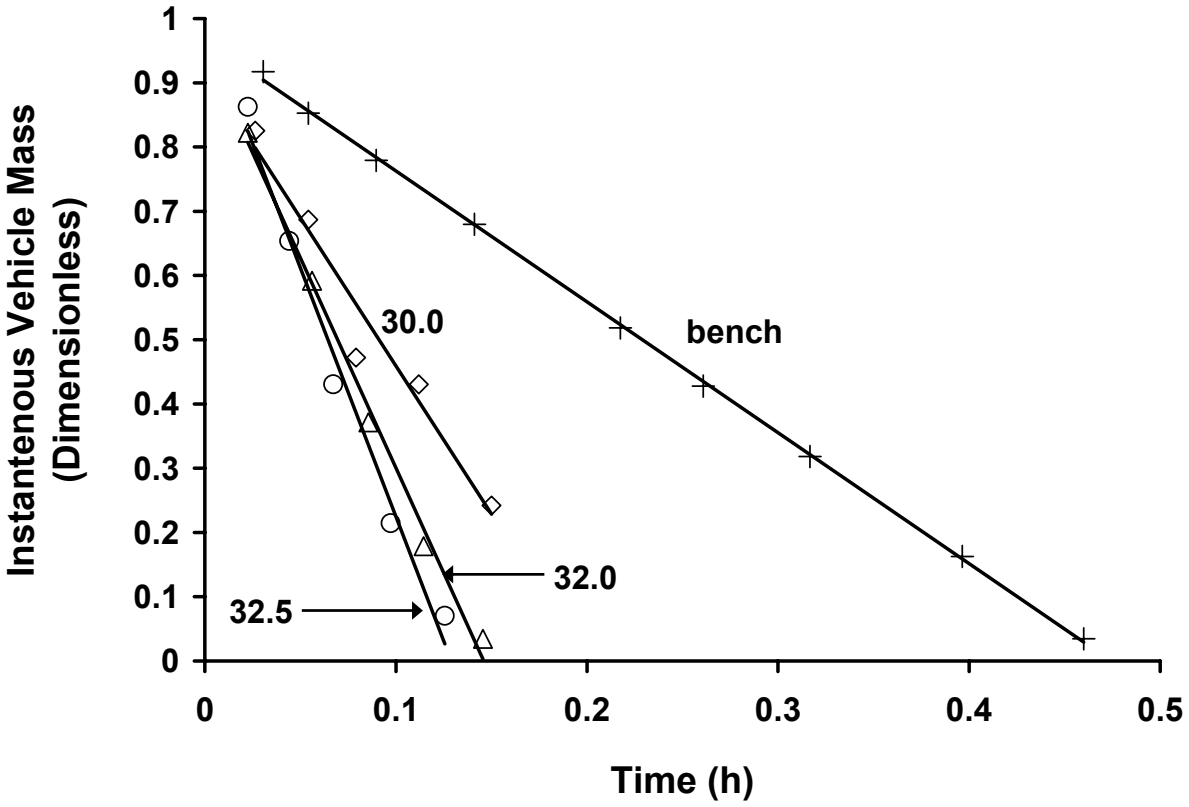


Fig. 4.2. Experimental evaporation rates of benzene under fume-hood and bench-top conditions

Table 4.3. Experimentally measured k_{evap} and airflow velocity for ethanol and benzene

Compound	Temperature (^{0}C)	Evaporative mass-transfer coefficient (k_{evap})	Airflow Velocity $u\,(m/s)$[a]
Ethanol	30.0	0.2866	0.5987
	32.0	0.4189	0.8528
	33.5	0.5209	1.022
			Mean: 0.8245 ± 0.2131
	32.9 (bench)	0.1183	0.1589
Benzene	30.0	0.4670	0.4732
	32.0	0.6617	0.6667
	32.5	0.7843	0.8076
			Mean: 0.6492 ± 0.1679
	34.1 (bench)	0.2063	0.1344

Average = 0.7368 m/s
[a] Calculated from Eqs. (71) and (72)

Values of k_{evap} in the fume hood environment exceeded those on the benchtop by a factor of 3.5 for ethanol and 3.6 for benzene. Table 4.3 also reports the effective air flow velocities u calculated according to the correlation given by Eqs. (71) and (72). It should be noted that this correlation was developed for evaporation from broad surfaces associated with chemical spills and is not expected to be quantitatively correct for evaporation from Franz cells. However, the values of u so calculated for various compounds can be compared. The values of u (m/s) for ethanol (0.825 ± 0.213) and benzene (0.649 ± 0.168) in the fume hood environment were not significantly different. However, they were substantially less than the value u ~ 1.5 m/s estimated for benzyl alcohol and DEET in the same environment by a somewhat different approach (Miller et al., 2006). The implications of this finding are discussed later.

4.3.2 Ethanol permeation through human skin

Results of the ^{14}C-ethanol skin permeation studies are shown in Table 4.4. Selected results and simulated absorption curves are shown in Figs. 4.3-4.4. Mean cumulative absorption of radioactivity ranged from 0.166% for the 40 μL dose to 0.395% for the 10 μL dose. A smooth trend of increasing absorption with decreasing dose was seen for the 40, 20 and 5 μL data; however, the 10 μL absorption results were higher than those at 5 μL. The mean values for the 10 μL data were skewed by high absorption in a few individual diffusion cells rather than representing consistently higher absorption rates. For this reason, the 10 μL data were exc;uded from the dataset used to fit the diffusion model parameters, D_{SC} and f_{dep}. Several variations on the parameter estimation process were explored. In all cases the model parameters in Table 4.2 (with the exception of D_{SC} and f_{dep}) and the average experimental value

Table 4.4. Cumulative Absorption Flux of ^{14}C-Ethanol through Human Cadaver Skin (Mean of three Donors, n = 5–6/ Donor)

Dose	Cumulative Absorption (% Dose Applied)									
	0.08	0.17	0.25	0.33	0.67	1.00	2.00	4.00	8.50	24.00
5	0.024	0.053	0.081	0.107	0.169	0.202	0.224	0.245	0.25	0.25
10	0.0174	0.062	0.115	0.157	0.269	0.324	0.367	0.384	0.391	0.395
20	0.006	0.021	0.041	0.061	0.117	0.149	0.179	0.192	0.197	0.201
40	0.003	0.011	0.025	0.043	0.094	0.12	0.145	0.154	0.159	0.166

u = 0.7368 m/s (Table 4.3) were employed. Variation A employed the values f_{dep} = 0.1 and D_{SC} = D_{WKN} = 9.29 × 10^{-10} cm^2/s estimated from previous studies and shown in Table 4.2. Variation B relaxed the restriction on D_{SC} (= 0.5172 D_{WKN}), whereas Variation C allowed both D_{SC} (= 0.5172 D_{WKN}) and f_{dep} (= 0.18) to be freely determined by the optimization routine. Results of this process are shown in Table 4.5 and Fig. 4.3. Variation A overpredicted the total absorption of the lowest dose by 5% (Fig. 4.3a) and that of the highest dose by 36% (Fig. 4.3b). Absorption rates were substantially overpredicted with variation A (as seen from the initial slopes of Curves A in Fig. 4.3), while they slightly underpredicted by variations B and C. Nevertheless, from the point of view of estimating systemic loads, this calculation may be regarded as successful. The root-mean-square deviation of the fit from the data, $s = \left(\chi_\nu^2\right)^{1/2}$ was 0.087% of the applied dose. Even the outlying 10 μL result was estimated to within 40% of the measured value or 0.16% of the applied dose.

With fractional deposition depth fixed at 0.1, an improved fit to the data was obtained with an SC diffusivity equal to 4.81 × 10^{-10} cm^2/s or 52% of the predicted value, D_{WKN} (Table 4.2). This change reduced the squared residual parameter χ_ν^2 from 0.00783 to 0.00226, a highly significant improvement in the fit ($F_{29,30}$ = 783/226 = 3.46). Allowing the value of f_{dep} to be freely determined from the data resulted in another significant improvement (χ_ν^2 = 0.00112, $F_{28,29}$ = 226/111 = 2.04). The optimum value of f_{dep} was 0.18 and that of D_{SC} was 4.81 × $10^{—10}$ cm^2/s or 52% of the predicted value D_{WKN} (Table 4.2). The decrease in D_{SC} versus D_{WKN} led to better estimations of the initial absorption rates, as well as total absorption (Fig. 4.3).

Table 4.5. Optimal f_{dep} and D_{SC} when data for all doses used simultaneously for regression

f_{dep}	D_{FIT}/D_{WKN}	χ^2 (*10^5)
0.100	0.6361	160.7
0.150	0.5759	123.0
0.170	0.5225	111.7
0.180	**0.5172**	**111.3**
0.190	0.4685	117.0
0.200	0.4422	126.0
0.250	0.3102	244.8
0.300	0.1946	460.5

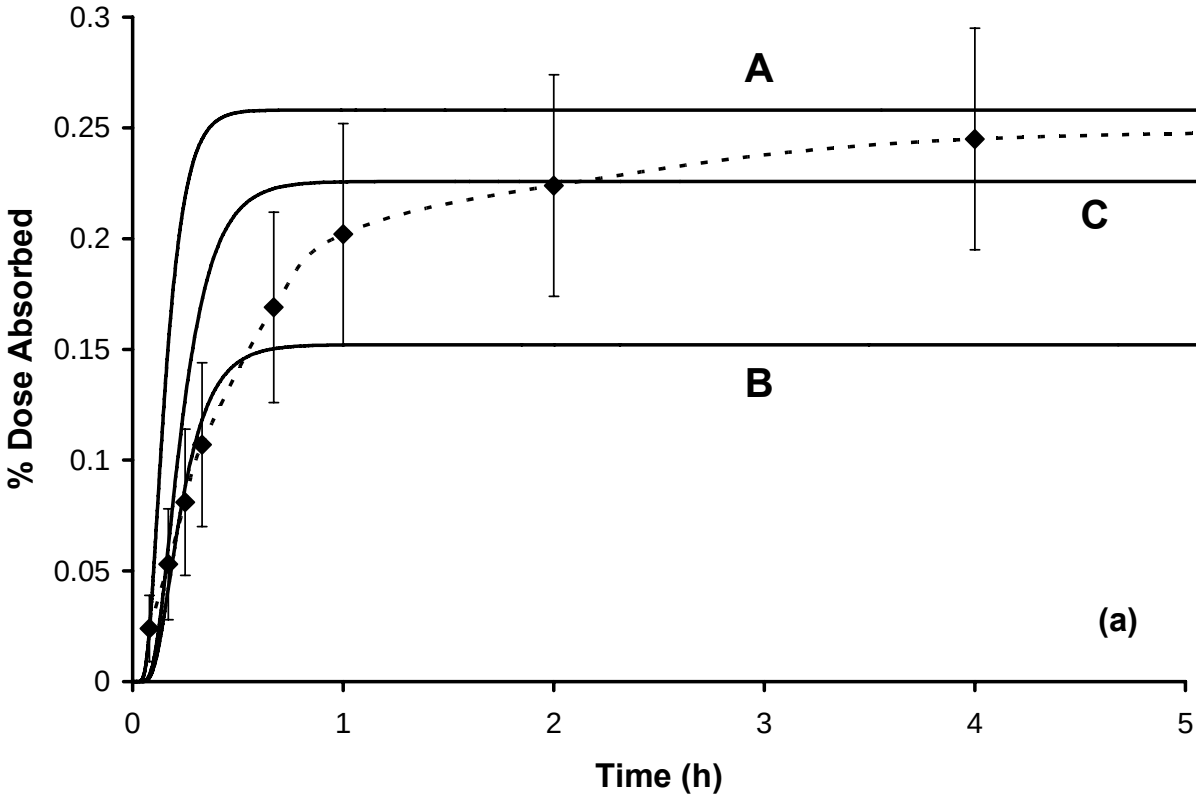


Fig. 4.3(a). Variation of fraction absorbed with time for a 5 μL dose of ethanol. Model variations A, B & C are discussed in the text.

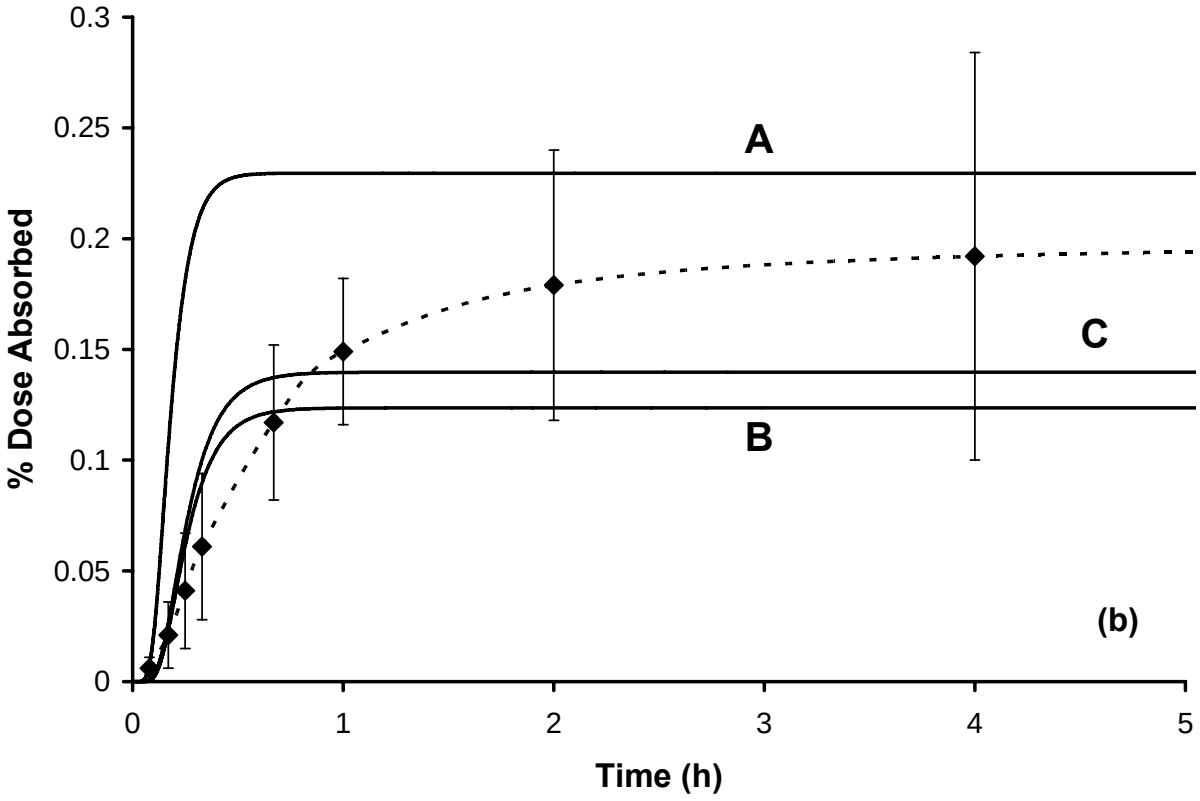


Fig. 4.3(b). Variation of fraction absorbed with time for a 20 μL dose of ethanol. Model variations A, B & C are discussed in the text.

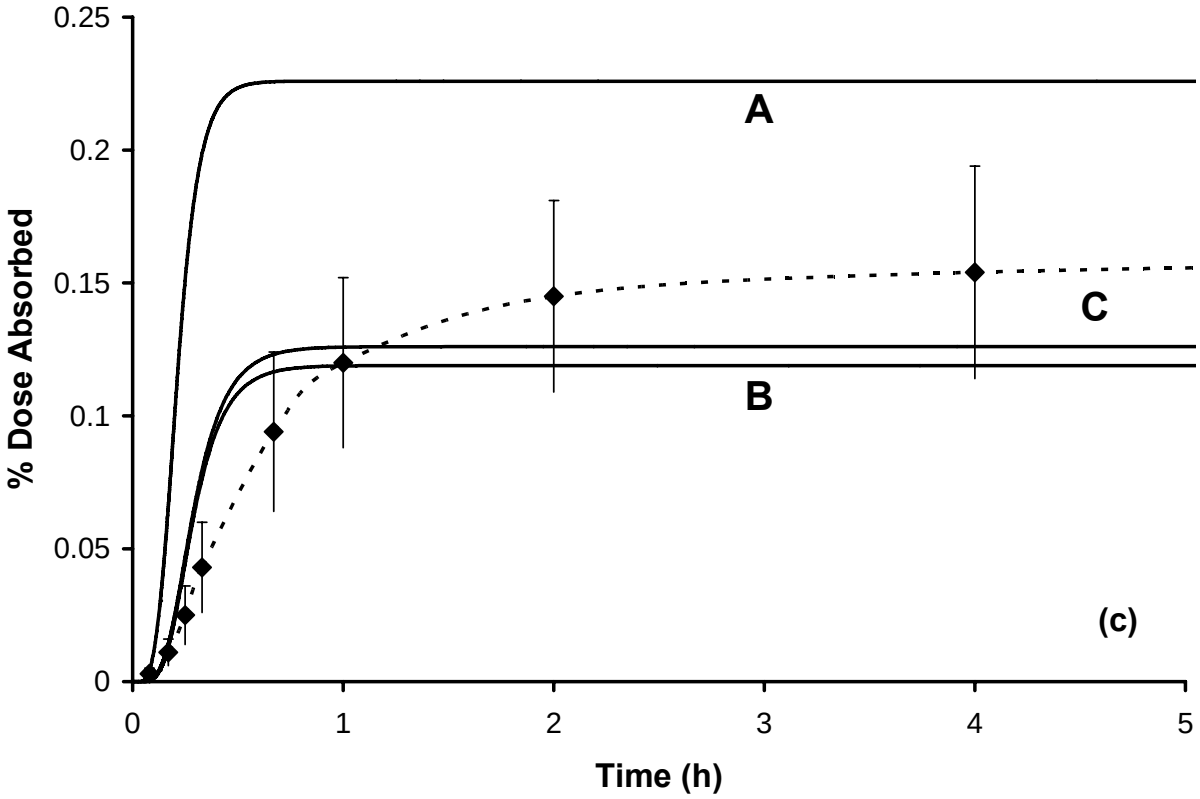


Fig. 4.3(c). Variation of fraction absorbed with time for a 40 μL dose of ethanol. Model variations A, B & C are discussed in the text.

Further improvements in the model were obtained by relaxing the constraint that f_{dep} and D_{SC} be considered constant values. Optimization of the value of f_{dep} separately for each dose (5, 20 and 40 μL) led to the results shown in Table 4.6 and Figure 4.4. A systematic increase in the value of f_{dep} with increasing dose was observed. Parameterization of this dependency according to the power law shown in Figure 4.5,

$$f_{dep} = 0.1305 \times \text{Dose}^{0.389} \tag{74}$$

led to Curves C in Figure 4.4 (χ_v^2 = 0.000524). Thus the additional parameter represented by Eq. (74) lead to a greater than two-fold reduction in the squared residuals versus a constant f_{dep} model ($F_{27,28}$ = 111/52.2 = 2.13). The value of D_{SC} associated with this fit was 2.17 × 10^{-10} cm^2/s, or 23% of D_{WKN} .

In all the analysis presented so far D_{SC} is a constant number. In reality, like any other diffusion coefficient, D_{SC} is a function of the local concentration of ethanol. Concentration of ethanol in SC can vary anywhere between zero and its saturation concentration, which means that D_{SC} will also vary between a minimum and maximum value. The minimum value may be determined from the intrinsic SC permeability in the absence of any sorbed ethanol (and corresponds to the dilute solution limit) and while the maximum value corresponds to the saturation concentration of ethanol. A plausible mathematical expression describing this concentration-varying behavior for SC is a sigmoidal function written as (Miller and Kasting, 2006):

$$D = D_0 + \frac{D_{SAT} - D_0}{1 + \exp\left[m\left(1 - C/C_{\text{inf}}\right)\right]} \tag{75}$$

where C is the instantaneous local concentration of permeant within SC, D_0 is the intrinsic SC diffusivity, D_{SAT} is the maximum or saturation diffusivity, $C_{\inf}$ represents the symmetry of the sigmoidal curve and m represents the slope of the line of inflexion. The most simplified representation of the concentration-dependent SC diffusivity would be that of a symmetric sigmoidal curve, where $C_{\inf}$ will be exactly half of the saturation concentration (C_{sat}). Also, based on an earlier work from this group, an optimal value of the slope of the line of inflexion was taken to be 5.0 (Miller et al., 2006). In the current work, since the correlation used to predict D_{SC} (D_{WKN}) involves the use of steady-state permeability data, it is assumed to represent the saturation limit or D_{SAT}. Once D_{SAT}, $C_{\inf}$ and m are determined, D_0 is the only unknown variable that needs to be optimized. Thus, in an alternative regression procedure D_{SC} is given by the expression in Eq. (75) and D_0 (instead of the constant D) is optimized. Now, along with D_0, D_{SAT} can also be optimized (multiple nonlinear regression analysis) in order to test our hypothesis that it is in fact given by D_{WKN}. In each case, the deviation of the optimum value of each diffusivity parameter (D, D_0 or D_{SAT}) from its predetermined values will reflect the strength of the assumptions used in order to predict their values. At first, both D_0 and D_{SAT} were optimized simultaneously, while systematically varying f_{dep}. The optimization process generated optimal value of D_{SAT}, which was very close to the predictive value of D_{WNK} ($D_{SAT} = 1.0380 D_{WKN}$). This verified our proposition that D_{SAT} is given by D_{WNK} and for the rest of the simulations, D_{SAT} was not optimized and held constant at D_{WKN}. D_0 was subjected to optimization using the grid search algorithm while varying f_{dep} systematically. Table 4.7 lists the optimal parameters for the variable diffusivity regression analysis. Even though the optimal

f_{dep} for this study came out to be 0.16, the χ_v^2 values for f_{dep} = 0.18 are very close to the global minima, Thus, an optimal combination of f_{dep} = 0.18, $D_0 = 0.22\,D_{WKN}$ and $D_{SAT} = D_{WKN}$ led to the best model fits obtained in our study. Both the initial absorption rates and the total absorption were well matched by this simulation. It should be noted that from a statistical viewpoint, this fit (χ_v^2 = 0.000446) was not significantly different from the variable f_{dep} model, χ_v^2 = 0.000524. The projections of the χ_v^2 surface shown in Tables 4.5 and 4.7 allow statistical error bounds to be placed on the deposition depth parameter f_{dep}. The critical value of F with ν = 28 and p = 0.05 is 1.87. Thus χ_v^2 ratios exceeding 1.87 indicate a statistically poorer fit at p = 0.05. Using this criterion, the 95 % confidence limits on f_{dep} are $0.18^{+0.06}_{-0.12}$ for the constant diffusivity model (Table 4.5) and $0.18^{+0.03}_{-0.07}$ for the variable diffusivity model (Table 4.7). These results are shown graphically in Fig. 4.6.

Table 4.6. Optimal f_{dep} and D_{SC} when data for individual doses used for regression

Dose (μL)	f_{dep}	D_{FIT}/D_{WNK}	χ^2 (*10^5)
5	0.210	0.3238	56.51
20	0.390	0.2022	20.06
40	0.500	0.1751	6.823

Table 4.7. Optimal f_{dep} and D_0 for variable-diffusivity regression

f_{dep}	D_0/D_{WNK}	$\chi^2 \times 10^5$
0.10	0.3246	90.20
0.15	0.2637	49.00
0.16	**0.2458**	**43.51**
0.17	0.2267	43.71
0.18	**0.2167**	**44.75**
0.20	0.1793	69.50
0.25	0.0995	226.5
0.30	0.0481	531.7

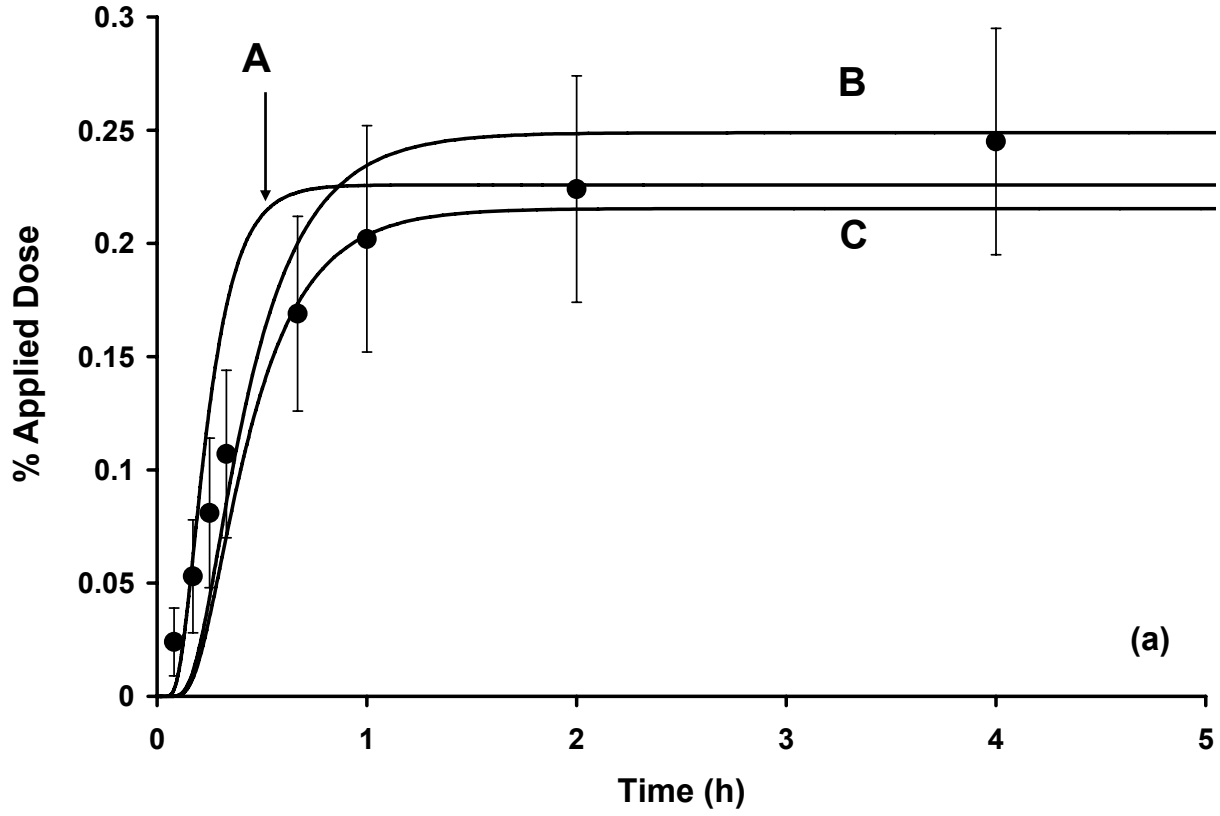


Fig. 4.4(a). Variation of fraction absorption for a 5 μL dose of ethanol. Systems A, B & C have been explained above.

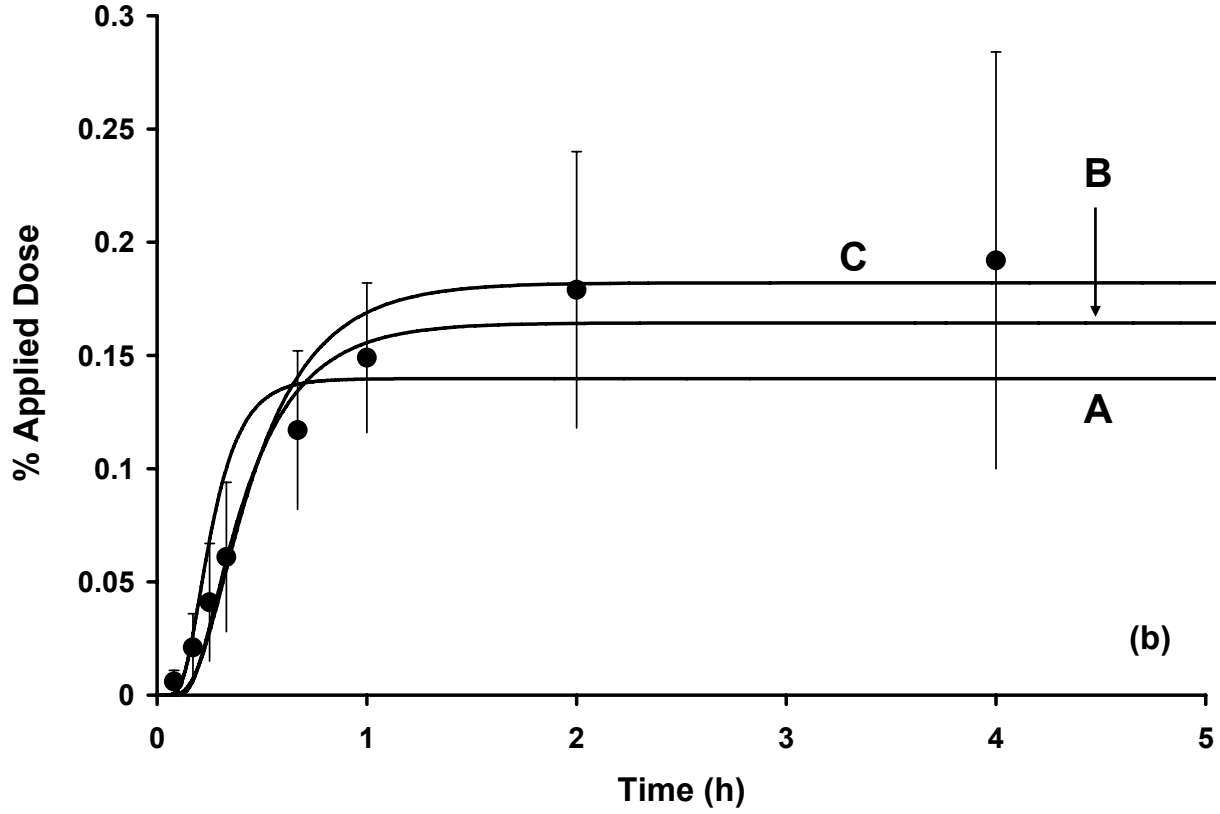


Fig. 4.4(b). Variation of fraction absorption for a 20 μL dose of ethanol. Systems A, B & C have been explained above.

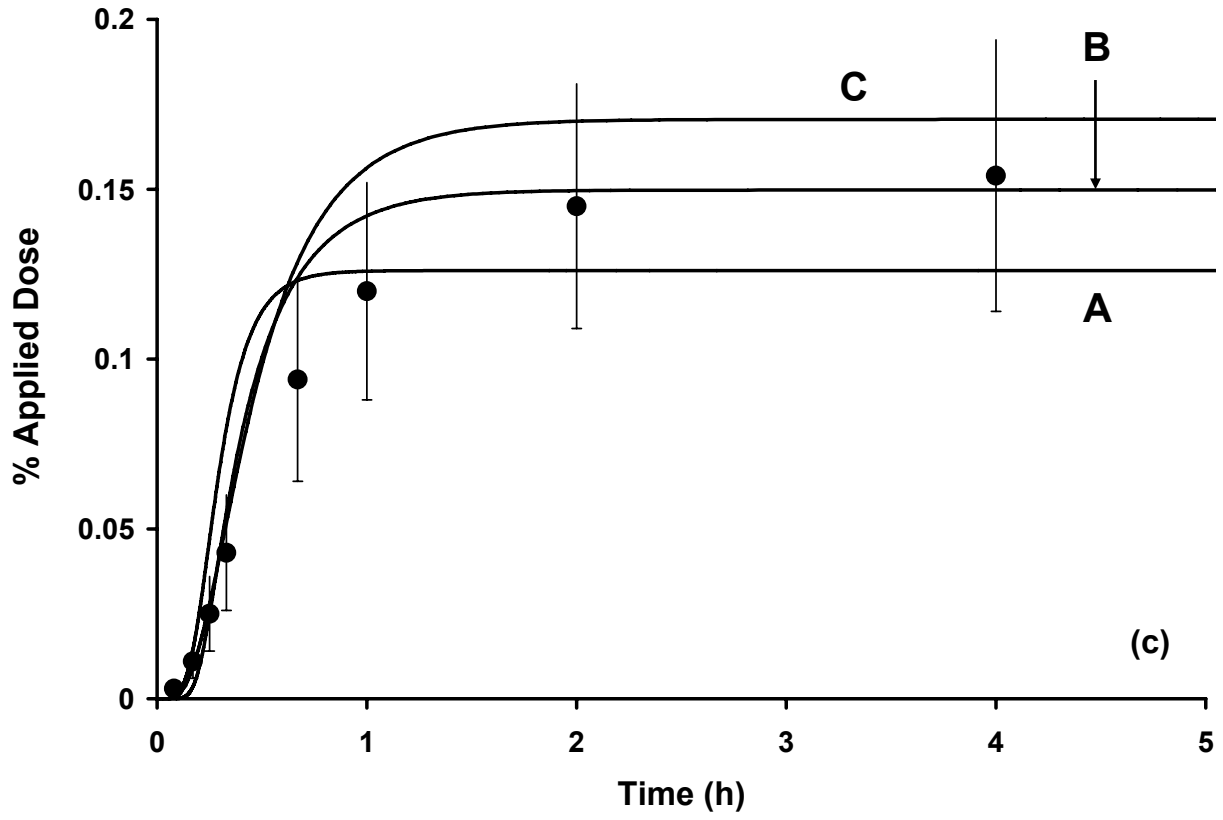


Fig. 4.4(c). Variation of fraction absorption for a 40 μL dose of ethanol. Systems A, B & C have been explained above.

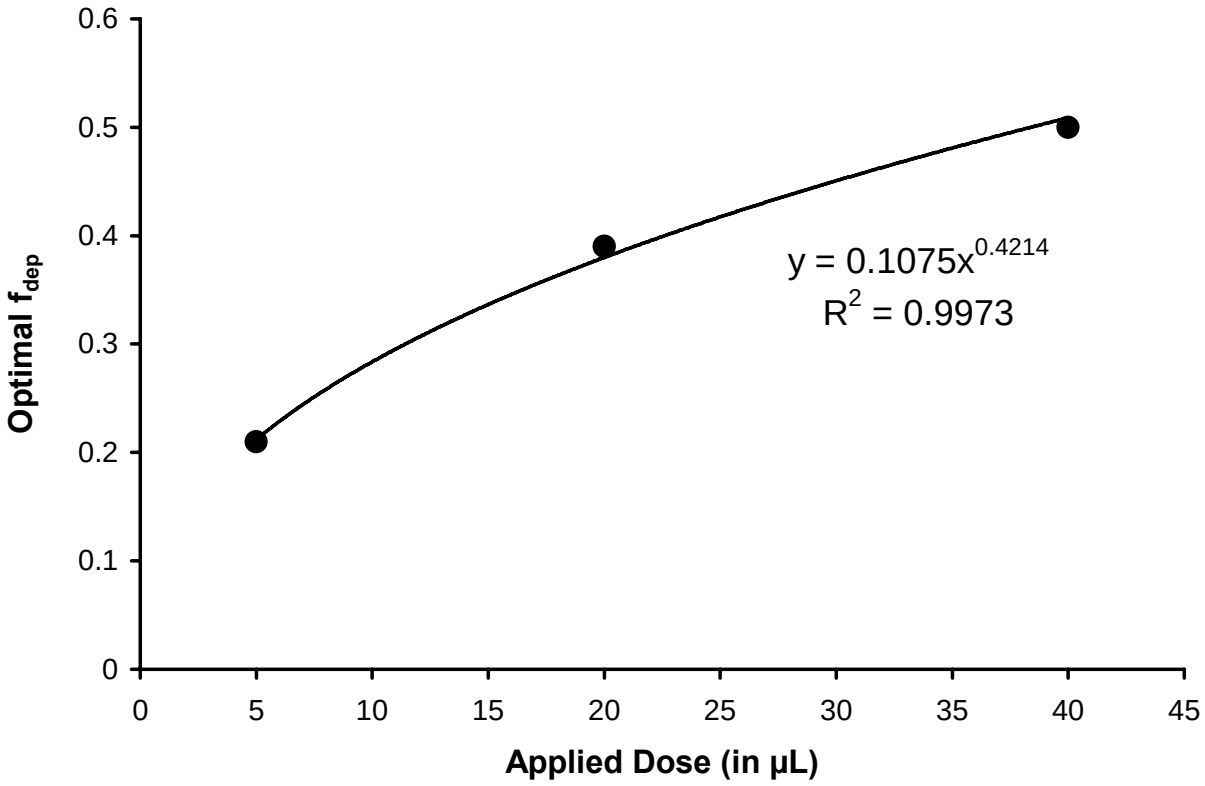


Fig. 4.5. Variation of optimal f_{dep} with varying applied dose.

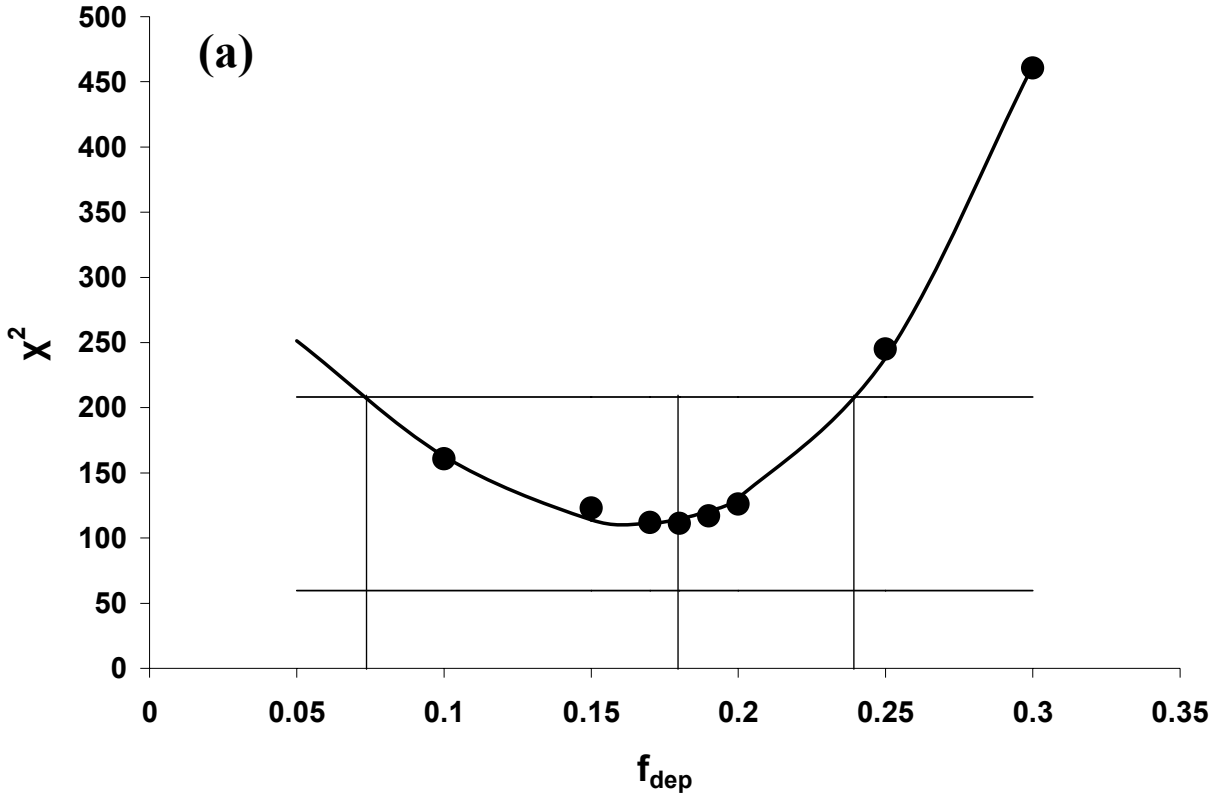


Fig. 4.6(a). Error surface and confidence intervals on χ_ν^2 and f_{dep}: constant diffusivity regression analysis.

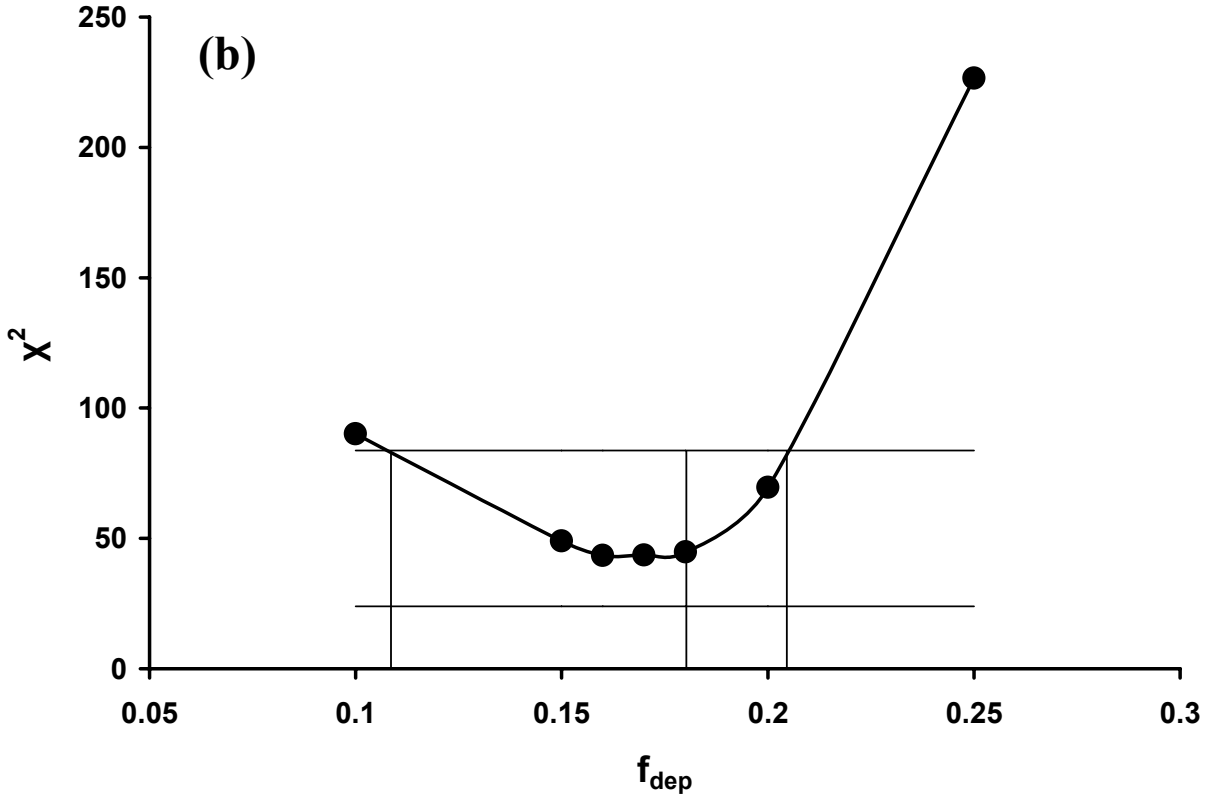


Fig. 4.6(b). Error surface and confidence intervals on χ_v^2 and f_{dep}: variable diffusivity regression analysis.

4.4 Conclusion

Percutaneous absorption of ethanol following transient liquid phase exposures was measured experimentally. The experimental data were found to be consistent with an unsteady-state mass transfer model using only two adjustable parameters, namely the fractional deposition depth and the diffusivity of ethanol inside the SC. Extensive non-linear regression analyses were performed on several versions of the mass transfer model. Even though the use of constant deposition depth and SC diffusivity resulted in good correlations, substantial improvements resulted through the use of concentration-dependent diffusivity and dose-dependent deposition depths. The constant and variable diffusivity models led to optimal f_{dep} values of 0.18. The variable f_{dep} model (Table 4.6) led to f_{dep} values in the range 0.21-0.50. All of these measures provide evidence that the optimum value of f_{dep} for ethanol is higher than the value f_{dep} = 0.1 determined for the larger molecular weight solutes benzyl alcohol (Miller et al., 2006) and DEET (Santhanam et al., 2005). The extent of agreement of these results is supportive of an eventual predictive use of the model for volatile liquids in general.

References

Abramowitz, M., Stegun, I.A., (Eds.), 1972. Handbook of Mathematical Functions, National Bureau of Standards, Washington, D.C., p. 288.

Ackermann, C., Flynn, G.L., Smith, W.M., (1987). Ether-water partitioning and permeability through nude mouse skin in vitro. II. Hydrocortisone 21-n-alkyl esters, alkanols and hydrophilic compounds. International Journal of Pharmaceutics, 36, 67-71.

Anderson, J.C., Hlastala, M.P., (2006). The kinetics of transdermal ethanol exchange. Journal of Applied Physiology, 100, 649-655.

Berner, B., Juang R.H., Mazzenga, G.C. (1989a). Ethanol and water sorption into stratum corneum and model systems. Journal of Pharmaceutical Sciences, 78, 472-476.

Berner, B., Mazzenga, G.C., Otte J.H., Steffens R.J., Juang R.H., Ebert C.D. (1989b). Ethanol:water mutually enhanced transdermal therapeutic system II: skin permeation of ethanol and nitroglycerin. Journal of Pharmaceutical Sciences, 78, 402-407.

Bevington, P.R., (1969). Data Reduction and Error Analysis for the Physical Sciences. McGraw-Hill, New York.

Brand, R.M., Charron, A.R., Dutton, L., Gavlik, T.L., Mueller, C., Hamel, F.G., Chakkalakal, D., Donohue, T.M. Jr., (2004). Effects of chronic alcohol consumption on dermal penetration of pesticides in rats. Journal of Toxicology and Environmental Health, 67, 153-161.

Brand, R.M., Jendrzejewski, J.L., Henery, E.M., Charron, A.R., (2006). A single oral dose of ethanol can alter transdermal absorption of topically applied chemicals in rats. Toxicological Sciences, 92, 349-355.

Cleek, R.L., Bunge, A.L., (1993). A new method for estimating dermal absorption from chemical exposure. 1. General approach. Pharmaceutical Research, 10, 497-506.

Cross, S.E., Magnusson, B.M., Winckle, G., Anissimov, Y., Roberts, M.S., (2003). Determination of the effect of lipophilicity on the in vitro permeability and tissue reservoir characteristics of topically applied solutes in human skin layers. Journal of Investigative Dermatology, 120, 759-764.

Durrheim, H., Flynn, G.L., Higuchi, W.I., Behl, C.R., (1980). Permeation of hairless mouse skin I: Experimental methods and comparison with human epidermis permeation by alkanols. Journal of Pharmaceutical Sciences, 69, 781-786.

Franz, T.J., Lehman, P.A., (1990). The use of water permeability as a means of validation for skin integrity in in vitro percutaneous absorption studies. Journal of Investigative Dermatology, 94, 525.

Johnson, M.E., Blankschtein, D., Langer, R., (1997). Evaluation of solute permeation through the stratum corneum: lateral bilayer diffusion as the primary transport mechanism. Journal of Pharmaceutical Sciences, 86, 1162-1172.

Kasting, G.B., Bowman, L.A., (1990). DC electrical properties of frozen, excised human skin. Pharmaceutical Research, 7, 134-143.

Kasting, G.B., Filloon, T.G., Francis, W.R., Meredith, M.P., (1994). Improving the sensitivity of in vitro skin penetration experiments. Pharmaceutical Research, 11, 1747-1754.

Kasting, G.B., Miller, M.A., (2006). Kinetics of finite dose absorption through skin 2. Volatile compounds. Journal of Pharmaceutical Sciences, 95, 268-280.

Khalil, E., Kretsos, K., Kasting, G.B., (2006). Glucose partition coefficient and diffusivity in the lower skin layers. Pharmaceutical Research, 23, 1227-1234.

Kretsos, K., Kasting, G.B., Nitsche, J.M., (2004). Distributed diffusion-clearance model for transient drug distribution within the skin. Journal of Pharmaceutical Sciences, 93, 2820-2835.

Martin, A., (1993). Physical Pharmacy, Lippincott Wiliams & Wilkins, Baltimore, MD, p. xxx.

Merritt, E.W., Cooper, E.R., (1984). Diffusion apparatus for skin penetration. Journal of Controlled Release, 1, 161-162.

Miller, M.A., Bhatt, V., Kasting, G.B., (2006). Dose and airflow dependence of benzyl alcohol disposition on skin. Journal of Pharmaceutical Sciences, 95 (2), 281–291.

N'Dri-Stempfer, B., Bunge, A.L., (2005). How much can evaporation of dermally absorbed chemical reduce the systemically absorbed dose? Occupational and Environmental Exposure of Skin to Chemicals, pp. 145-146.

Nitsche, J.M., Wang, T.F., Kasting, G.B., (2006). A two-phase analysis of solute partitioning into the stratum corneum. Journal of Phamaceutical Sciences, 95, 649-666.

Peress, J., (2003). Estimate evaporative losses from spills. Chemical Engineering Progress, 32-34.

Santhanam, A., Miller, M.A., Kasting, G.B., (2004). Absorption and evaporation of N, N-diethyl-m-toluamide from human skin in vitro. Toxicology and Applied Pharmacology, 204, 81-90.

Scheuplein, R.J., (1965). Mechanism of percutaneous absorption I. Routes of penetration and the influence of solubility. Journal of Investigative Dermatology, 45, 334-346.

Scheuplein, R.J., (1967). Molecular structure and diffusional processes across intact epidermis. U.S. Army, Edgewood Arsenal.

Scheuplein, R.J., Blank, I.H., (1973). Mechanism of percutaneous absorption. IV. Penetration of nonelectrolytes (alcohols) from aqueous solutions and from pure liquids. Journal of Investigative Dermatology, 60, 286-296.

Vuilleumier, C., Flament, I., Sauvegrain, P., (1995). Headspace analysis study of evaporation rate of perfume ingredients applied onto skin. International Journal of Cosmetic Sciences, 17, 61-76.

Wang, T.-F. et al. (2006). A multiphase microscopic model for stratum corneum permeability. I. Formulation, solution and illustrative results for representative compounds. Journal of Pharmaceutical Sciences, 95, 620-648.

Wang, T.F., Kasting, G.B., Nitsche, J.M., (2007). A multiphase microscopic model for stratum corneum permeability. II. Estimation of physicochemical parameters and application to a large permeability database. Journal of Pharmaceutical Sciences, article in press.

Williams, A.C., Barry, B.W., (2004). Penetration enhancers. Advanced Drug Delivery Reviews, 56, 603-618.

Williams, P.L., Brooks, J.D., Inman, A.O., Monteiro-Riviere, N.A., Riviere, J.E., (1994). Determination of physicochemical properties of phenol, p-nitrophenol, acetone and ethanol relevant to quantifying their percutaneous absorption in porcine skin. Research Communication in Chemical Pathology and Pharmacology, 83, 61-75.

Yamazki, K., Kanaoka, M., (2004). Computational Prediction of the Plasma Protein-Binding Percent of Diverse Pharmaceutical Compounds, Journal of Pharmaceutical Sciences, 93, 1480-1494.

CHAPTER 5

PERCUTANEOUS PERMEATION OF BENZENE

5.1 Background

The human population is exposed daily to benzene since it is one of the most common organic solvents used in the industry. Inside the human body, benzene has several adverse toxicological and carcinogenic effects (Medlin, 2003). Even though it has been identified that liquid benzene can contact the skin and be absorbed percutaneously (Blank and McAuliffe, 1985), there are limited experimental data regarding benzene absorption through the skin (Maibach and Anjo, 1981). These have been summarized by Blank and McAuliffe (1985). Most of these skin penetration data are steady-state in vivo experiments, although some represent in vitro or finite dose (transient) studies. Even though the translation of in vitro experimental data to predict in vivo absorption is debatable, with in vitro techniques it is easier to investigate certain parameters that are more difficult to investigate in vivo (Blank and McAuliffe, 1985).

The objective of this chapter is to characterize the skin disposition (evaporation from and absorption into the human skin) under unoccluded conditions of benzene in terms of a mass-transport model described in Chapter 2 of this dissertation. This is a direct extension of the analysis performed on skin disposition of ethanol, as discussed in Chapter 4 of this dissertation. Similar to ethanol, this analysis involves relating the topical disposition of benzene to physicochemical properties and exposure variables in a way that can be extrapolated to new compounds and exposure conditions. In this study, the permeation of pure benzene through human skin in vitro has been experimentally measured and the absorption data have been analyzed using the accompanying mathematical model.

5.2 Materials and Methods

5.2.1 Chemicals

[^{14}C] benzene at specific activity of 15.6 mCi/mmol was purchased from Sigma-Aldrich (St. Louis, MO). Radiochemical purities as stated by the suppliers were 100% and 99.34 %, respectively. Unlabeled neat benzene and calcium-free Dulbecco's phosphate-buffered saline were purchased from Sigma-Aldrich (St. Louis, MO). Ultima Gold Scintillation fluid and Soluene-350 were purchased from Perkin-Elmer Biosciences. All other chemicals used were reagent grade.

5.2.2 Permeation methodology and Evaporative Rate Measurements

The experimental methodology for benzene pertaining to tissue preparation, handling, permeation rate experiments as well as evaporation rate measurements of benzene are identical to those for ethanol and have already been discussed in detail in Section 4.2 of this dissertation.

5.2.3 Estimation of Transport and Thermodynamic Parameters

As in ethanol, application of the transport model (described in Chapter 2 of this dissertation) to benzene requires as inputs a number of vehicle and tissue dimensions, physicochemical properties of the permeants and their diffusivities and partition coefficients in the two skin layers, designated SC (stratum corneum) and VT (viable tissues). Also, as in ethanol, the viable epidermis and residual dermis in the split-thickness skin samples are treated as a single layer with the properties of unperfused dermis (Kretsos and Kasting, 2005; Kretsos et al., 2006. It is particularly appropriate in cases where accurate estimates of skin concentrations are not required and the lower skin layers are not rate-determining for absorption. Such is the case for benzene, as will be shown.

Relevant physicochemical properties for benzene transport in skin are readily obtained from the literature and are listed in Table 5.1. These properties were used as input parameters for the previously developed correlations for transport parameters and partition coefficients summarized in Table 5.2. Parameters for which the present analysis departs from the literature values are discussed below.

5.2.3.1 Layer thicknesses

The initial thickness of the vehicle layer deposited on the skin surface was calculated from the volume of benzene applied, its density at 32°C and the diffusion cell cross-sectional area. A correction was made for the amount of benzene assumed to be rapidly deposited into the SC deposition layer h_{dep} (Kasting and Miller, 2006). The unswollen SC thickness was taken to be 13.4 μm (Johnson et al., 1997; Wang et al., 2006); this value was adjusted upward in a concentration-dependent manner to account for swelling by benzene following application of the dose. A zero volume of mixing assumption resulting in a linear density correction was used for this adjustment. The thickness of the VT layer was taken to be 500 μm. This sample thickness for in vitro studies typically results when cadaver skin dermatomed to a nominal thickness of 300 μm swells in a saline buffer (Khalil et al., 2006).

Table 5.1. Physicochemical properties of benzene for skin disposition estimation (T = 32.0 ^{o}C)

Property	Symbol	Units	Value	Ref.
Molecular weight	MW	g/mol	78.11	Yaws (1999)
Density	ρ	g/cm^3	0.8679	Yaws (1999)
Vapor pressure	P_{vp}	torr	129.1	Yaws (1999)
Log (octanol/water partititon coefficient	log K_{oct}		2.130	Yaws (1999)
Water solubility	S_w	g/L	1.787×10^{-3}	Yaws (1999)
Fraction unbound to albumin	f_u		0.700	Yamazaki & Kanaoka (2004)
Fraction nonionized	f_{non}		1	Martin (1993)

Table 5.2. Thermodynamic and transport properties for estimation of benzene skin disposition

Property	Units	Value	Correlation	Reference
			Vehicle layer (VH)	
u	m/s	0.7368	Non-linear regression	Chapter 4
k_{evap}	cm/h	0.722	$k_g = 6320\,u^{0.78}/MW^{1/3}$ $k_{evap}\rho = k_g P_{vp} MW/\left(7.6\times10^{-5} RT\right)$	Peress (2003), N-Dri-Stempfer & Bunge (2005), Kasting & Miller (2006)
			Stratum corneum (SC)	
L_{SC}	μm	13.40		Johnson et al. (1996), Wang et al. (2006)
ρ_{SC}	g/cm^3	1.194	$\rho_{SC} = (1+v)/\left(\omega_{lip}/\rho_{lip} + \omega_{pro}/\rho_{pro} + v/\rho_w\right)$	Nitsche et al. (2006)
D_{SC}	cm^2/s	4.254×10^{-10}	WKN correlation	Wang et al. (2007)
$K_{SC/VH}$		0.0365	WKN correlation	Wang et al. (2007)
			Viable tissue (VT)	
L_{VT}	μm	500.0		Khalil et al. (2006)
ρ_{VT}	g/cm^3	1.019	$\rho_{VT}\left(g/cm^3\right) = \left(\omega_{H_2O} + 0.649\omega_{pro} + 1.227\omega_{lip}\right)^{-1}$	Anderson et al. (2000)
D_{free}	cm^2/s	4.077×10^{-6}	$\log D_{free} = -4.15 - 0.655\log MW$	unpublished result
D_{VT}	cm^2/s	1.542×10^{-6}	$D_{VT} = D_{free}/\left(0.68 + 0.32/f_u + 0.001 f_{non} K_{oct}\right)$	unpublished result
$K_{VT/SC}$		0.0894		Cross et al. (2003)

5.2.3.2 Diffusivity in stratum corneum (D_{SC})

The diffusivity of benzene in the SC was estimated from the skin permeation data by a least squares fitting procedure. Its value was compared to the estimate D_{WKN} = 4.254 × 10^{-10} cm^2/s (Table 5.2) derived from the correlation for partially hydrated skin proposed by (Wang et al., 2007). In one version of the fitting procedure, a concentration-dependent diffusivity of the form proposed by (Miller et al., 2006) was tested, that is,

$$D_{SC} = D_0 + \frac{D_{sat} - D_0}{1 + \boldsymbol{exp}\left[m\left(1 - C / C_{\boldsymbol{inf}}\right)\right]}. \qquad (76)$$

In Eq. (76), C is the instantaneous local concentration of permeant within SC, D_0 is the intrinsic SC diffusivity, D_{sat} is the maximum or saturation diffusivity, $C_{\boldsymbol{inf}}$ represents concentration corresponding to the inflection point of the sigmoidal curve and m describes its slope. The values of $C_{\boldsymbol{inf}}$ and m were taken to be $C_{sat}/2$ and 5.0 as in (Miller et al., 2006), whereas the values of D_0 and D_{sat} were determined from the data.

5.2.3.3 Fractional deposition depth in SC *(f_{dep})*

Similar to previous studies from our group and the analysis of ethanol permeation, this parameter was introduced. A value of 0.1 was proposed by (Kasting and Miller, 2006) and supported experimentally for benzyl alcohol (Miller et al., 2006) and DEET (Santhanam et al., 2005). It was clear from the analysis of ethanol that a higher value of f_{dep} is necessary to appropriately describe the disposition characteristics of volatile solvents in small doses. In the present study it was of interest to determine whether the absorption of benzene would be

effectively described by the same value of f_{dep} as the larger molecules, benzyl alcohol and DEET or by a value higher than 0.1, as observed in the case of ethanol (Chapter 4).

5.2.4 Regression analysis

The model parameters in Table 5.2 were used without modification to calculate the time course of absorption and evaporation of benzene from skin according to the described model (Chapter 2), with three exceptions: k_{evap}, D_{SC} and f_{dep}. These parameters were varied to provide the best fits of the model to the experimental data. They were selected because their values were believed to be the least accurately predicted by the developed correlations (Table 5.2) and each plays a key role in determining the rate and extent of benzene absorption. The value of k_{evap} was determined from the evaporation rate experiments, after which the values of D_{SC} and f_{dep} were determined from the absorption data. Similar to ethanol, values of k_{evap} for each evaporation trial were determined by linear regression. The result of such evaporation rate measurements for benzene is already given in Fig. 4.2 of this dissertation. The process of decomposing experimentally evaluated k_{evap} to gas-phase mass transfer coefficients (k_g) and airflow velocity (u) are already discussed in Section 4.3.1 of this dissertation and the final result given in Table 4.3.

Values of D_{SC} and f_{dep} were determined by nonlinear regression by minimizing the normalized sum of squared residuals χ_v^2, defined by Bevington (1969)

$$\chi_v^2 = \frac{1}{n-p}\sum_{i=1}^{n} w_i \left[y_i^{obs} - y_i^{fit} \right]^2 \qquad (77)$$

Here n is the number of observations, p is the number of adjustable parameters, w_i is the weight, y_i^{obs} and y_i^{fit}, respectively are the observed and calculated values of cumulative absorption flux expressed as percent of applied dose. The denominator $v = n - p$ in Eq. (77) is the number of degrees of freedom in the fit. The significance of improvements in the fit with added parameters was ascertained by comparing the ratio $\chi^2_{v2} / \chi^2_{v1}$ with the value of $F_{v1,v2}$ at the desired level of significance (Bevington, 1969). Values of $F_{v1,v2}$ were obtained from (Abramowitz and Stegun, 1972). Equal weighting, $w_i = 1$, was used for all calculations. This choice emphasized the data at longer times (higher percents absorbed) and resulted in optimized parameters that best reconstruct the cumulative absorption values rather than the transient absorption rates.

The model fitting procedure entailed repetitively running the numerical model with varying values of the adjustable parameters D_{SC} and f_{dep} and calculating the residuals for each trial according to Eq. (77). Parameters were optimized by carefully integrating Bevington's (1969) grid search (GRID SEARCH) and parabolic expansion (CHIFIT) methods with the chosen PDE-solving numerical algorithm of NAG® (Chapter 2).

5.3 Results and Discussion

5.3.1 Benzene permeation through human skin

Results of the ^{14}C-benzene skin permeation studies are shown in Table 5.3. Selected results and simulated absorption curves are shown in Figures 5.3-5.4. Mean cumulative absorption of radioactivity ranged from 0.0464% for the 40 μL dose to 0.0581% for the 5 μL dose. A smooth trend of increasing absorption with decreasing dose was seen for the 40, 20, 10 and 5 μL data. Unlike the ethanol absorption data, there were no outliers and hence all the

Table 5.3. Cumulative Absorption Flux of ^{14}C-Benzene through Human Cadaver Skin (Mean of three Donors, n = 5–6/ Donor)

Dose	Cumulative Absorption (% Dose Applied)									
	0.08	0.17	0.25	0.33	0.67	1.00	2.00	4.00	8.50	24.00
5	0.0095	0.0173	0.0236	0.0284	0.0413	0.0470	0.0526	0.0553	0.0567	0.0581
10	0.0043	0.0113	0.0176	0.0222	0.0340	0.0395	0.0441	0.0481	0.0495	0.0510
20	0.0030	0.0097	0.0164	0.0221	0.0342	0.0399	0.0451	0.0474	0.0486	0.0501
40	0.0021	0.0089	0.0161	0.0215	0.0326	0.0376	0.0415	0.0434	0.0447	0.0464

absorption data have been included in the regression analysis. Several variations on the parameter estimation process were explored. In all cases the model parameters in Table 5.2 (with the exception of D_{SC} and f_{dep}) and the average experimental value u = 0.7368 m/s (Table 4.3) were employed. Variation A employed the values f_{dep} = 0.1 and $D_{SC} = D_{WKN}$ = 4.254 × 10^{-10} cm^2/s as shown in Table 5.2. Variation B relaxed the restriction on D_{SC} (= 11.7029 D_{WKN}), whereas Variation C allowed both D_{SC} (= 11.7029 D_{WKN}) and f_{dep} (= 0.36) to be freely determined by the optimization routine. Results of this process are shown in Table 5.5 and Fig. 5.1. Unlike ethanol, all the variations underpredicted the experimental absorption. Absorption rates were similar but the net amount of absorption was underpredicted from all the variations. Nevertheless, from the point of view of estimating systemic loads, this calculation may be regarded as successful.

With fractional deposition depth fixed at 0.1, an improved fit to the data was obtained with an SC diffusivity equal to 4.979 × 10^{-9} cm^2/s or 11.7029 times the predicted value, D_{WKN} (Table 5.2). It is noteworthy that, contrary to the ethanol analysis where the predicted D_{WKN} had to be reduced in order to obtain a better fit, the D_{WKN} value for benzene had to be raised by an order of magnitude in order to achieve similar fits. This is because the value of the SC diffusivity in case of ethanol had been estimated from steady-state permeation data under saturated conditions (Berner et al., 1989). However, in the case of benzene, they have been estimated from a correlation which derives form the dilute solution limit with no permeation enhancement (Wang, 2003). This change reduced the squared residual parameter χ_v^2 from 0.00105 to 0.00021, a highly significant improvement in the fit ($F_{29,30}$ = 105/21 = 5.00). Allowing the value of f_{dep} to be freely determined from the data resulted in another significant improvement (χ_v^2 = 0.00011,

Table 5.4. Optimal f_{dep} and D_{SC} when data for all doses used simultaneously for regression

f_{dep}	D_{FIT}/D_{GBK}	χ^2 (*10^5)
0.200	18.4079	13.1616
0.340	12.1999	10.9427
0.350	12.1033	10.9398
0.360	**11.7029**	**10.9101**
0.370	11.4445	10.9316
0.380	10.2880	11.2238
0.400	9.7345	11.8397
0.420	10.0851	12.3535
0.450	12.3028	15.8549
0.500	14.5209	23.1404
0.600	57.2318	37.4534

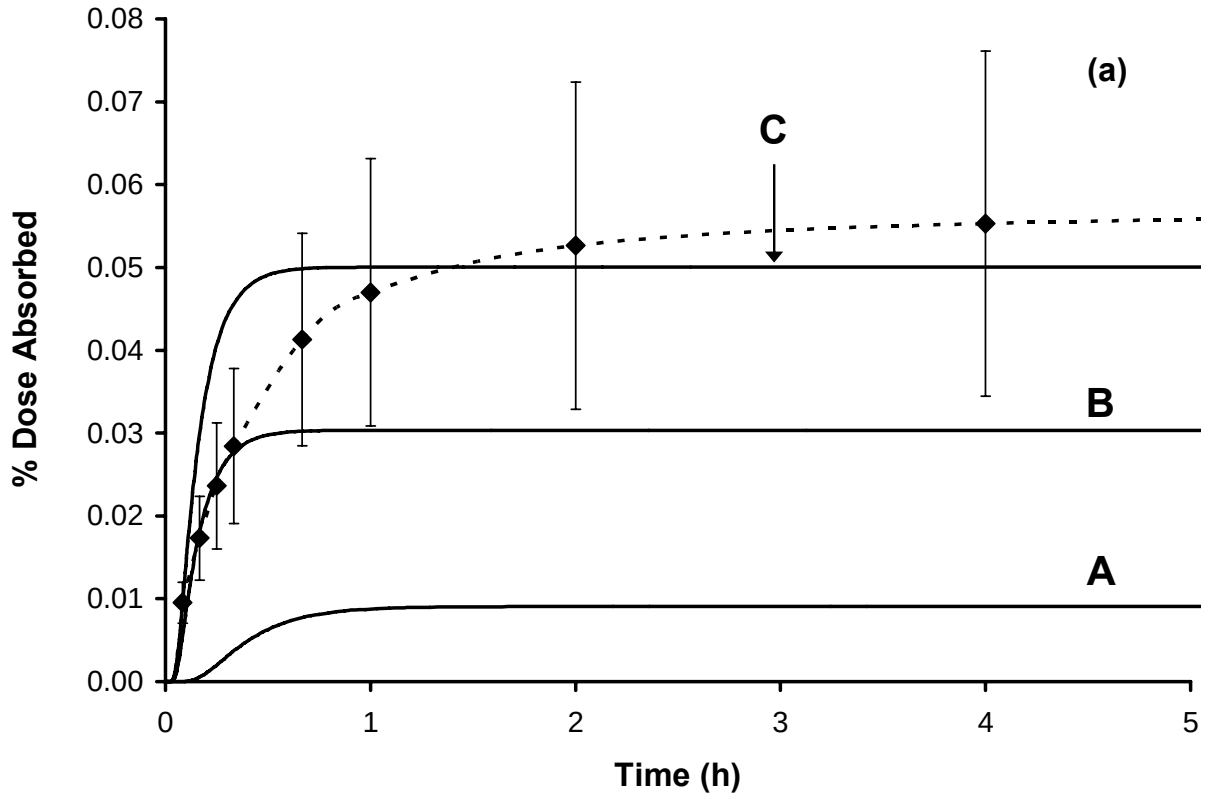


Fig. 5.1(a). Variation of fraction absorbed with time for a 5 μL dose of benzene. The line (---) is simply a guide to the eye. Systems A, B and C have been described in the text.

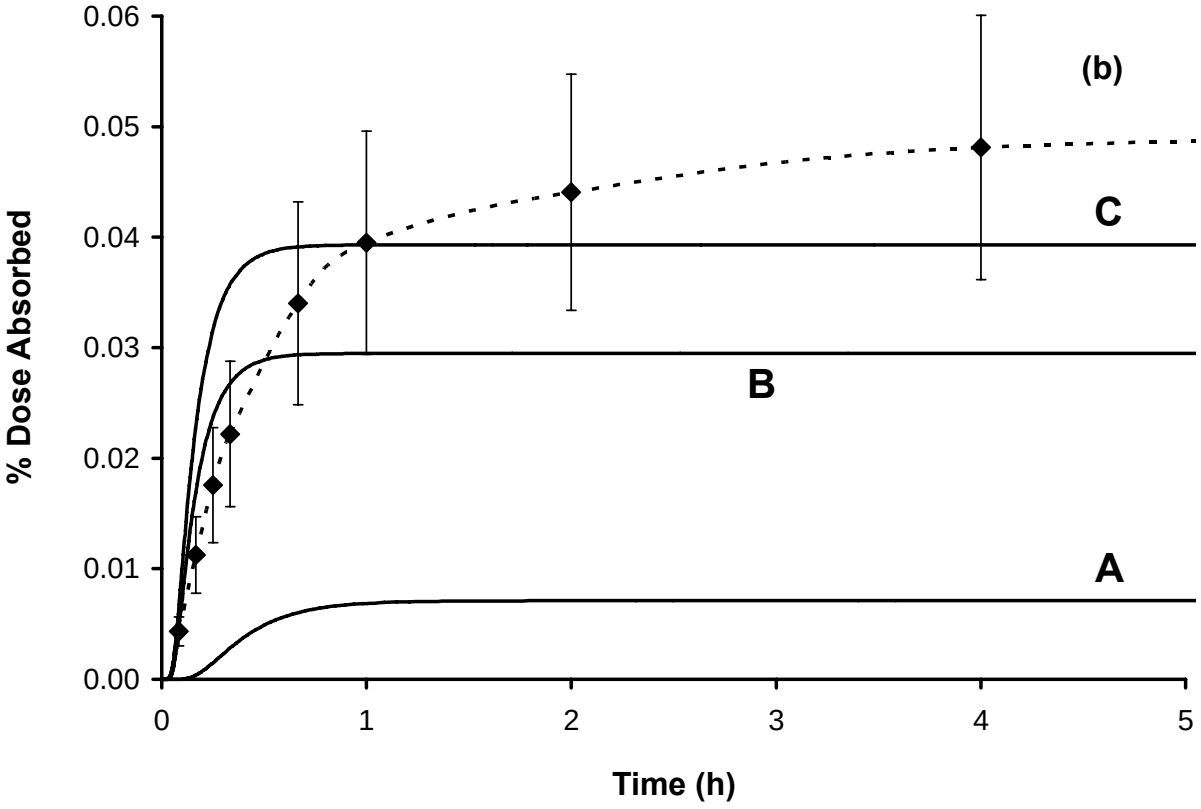


Fig. 5.1(b). Variation of fraction absorbed with time for a 10 μL dose of benzene. The line (---) is simply a guide to the eye. Systems A, B and C have been described in the text.

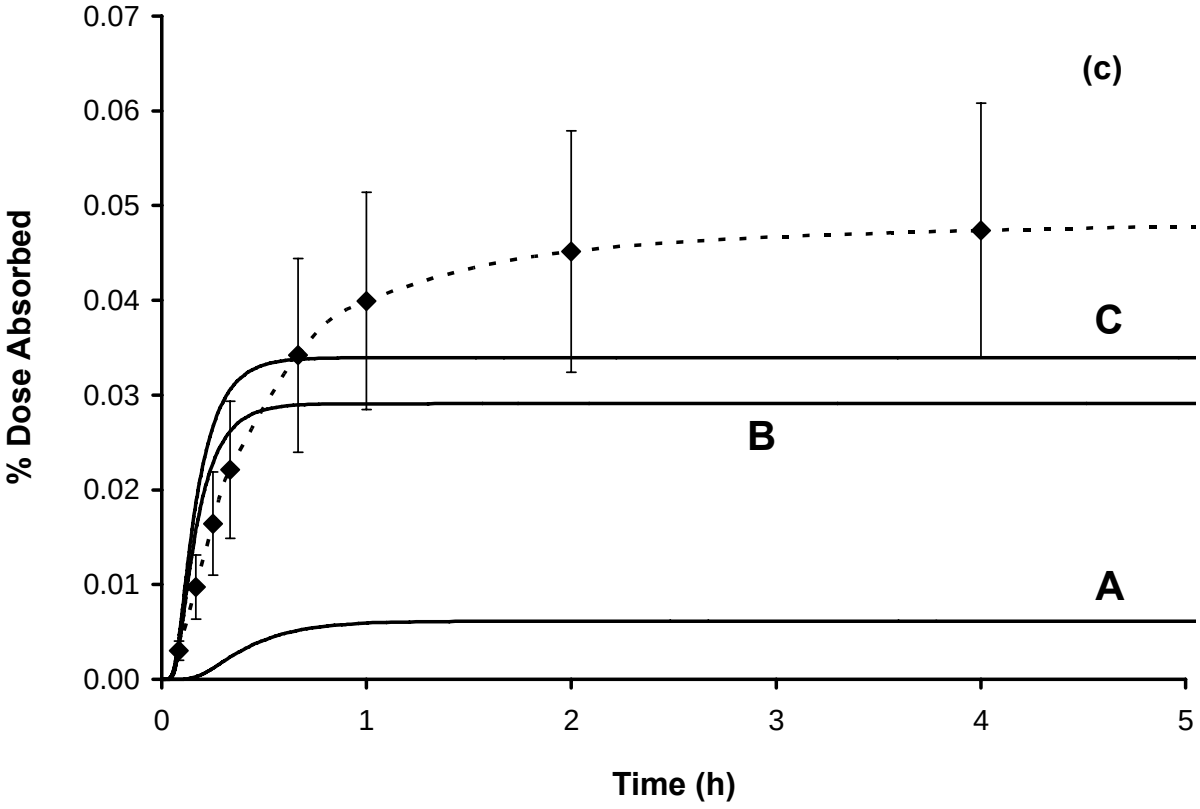


Fig. 5.1(c). Variation of fraction absorbed with time for a 20 μL dose of benzene. The line (---) is simply a guide to the eye. Systems A, B and C have been described in the text.

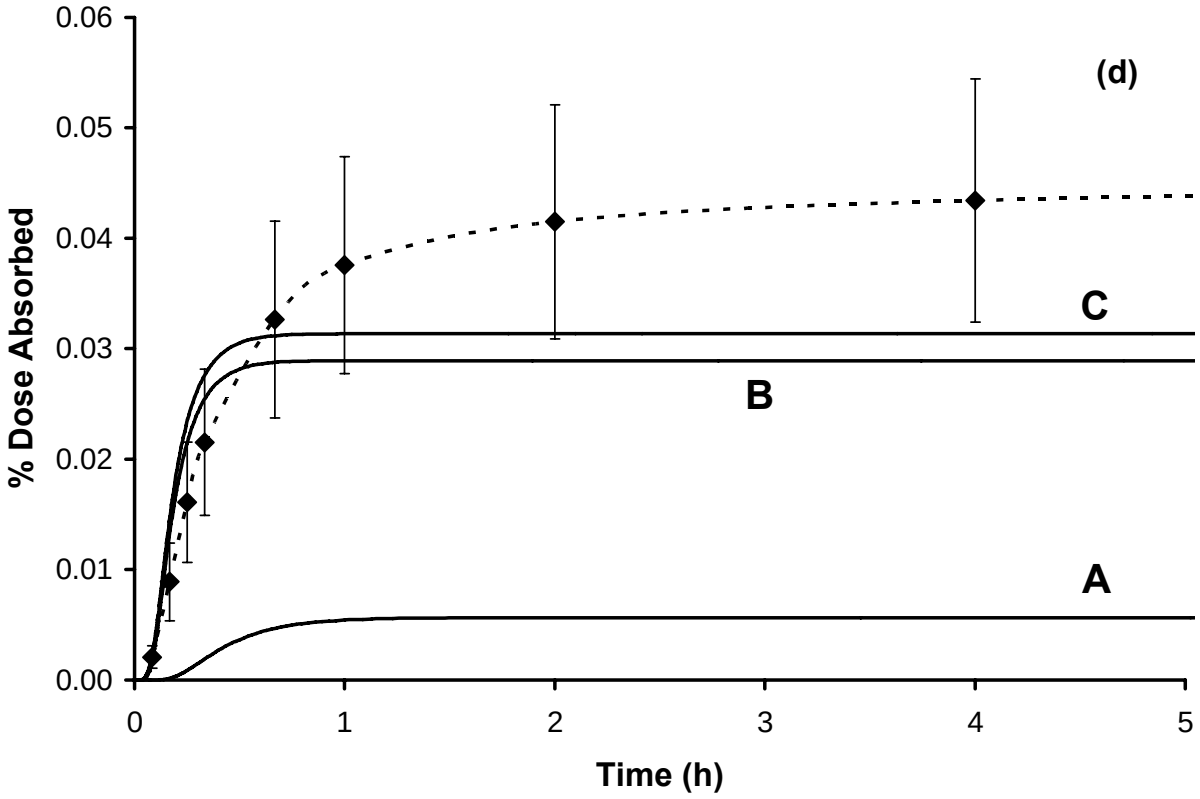


Fig. 5.1(d). Variation of fraction absorbed with time for a 5 μL dose of benzene. The line (---) is simply a guide to the eye. Systems A, B and C have been described in the text.

$F_{28,29}$ = 21/11 = 1.91). The optimum value of f_{dep} was 0.36 and that of D_{SC} was 4.979 × 10^{-9} cm^2/s or 11.7029 times the predicted value D_{WKN} (Table 4.2). The increase in D_{SC} versus D_{WKN} led to better estimations of the initial absorption rates, as well as total absorption (Fig. 5.3).

Further improvements in the model were obtained by relaxing the constraint that f_{dep} and D_{SC} be considered constant values. Optimization of the value of f_{dep} separately for each dose (5, 10, 20 and 40 μL) led to the results shown in Table 5.5 and Figure 5.2. A systematic increase in the value of f_{dep} with increasing dose was observed. Parameterization of this dependency according to the power law shown in Figure 5.3,

$$f_{dep} = 0.1786 \times \text{Dose}^{0.440} \tag{78}$$

led to Curves C in Figure 5.2 (χ_ν^2 = 3.477 × 10^{-5}). Thus the additional parameter represented by Eq. (78) lead to a greater than three-fold reduction in the squared residuals versus a constant f_{dep} model ($F_{27,28}$ = 10.91/3.477 = 3.138). The value of D_{SC} associated with this fit was 5.55 × 10^{-10} cm^2/s or 1.31 times D_{WKN}.

In all the analysis presented so far D_{SC} is a constant number. Similar to the ethanol analysis, it is assumed that D_{SC} is a function of the local concentration of benzene, which means it can vary between a minimum and maximum value. The minimum value is intrinsic SC permeability in absence of any sorbed benzene (and corresponds to the dilute solution limit) and while the maximum value corresponds to the saturation concentration of benzene. The mathematical function employed to describe this concentration dependent behavior of D_{SC} for benzene is identical to that employed for ethanol, with the exact same values for C_{inf} and *m*. In the current work, since the correlation used to predict D_{SC} (D_{WKN}) involves the use of a dilute-

solution limit correlation, it should naturally represent its lower limit or D_0. Once D_0, $C_{\inf}$ and m are determined, D_{SAT} is the only unknown variable that needs to be optimized. Thus, in an alternative regression procedure D_{SC} is given by Eq. (75) and D_{SAT} (instead of the constant *D*) is optimized. Now, along with D_{SAT}, D_0 can also be optimized (multiple nonlinear regression analysis) in order to test our hypothesis that it is in fact given by D_{WKN}. In each case, the deviation of the optimum value of each diffusivity parameter (D, D_0 or D_{SAT}) from its predetermined values will reflect the strength of the assumptions used in order to predict their values. At first, both D_0 and D_{SAT} were optimized simultaneously, while systematically varying f_{dep}. The results are given in Table 5.6. The optimization process generated optimal value of D_0, which was very close to the predictive value of D_{WNK} ($D_0 = 0.9675 D_{WKN}$). This verified our proposition that D_0 is given by D_{WNK} and for the rest of the simulations, D_0 was not optimized and held constant at D_{WKN}. D_{SAT} was subjected to optimization using the grid search algorithm while varying f_{dep} systematically. Table 5.6 lists the optimal parameters for the variable diffusivity regression analysis. Even though the optimal D_{SAT} was roughly 14 times greater than D_{WKN}, it is very close to the factor 11.7029 as obtained from the constant diffusivity regression analysis. Thus, both D_0 and D_{SAT} were fixed and f_{dep} was manually varied. The resulting optimal f_{dep} was 0.25, which is substantially different than the optimal f_{dep} of 0.36 as obtained from the constant diffusivity analysis. This is different than the behavior of ethanol where both the constant diffusivity and the variable diffusivity analyses predicted the same optimal f_{dep}. This behavior could be an indication of substantially higher solubility of benzene

Table 5.5. Optimal f_{dep} and D_{SC} when data for individual doses used for regression

Dose (µL)	f_{dep}	D_{FIT}/D_{WNK}	χ^2 (*10^5)
5	0.370	1.9713	4.8809
10	0.470	1.3477	3.0774
20	0.690	0.9004	2.6264
40	0.900	1.0042	1.7478

Table 5.6. Optimal f_{dep} and D_0 for variable-diffusivity regression

f_{dep}	$\frac{D_0}{D_{WKN}}$	$\frac{D_{SAT}}{D_{WKN}}$	χ^2 (*10^5)
0.200	1.0686 (2.0)	11.7029	6.5264
0.240	**0.9765 (2.0)**	**11.7029**	**4.7782**
0.300	1.0149 (2.0)	11.7029	5.0484
0.190	1.0	14.6884 (2.0)	3.1910
0.200	**1.0**	**14.4741 (2.0)**	**3.1644**
0.210	1.0	14.4196 (2.0)	3.1627
0.250	1.0	13.0922 (5.0)	3.5508
0.300	1.0	11.4769 (5.0)	5.0178
0.100	1.0	11.7029	13.6222
0.150	1.0	11.7029	10.5224
0.200	1.0	11.7029	6.5404
0.250	**1.0**	**11.7029**	**4.3663**
0.300	1.0	11.7029	5.0485
0.350	1.0	11.7029	10.8114

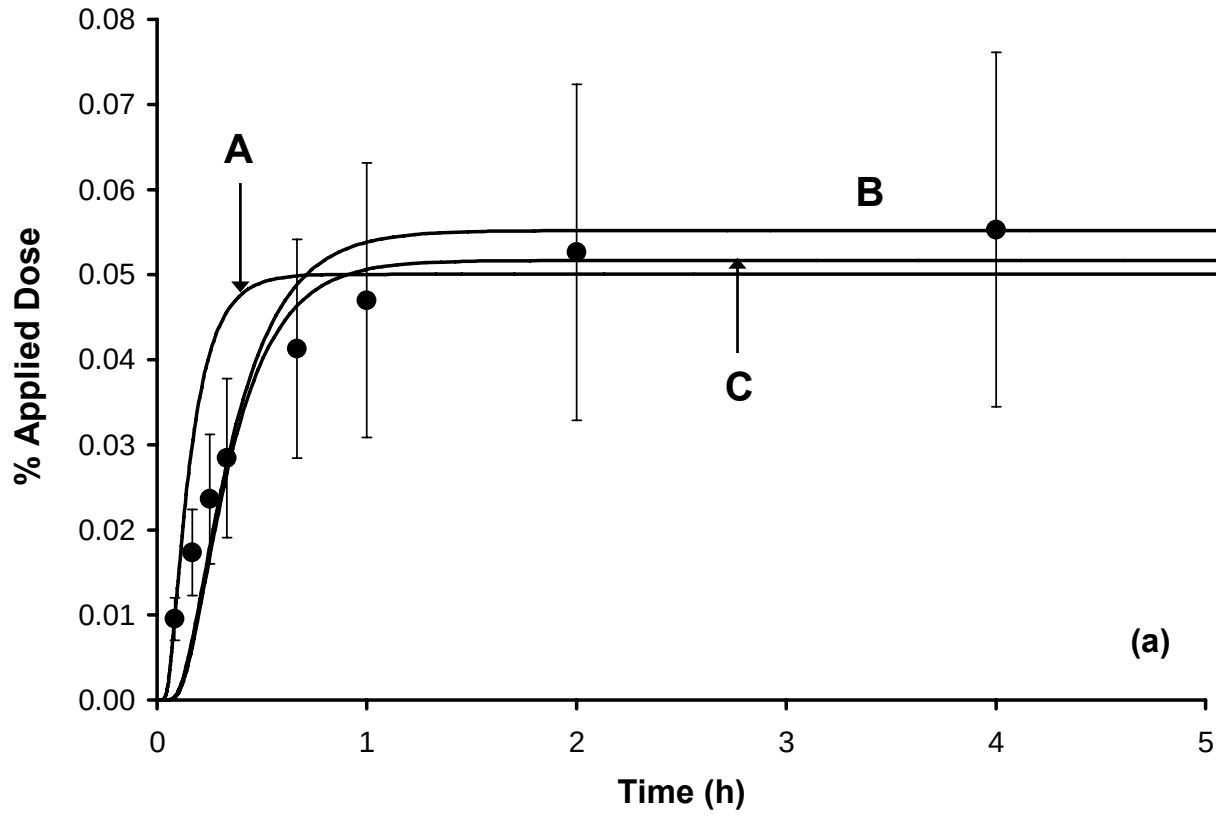


Fig. 5.2(a). Variation of fraction absorbed with time for a 5 μL dose of benzene. The line (---) is simply a guide to the eye. Systems A, B and C have been described in the text.

.

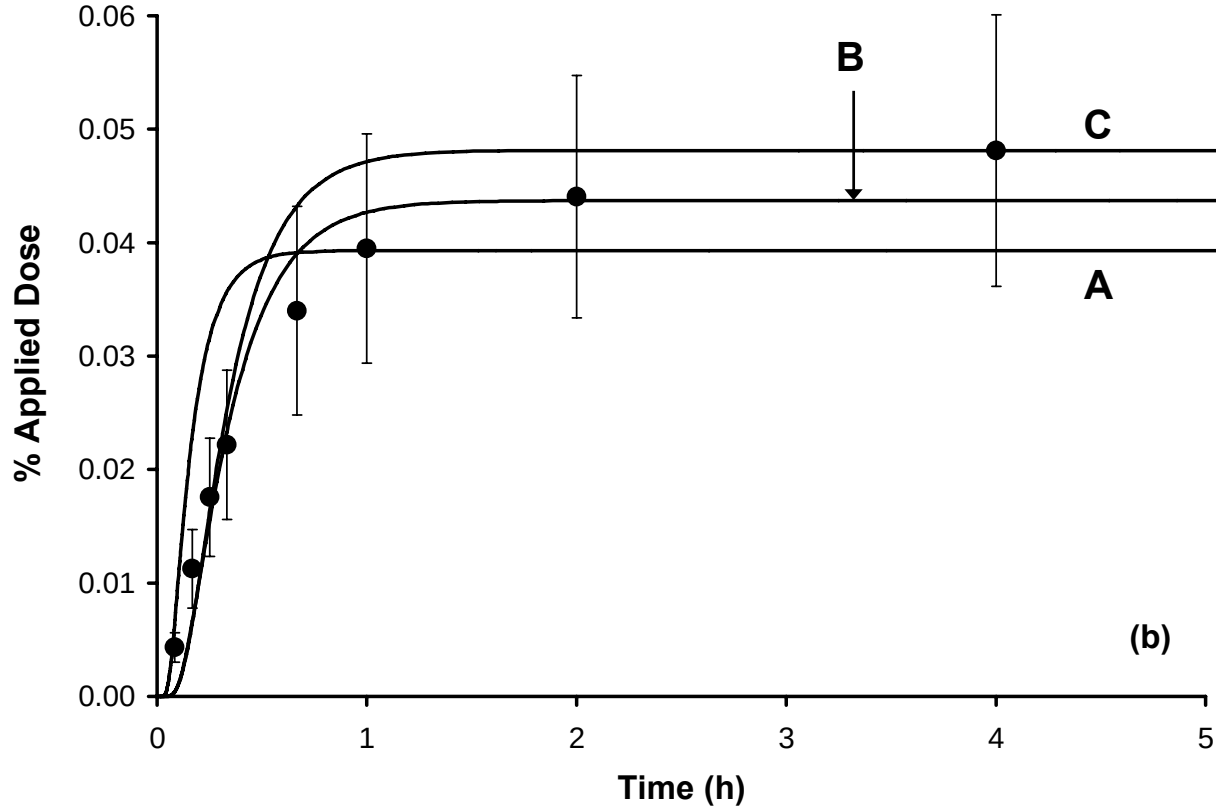


Fig. 5.2(b). Variation of fraction absorbed with time for a 10 μL dose of benzene. The line (---) is simply a guide to the eye. Systems A, B and C have been described in the text.

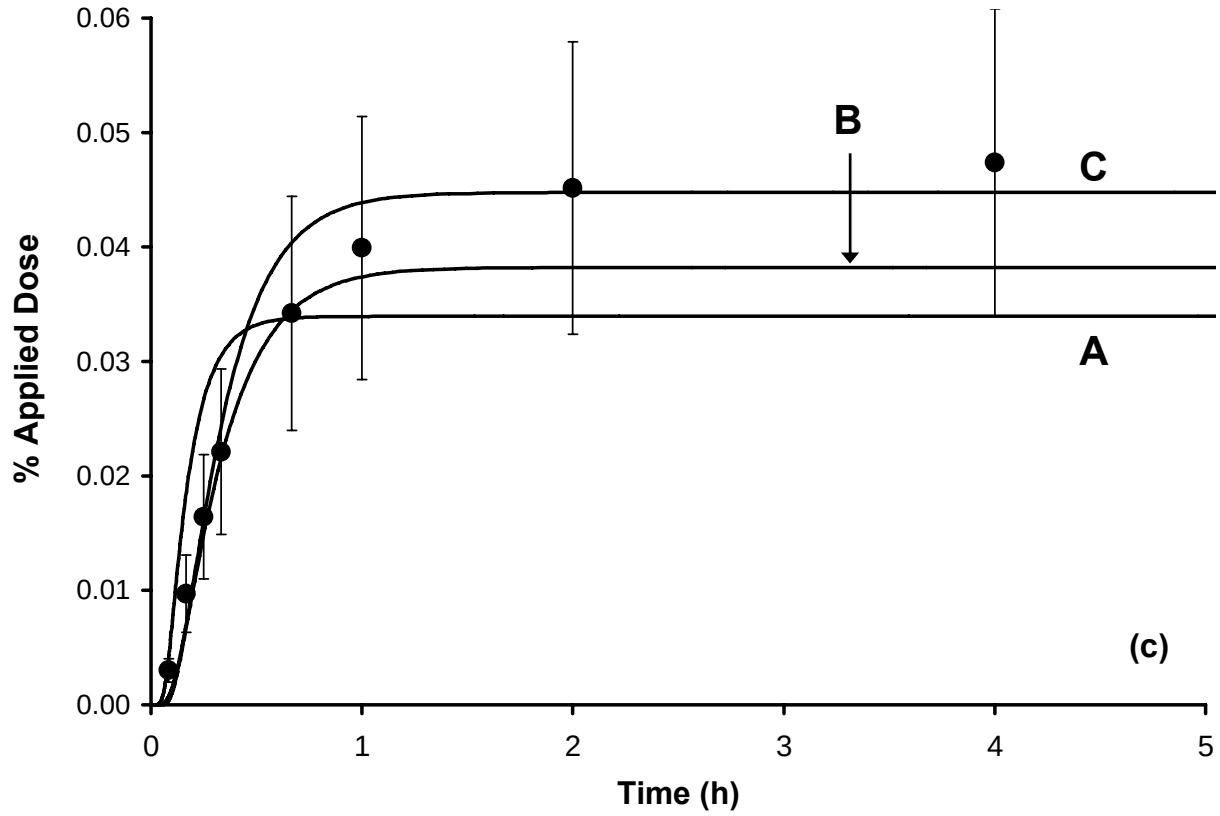


Fig. 5.2(c). Variation of fraction absorbed with time for a 20 μL dose of benzene. The line (---) is simply a guide to the eye. Systems A, B and C have been described in the text.

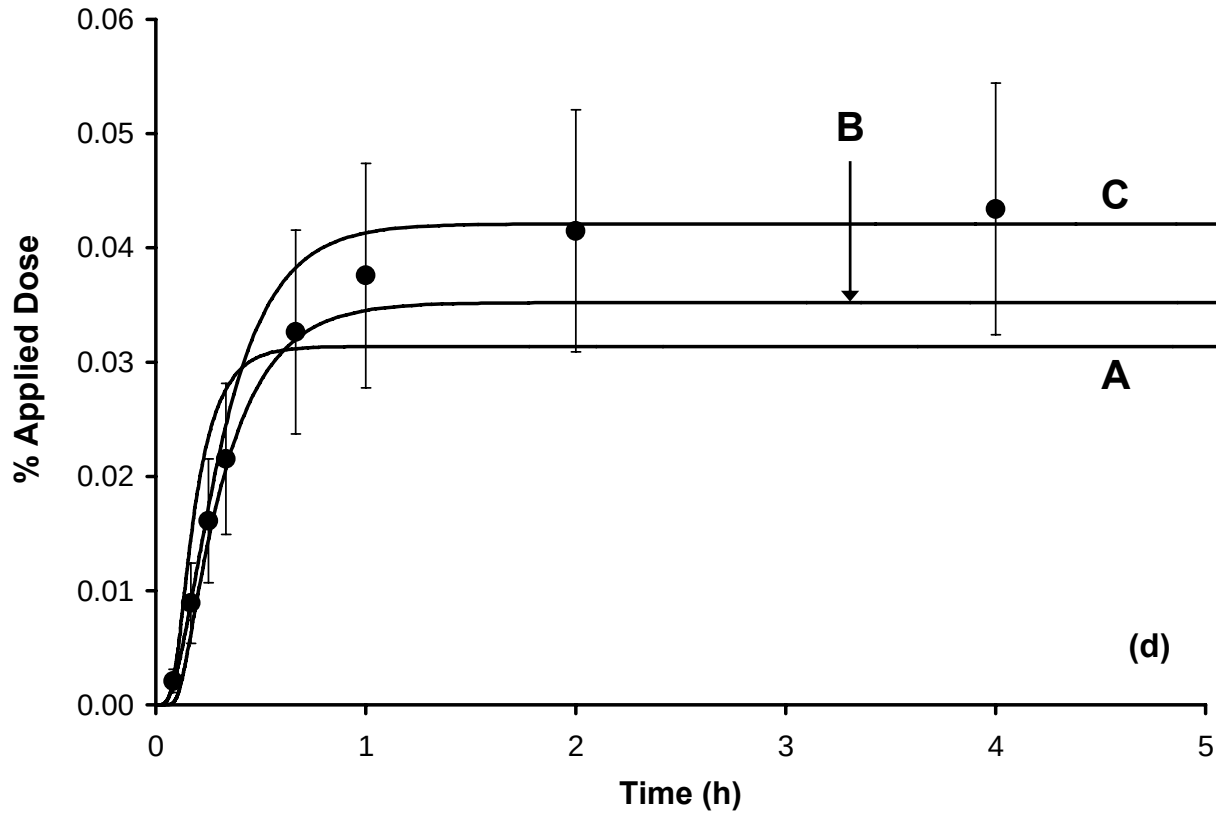


Fig. 5.2(d). Variation of fraction absorbed with time for a 40 μL dose of benzene. The line (---) is simply a guide to the eye. Systems A, B and C have been described in the text.

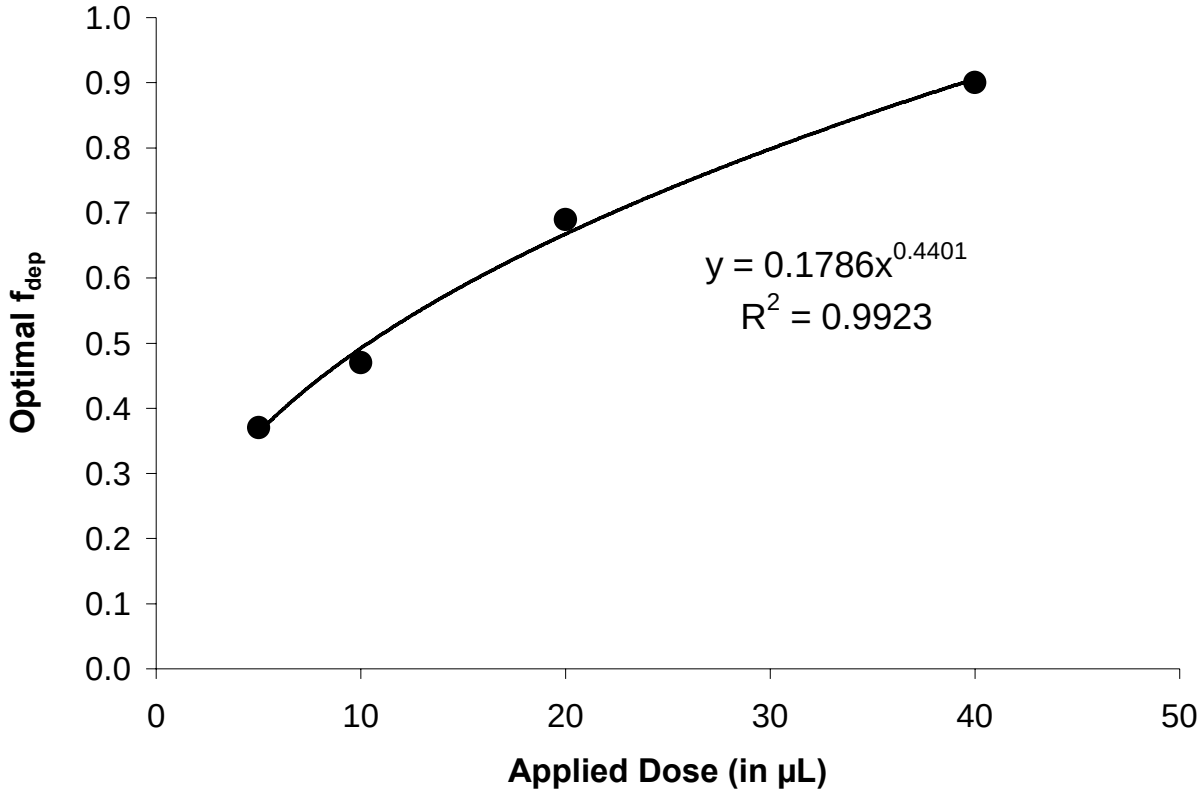


Fig. 5.3. Variation of optimal f_{dep} with varying applied dose.

in the SC. Both the initial absorption rates and the total absorption were well matched by the variable diffusivity and variable f_{dep} simulations. It should be noted that from a statistical viewpoint, this fit ($\chi_\nu^2 = 4.366 \times 10^{-5}$) was not significantly different from the variable f_{dep} model, ($\chi_\nu^2 = 3.477 \times 10^{-5}$)

The projections of the χ_ν^2 surface shown in Tables 5.5 and 5.7 allow statistical error bounds to be placed on the deposition depth parameter f_{dep}. The critical value of F with $\nu = 28$ and $p = 0.05$ is 1.87. Thus χ_ν^2 ratios exceeding 1.87 indicate a statistically poorer fit at $p = 0.05$. Using this criterion, the 95 % confidence limits on f_{dep} are $0.36^{+0.12}$ for the constant diffusivity model (Table 5.5). For the lower limit, the originally assumed value for $f_{dep} = 0.1$ is still within the given 95% confidence intervals. This result has been shown in Fig. 5.4.

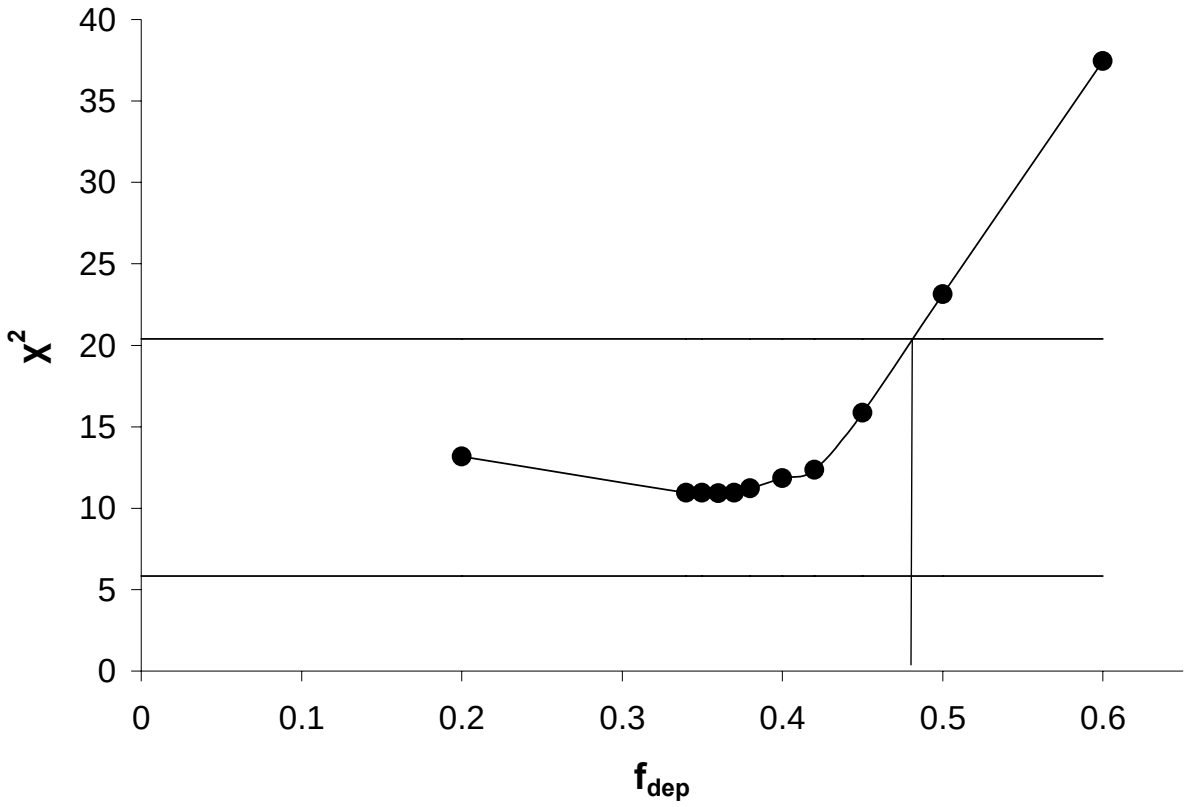


Fig. 5.4. Error surface and confidence intervals on χ_v^2 and f_{dep} for constant diffusivity regression analysis.

5.4 Conclusion

Percutaneous absorption of benzene following transient liquid phase exposures was measured experimentally. The experimental data were found to be consistent with an unsteady-state mass-transfer model using only two adjustable parameters, namely the fractional deposition depth and the diffusivity of benzene inside the SC. Extensive non-linear regression analyses were performed on several versions of the mass transfer model. Even though the use of constant deposition depth and SC diffusivity resulted in better correlations, substantial improvements resulted through the use of concentration-dependent diffusivity and dose-dependent deposition depths. Moreover, the disposition characteristics matched closely with another volatile compound ethanol, as analyzed in Chapter 4 of this dissertation. Benzene exhibited a very similar tendencies as ethanol in displaying a better fit using variable diffusivity and variable f_{dep}. Both the compounds exhibited similar power-law dependencies for optimal fractional deposition depth. Both of them showed a sharp increase in χ_v^2 for f_{dep} values larger than the minima. However, the optimal f_{dep} for benzene for both the individual dose and combined dose regression analyses are much higher than the corresponding optimal f_{dep} for ethanol. This may be indicative of higher SC solubility of benzene or a larger permeation enhancement. Thus, the results of benzene confirm with the analysis of ethanol as presented in Chapter 4 of this dissertation and ratify the mathematical model presented in Chapter 2. In, general the agreement of all three systems is supportive of an eventual predictive use of the model for volatile liquids in general.

References

Abramowitz, M., Stegun, I.A., (Eds.), 1972. Handbook of Mathematical Functions, National Bureau of Standards, Washington, D.C., p. 288.

Berner, B., Juang R.H., Mazzenga, G.C. (1989a). Ethanol and water sorption into stratum corneum and model systems. Journal of Pharmaceutical Sciences, 78, 472-476.

Bevington, P.R. (1969). Data Reduction and Error Analysis for the Physical Sciences. McGraw-Hill, New York.

Blank, I.H., MCAuliffe, D.J., (1985). Penetration of benzene through human skin. Journal of Investigative Dermatology. 85, 522-526.

Johnson, M.E., Blankschtein, D., Langer, R., (1997). Evaluation of solute permeation through the stratum corneum: lateral bilayer diffusion as the primary transport mechanism. Journal of Pharmaceutical Sciences, 86, 1162-1172.

Kasting, G.B., Miller, M.A., (2006). Kinetics of finite dose absorption through skin 2. Volatile compounds. Journal of Pharmaceutical Sciences, 95 (2), 268–280.

Khalil, E., Kretsos, K., Kasting, G.B., (2006). Glucose partition coefficient and diffusivity in the lower skin layers. Pharmaceutical Research, 23, 1227-1234.

Kretsos, K., Kasting, G.B., (2005). Dermal capillary clearance: physiology and modeling. Journal of Pharmacological and Biophysical Research, 18, 55-74.

Kretsos, K., Kasting, G.B., Nitsche, J.M., (2004). Distributed diffusion-clearance model for transient drug distribution within the skin. Journal of Pharmaceutical Sciences, 93, 2820-2835.

Maibach, H.I., Anjo, D.M., (1981). Percutaneous penetration of benzene and benzene contained in solvents used in the rubber industry. Arch. Env. Health. 36(5), 256-260.

Medlin, J., (2003). Benzene's adverse effects. Microarays reval breadth of toxicity. Environmental Health Perspectives, 111 (11), A590-A591.

Miller, M.A., Bhatt, V., Kasting, G.B., (2006). Dose and airflow dependence of benzyl alcohol disposition on skin. Journal of Pharmaceutical Sciences, 95 (2), 281–291.

Wang, T.-F. et al. (2006). A multiphase microscopic model for stratum corneum permeability. I. Formulation, solution and illustrative results for representative compounds. Journal of Pharmaceutical Sciences, 95, 620-648.

Yaws, C.L., (1999). Chemical Properties Handbook. McGraw-Hill, New York.

CHAPTER 6

BINARY MODEL DEVELOPMENT

6.1 Background

The mathematical model presented in Chapter 2 of this dissertation represents the disposition of a pure (volatile) liquid as it is applied onto skin. This model is general and can be potentially used to describe permeation of any pure liquid (volatile or non-volatile). Even though skin permeation of pure liquids is of sufficient interest pertaining to acute toxic and chronic occupational exposures, in reality, almost all transdermal drugs and topically applied pharmaceuticals are administered in the form of a mixture with solvents and permeation enhancers. The simplest example is that of a binary mixture containing the compound of interest (solute) and the medium (solvent). The skin permeation behavior of a particular penetrant can be markedly different if applied as a binary solution than in its pure form. This chapter aims to address this difference through the development of a comprehensive mathematical model describing transport of multiple (two) species through the skin-sublayers. This binary evaporation-absorption process is schematically similar to the pure component model shown in Fig. 2.1, except that the vehicle layer is now a binary mixture, henceforth also referred to as the donor solution (DS). This model is based upon the same mathematical principles and underlying considerations that were used to develop the pure component model in Chapter 2 of this dissertation. The process begins with the application of a known amount of binary solution (L_0) with known composition (ω_{A0}). Both the solute and the solvent begin to dissipate immediately from the skin surface through simultaneous evaporation and absorption. Unlike the pure component model, the concentration of solute and solvent in the liquid layer change with time. Moreover, due to differences in evaporation and absorption rates for the solute and the solvent, there will be concentration gradients for each component in the liquid layer. These features have been incorporated in the form of appropriate liquid-layer conservation-of-species and continuity

equations that were not necessary during the development of the pure component model. Similar to the pure component model, two skin sublayers, the stratum corneum (SC) and the viable tissue (VT) have been considered. The liquid layer is considered to be finite and its changing thickness has been represented through an appropriate coupled ordinary differential equation resulting out of an instantaneous material balance on the liquid layer. Similar to the pure component model, the binary model does not consider either swelling or shrinking; hence the skin layer thicknesses remain constant throughout the permeation process. Also, similar to the pure component model, the concept of the deposition layer has been considered (Kasting and Miller, 2006).

6.2 Development of the Model Equations

6.2.1 Model Differential Equations

The principal equation describing DS mass transport is the conservation–of–species equation and written as:

$$\frac{\partial \rho_A^{DS}}{\partial t} = -\frac{\partial n_A^{DS}}{\partial z} \tag{79}$$

Here ρ_A^{DS} represents the mass concentration of A in the DS, t is the time, z is the spatial coordinate and n_A^{DS} is the solute mass flux with stationary coordinates in the z direction. The solute mass flux can be expanded in terms of diffusive and convective flux as (Bird et al., 2001):

$$n_A^{DS} = j_A^{DS} + \rho_A^{DS} W^{DS} \tag{80}$$

where W^{DS} is the mass average velocity. Ideally, the mass-average velocity can be determined by solving the continuity equation along with the relevant conservation-of-species equations. However, this may not be possible in all situations because the continuity equation is a hyperbolic PDE and there is no global algorithm that is capable of achieving this for any

generalized situation. In order to circumvent this problem, only recently Lee et al. (2006) have developed a closed-form analytical expression for mass-average velocity, which can be readily applied to such systems involving mass transport in binary mixtures that have no volume-of-mixing. This analytical expression can be written as:

$$W^{DS} = -\left(\frac{1}{\rho_A^0} - \frac{1}{\rho_B^0}\right) j_A^{DS} + \left(n_T^{SC}\right)_{z=0} \tag{81}$$

where ρ_A^0 and ρ_B^0 are intrinsic densities of the compounds present in the binary solution, j_A^{DS} is the diffusive mass flux in DS and $\left(n_T^{SC}\right)_{z=0}$ is the total mass flux at the DS-SC interface. j_A^{DS} can be given by Fick's First Law as:

$$j_A^{DS} = -D_{AB} \frac{\partial \rho_A^{DS}}{\partial z} \tag{82}$$

where D_{AB} is the mutual binary diffusion coefficient of the solute in the solvent. Thus, Eqs. (79-82) describe the mass transfer characteristics of the DS.

Similar to the mass transport equation for the DS, the SC and VT conservation–of–species equations are written as:

$$\frac{\partial \rho_A^{SC}}{\partial t} = -\frac{\partial n_A^{SC}}{\partial z} \tag{83}$$

$$\frac{\partial \rho_B^{SC}}{\partial t} = -\frac{\partial n_B^{SC}}{\partial z} \tag{84}$$

$$\frac{\partial \rho_A^{VT}}{\partial t} = -\frac{\partial n_A^{VT}}{\partial z} \tag{85}$$

$$\frac{\partial \rho_B^{VT}}{\partial t} = -\frac{\partial n_B^{VT}}{\partial z} \tag{86}$$

where ρ_A^{SC}, ρ_B^{SC}, ρ_A^{VT}, ρ_B^{VT} are the mass concentrations, n_A^{SC}, n_B^{SC}, n_A^{VT}, n_B^{VT} are the total mass flux for components A and B, respectively in SC and VT. Similar to the pure component model, it is assumed that absorption of solute and solvent into the SC and VT is not appreciable in order to alter layer dimensions. Thus, the thickness and density of the skin sublayers are assumed to be constant. The expression for mass flux within a stationary phase (membrane) is given by Bird et al. (2001) and has been used in the development of the pure component model. This can be extended for a ternary system and written as:

$$n_A^{SC} = \frac{1}{1-\omega_A^{SC}-\omega_B^{SC}}\left[j_A^{SC}\left(1-\omega_B^{SC}\right)+ j_B^{SC}\omega_A^{SC}\right] \tag{87}$$

$$n_B^{SC} = \frac{1}{1-\omega_A^{SC}-\omega_B^{SC}}\left[j_B^{SC}\left(1-\omega_A^{SC}\right)+ j_A^{SC}\omega_B^{SC}\right] \tag{88}$$

$$n_A^{VT} = \frac{1}{1-\omega_A^{VT}-\omega_B^{VT}}\left[j_A^{VT}\left(1-\omega_B^{VT}\right)+ j_B^{VT}\omega_A^{VT}\right] \tag{89}$$

$$n_B^{VT} = \frac{1}{1-\omega_A^{VT}-\omega_B^{VT}}\left[j_B^{VT}\left(1-\omega_A^{VT}\right)+ j_A^{VT}\omega_B^{VT}\right] \tag{90}$$

where j_A^{SC}, j_B^{SC}, j_A^{VT}, j_B^{VT} are the diffusive fluxes for components A and B respectively in SC and VT layers and is given by the constant–density version of Fick's 1st Law of Diffusion as:

$$j_A^{SC} = -\rho_{SC}^0 D_{ASC}\frac{\partial \omega_A^{SC}}{\partial z} \tag{91}$$

$$j_B^{SC} = -\rho_{SC}^0 D_{BSC}\frac{\partial \omega_B^{SC}}{\partial z} \tag{92}$$

$$j_A^{VT} = -\rho_{VT}^0 D_{AVT}\frac{\partial \omega_A^{VT}}{\partial z} \tag{93}$$

$$j_B^{VT} = -\rho_{VT}^0 D_{BVT}\frac{\partial \omega_B^{VT}}{\partial z} \tag{94}$$

where ρ_{SC}^{0}, ρ_{VT}^{0} is the density of SC and VT and D_{ASC}, D_{BSC} and D_{AVT}, D_{BVT} are the diffusivities of A and B in the SC and VT, respectively. Combining Eqs. (5-16), we obtain final expressions for conservation–of–species equations for both components inside the SC and VT can be written as:

$$\frac{\partial \omega_A^{SC}}{\partial t} = \frac{\partial}{\partial z}\left[\frac{1}{1-\omega_A^{SC}-\omega_B^{SC}}\left\{D_{ASC}\left(1-\omega_B^{SC}\right)\frac{\partial \omega_A^{SC}}{\partial z} + D_{B-SC}\omega_A^{SC}\frac{\partial \omega_B^{SC}}{\partial z}\right\}\right] \tag{95}$$

$$\frac{\partial \omega_B^{SC}}{\partial t} = \frac{\partial}{\partial z}\left[\frac{1}{1-\omega_A^{SC}-\omega_B^{SC}}\left\{D_{BSC}\left(1-\omega_A^{SC}\right)\frac{\partial \omega_B^{SC}}{\partial z} + D_{A-SC}\omega_B^{SC}\frac{\partial \omega_A^{SC}}{\partial z}\right\}\right] \tag{96}$$

$$\frac{\partial \omega_A^{VT}}{\partial t} = \frac{\partial}{\partial z}\left[\frac{1}{1-\omega_A^{VT}-\omega_B^{VT}}\left\{D_{AVT}\left(1-\omega_B^{VT}\right)\frac{\partial \omega_A^{VT}}{\partial z} + D_{BVT}\omega_A^{VT}\frac{\partial \omega_B^{VT}}{\partial z}\right\}\right] \tag{97}$$

$$\frac{\partial \omega_B^{VT}}{\partial t} = \frac{\partial}{\partial z}\left[\frac{1}{1-\omega_A^{VT}-\omega_B^{VT}}\left\{D_{BVT}\left(1-\omega_A^{VT}\right)\frac{\partial \omega_B^{VT}}{\partial z} + D_{AVT}\omega_B^{VT}\frac{\partial \omega_A^{VT}}{\partial z}\right\}\right] \tag{98}$$

The conservation-of-species equations for each component in each layer require an initial condition and two boundary conditions.

6.2.2 Initial Conditions

The initial conditions are given by the known initial concentration of the components in these layers. For the DS, it is given by:

$$\omega_A^{DS} = \omega_{A0}^{DS} \text{ at } t = 0 \tag{99}$$

For the VT, this is equal to zero:

$$\omega_A^{VT}\Big|_{z,0} = \omega_B^{VT}\Big|_{z,0} = 0 \tag{100}$$

For SC, the initial condition is a square-wave function depicting the presence of the permeant deposited region. It is written as:

$$\omega_A^{SC}\Big|_{z,0} = \omega_B^{SC}\Big|_{z,0} = 0 \text{ for } 0 \le z < \left(1 - f_{dep}\right) L_{SC} \tag{101}$$

$$\left.\begin{aligned} \omega_A^{SC}\Big|_{z,0} &= \omega_{ASC0} \\ \omega_B^{SC}\Big|_{z,0} &= \omega_{BSC0} \end{aligned}\right] \text{ for } \left(1 - f_{dep}\right) L_{SC} \le z \le L_{SC} \tag{102}$$

The initial conditions in the deposition layer of SC will depend on the initial composition of the DS. If the concentration of either the solute or the solvent is high, such that it corresponds to a value greater than its saturation concentration, then its value will be equal to its saturation concentration in the SC. However, if its value in the DS is low, then the initial condition in the deposition layer of SC will be the product of its DS concentration and the corresponding SC-DS partition coefficient. These can be expressed in terms of the following algorithms:

If ($\omega_{ASC0} \le \omega_{Asat}$) *then*

$$\omega_{ASC0} = \frac{K_{SC/DS}^{A} \rho_{DS}^{0} \omega_{A0}}{\rho_{SC}^{0}} \tag{103}$$

else

$$\omega_{ASC0} = \omega_{Asat} \tag{104}$$

end if

and:

If ($\omega_{BSC0} \le \omega_{Bsat}$) *then*

$$\omega_{BSC0} = \frac{K_{SC/DS}^{B} \rho_{DS}^{0} \left(1 - \omega_{A0}\right)}{\rho_{SC}^{0}} \tag{105}$$

else

$$\omega_{BSC0} = \omega_{Bsat} \tag{106}$$

end if

6.2.3 Boundary conditions

The boundary condition at the DS – air interface, $L(t)$ cannot be obtained by the simple continuity of flux condition, as is the case with stationary boundaries. Since $z = L(t)$ is a moving boundary, the required boundary condition is obtained by an instantaneous integral material balance on the species and overall mass in the DS. Combination of these balance equations yields:

$$\left(j_A^{DS}\right)_{z=L} = \left(n_A^G - \omega_A^{DS} n_T^G\right)_{z=L} \tag{107}$$

where n_A^G is the gas–phase total mass flux of species A and n_T^G is gas–phase total mass flux. Similar to the pure component model the gas-phase mass transfer flux can be expressed in terms of evaporation mass transfer coefficients.

The solvent mass–flux can be expressed in terms of mass-transfer coefficient as follows:

$$n_A^G = k_A^{evap}\left(\rho_A^{DS}\right)_{z=L} \tag{108}$$

$$n_B^G = k_B^{evap}\left(\rho_B^{DS}\right)_{z=L} \tag{109}$$

In order to complete the description of the boundary condition at $z = L$, it is necessary to have an auxiliary condition, defining the location of the DS – air interface. This is obtained by the integral mass balance of total solution mass as:

$$\frac{dL}{dt} - W = -\left(\frac{n_T^G}{\rho^{DS}}\right)_{z=L} \tag{110}$$

where the initial condition for the above ODE is at $t = 0,\ L(t) = L_0$. Detailed derivation of Eqs. (29) and (32) are given in Appendix C.

The BC at the DS-SC interface is also complex. Similar to initial condition explained above, it will depend on the presence or absence of the DS and also on the saturation

concentration of the solute and the solvent. These can be expressed according to the following algorithm:

if ($0 \le t \le t_{depl}$) *then*

if ($\omega_A^{SC}\big|_{z,0} \le \omega_{Asat}$) *then*

$$\omega_A^{SC}\Big|_{z,0} = \mathrm{K}_{SC/DS}^{A}\left(\frac{\rho^{DS}}{\rho_{SC}^{0}}\right)\omega_A^{DS}\Big|_{z,0} \tag{111}$$

else

$$\omega_A^{SC}\Big|_{z,0} = \omega_{Asat} \tag{112}$$

end if

else

$$n_A^{SC}\Big|_{L_{SC},t} = k_A^{evap}\rho^{SC}\omega_A^{SC} \tag{113}$$

end if

Similarly for component B, we have:

if ($0 \le t \le t_{depl}$) *then*

if ($\omega_B^{SC}\big|_{z,0} \le \omega_{Bsat}$) *then*

$$\omega_B^{SC}\Big|_{z,0} = \mathrm{K}_{SC/DS}^{B}\left(\frac{\rho^{DS}}{\rho_{SC}^{0}}\right)\left(1-\omega_A^{DS}\right)\Big|_{z,0} \tag{114}$$

else

$$\omega_B^{SC}\Big|_{z,0} = \omega_{Bsat} \tag{115}$$

end if

else

$$n_B^{SC}\Big|_{L_{SC},t} = k_B^{evap}\rho^{SC}\omega_B^{SC} \tag{116}$$

end if

The boundary conditions at the SC-VT interface remain the same at all times and are given by a partition relation and continuity of flux.

$$\omega_A^{SC}\Big|_{0,t} = \frac{\rho^{VT}}{K_{VT/SC}^{A}\rho^{SC}}\omega_A^{VT}\Big|_{0,t} \tag{117}$$

$$\omega_B^{SC}\Big|_{0,t} = \frac{\rho^{VT}}{K_{VT/SC}^{B}\rho^{SC}}\omega_B^{VT}\Big|_{0,t} \tag{118}$$

$$n_A^{VT}\Big|_{0,t} = n_A^{SC}\Big|_{0,t} \tag{119}$$

$$n_B^{VT}\Big|_{0,t} = n_B^{SC}\Big|_{0,t} \tag{120}$$

Similar to the pure component model, the bottom of the VT has a sink condition for both solute and solvent and is written as:

$$\omega_A^{VT}\Big|_{-L_{VT},t} = \omega_B^{VT}\Big|_{-L_{VT},t} = 0 \tag{121}$$

6.3 Methodology

6.3.1 Numerical Solution Methodology

Similar to the pure component model discussed in Chapter 2, Eqs. (1-43) were solved numerically using a finite difference/finite element based subroutine D03PPF from the NAG® (Numerical Algorithms Group) mathematical library (NAG®, 2006). The multilayer system of PDEs is converted to a monolayer system of PDEs through the same coordinate transformation discussed in Appendix A.

6.3.2 Experimental Methodology

A detailed validation of the proposed model requires experimental data on skin permeation of binary solutions. Benzyl alcohol (BA) has been chosen as the representative ingredient in a solution with ethanol. Such systems are of direct relevance to cosmetic and fragrance industries (Saiyasombati and Kasting, 2003; Miller et al, 2006). Saiyasombati and Kasting (2003) studied skin disposition of BA (1% solution in ethanol) as they measured evaporation and absorption rates of BA on human cadaver skin from a Franz diffusion cell modified with a vapor trap. The experimental parameter varied was the airflow rate, which was found to have an appreciable effect on the evaporative mass transfer coefficient and hence affected both the evaporation and (as a result) the absorption rates. The data-acquisition technique, materials used and experimental methodology have been discussed in detail in Saiyasombati and Kasting (2003) along with presentation of evaporation and absorption time profiles. Although, BA-ethanol has been chosen to be the representative system, the diffusion model can be applied to any binary solution in its liquid state. In this study, the experimental data on BA and ethanol permeation through human skin in vitro from Saiyasombati and Kasting (2003) have been analyzed using the mathematical model developed in this chapter. The additional dose-dependence study discussed in Miller et al. (2006) was not analyzed.

6.3.3 Estimation of Transport and Thermodynamic Parameters

Similar to ethanol and benzene (Chapters 4 and 5 of this dissertation), application of the transport model to BA requires as inputs a number of vehicle and tissue dimensions, physicochemical properties of the permeants and their diffusivities and partition coefficients in

Table 6.1. Physicochemical properties of BA for skin disposition estimation

Property	Symbol	Units	Value	Ref.
Molecular weight	MW	g/mol	108.1	Yaws (1999)
Density	ρ	g/cm^3	1.035	Yaws (1999)
Vapor pressure	P_{vp}	torr	0.1511	Yaws (1999)
Log (octanol/water partititon coefficient	log K_{oct}		1.100	Yaws (1999)
Water solubility	S_w	g/L	4.47×10^{-2}	Yaws (1999)
Fraction unbound to albumin	f_u			Yamazaki & Kanaoka (2004)
Fraction nonionized	f_{non}		1.0	Martin (1993)

Table 6.2. Thermodynamic and transport properties for estimation of BA skin disposition

Property	Units	Value	Correlation	Reference
			Donor Solution (DS)	
u	m/s	0.7368	Non-linear regression	Chapters 4 & 5
k_{evap}	cm/h	0.0007	$k_g = 6320\,u^{0.78}/MW^{1/3}$ $k_{evap}\rho = k_g P_{vp} MW/\left(7.6\times10^{-5} RT\right)$	Peress (2003), N-Dri-Stempfer & Bunge (2005), Kasting & Miller (2006)
			Stratum Corneum (SC)	
L_{SC}	μm	13.40		Johnson et al. (1996), Wang et al. (2003)
ρ_{SC}	g/cm^3	1.194	$\rho_{SC} = (1+v)/\left(\omega_{lip}/\rho_{lip} + \omega_{pro}/\rho_{pro} + v/\rho_w\right)$	Nitsche et al. (2006)
D_{SC}	cm^2/s	5.242×10^{-11}	WKN correlation	Wang et al. (2007)
$K_{SC/DS}$		0.3761	WKN correlation	Nitsche et al. (2006)
			Viable Tissue (VT)	
L_{VT}	μm	500.0		Khalil et al. (2006)
ρ_{VT}	g/cm^3	1.019	$\rho_{VT}\left(g/cm^3\right) = \left(\omega_{H_2O} + 0.649\omega_{pro} + 1.227\omega_{lip}\right)^{-1}$	Anderson et al. (2000)
D_{VT}	cm^2/s	1.80×10^{-6}	$\log D_{free} = -4.15 - 0.655\log MW$ $D_{VT} = D_{free}/\left(0.68 + 0.32/f_u + 0.001\,f_{non}K_{oct}\right)$	unpublished result
$K_{VT/SC}$		0.1264		Cross et al. (2003)

the two skin layers, designated SC (stratum corneum) and VT (viable tissues). Similar to the analysis of ethanol and benzene, the viable epidermis and residual dermis in the split-thickness skin samples are treated as a single layer with the properties of unperfused dermis (Kretsos and Kasting, 2005; Kretsos et al., 2006. Relevant physicochemical properties for BA in skin are readily obtained from the literature and are listed in Table 6.1. These properties were used as input parameters for the previously developed correlations for transport parameters and partition coefficients summarized in Table 6.2. Parameters for which the present analysis departs from the literature values are discussed below. The physicochemical properties of ethanol have already been listed in Tables 4.1 and 4.2 and discussed in Chapter 4 of this thesis. The ethanol parameters have been kept unaltered and used in the present analysis.

6.3.3.1 Ethanol diffusivity in stratum corneum (D_{SC})

Constant diffusivity regression analysis of the ethanol data in Chapter 4 resulted in a revised value of optimal SC diffusivity for ethanol, which was 0.5172 times the predicted correlation ($D_{SC} = 0.5172 D_{WKN} = 4.805 \times 10^{-10}$ cm^2/s). This value is used in the analysis of the current BA-ethanol system.

6.3.3.2 Fractional deposition depth in SC (f_{dep})

Similar to previous studies from our group and the analysis of ethanol and benzene, this parameter was introduced. A value of 0.1 was proposed by (Kasting and Miller, 2006) and supported experimentally for benzyl alcohol (Miller et al., 2006) and DEET (Santhanam et al., 2005). It was clear from the analysis of ethanol and benzene that a higher value of f_{dep} is necessary to appropriately describe the disposition characteristics of small molecular weight

volatile solvents in small doses. In the present study, two different values of f_{dep} have been considered in order to observe the behavior of the system at these values. Unlike the analysis of ethanol and benzene, detailed variation of f_{dep} along with determining optimal parameters through regression analysis of ethanol have not been performed.

6.3.3.3 Airflow Correction Factor (ACF)

The experimental study was airflow dependent and several values of ambient airflow were used to acquire the experimental data on BA evaporation and absorption. The default value of k_{evap} is given in Table 6.2 and seven different airflow correction factors (ACF) were employed. These are 10/42, 20/42, 30/42, 40/42, 50/42, 65/42 and 80/42. Of these, results corresponding to 4 different values of correction factors (10/42, 20/42, 40/42, 80/42) have been reported in this section. It should be noted that this range covers airflow values are both less than and greater than the default airflow conditions that were used for the constant-airflow ethanol and benzene analysis.

6.3.4 Regression analysis

The model parameters given in section 6.3.3 were used without modification to calculate the time course of absorption and evaporation of BA from skin according to the described model with one exception: D_{BSC}. This parameter was varied to provide the best fits of the model to the experimental data. It was selected because it was only free parameter that could play a key role in determining the rate and extent of BA absorption. Also, in order to test the appropriateness of the airflow velocity and the correlation used for determining the evaporation rate, the parameter k_{evap} was optimized. Similar to the analysis of ethanol and benzene, values of D_{SC} was

determined by nonlinear regression by minimizing the normalized sum of squared residuals χ_v^2, defined by Bevington (1969).

$$\chi_v^2 = \frac{1}{n-p}\sum_{i=1}^{n} w_i \left[y_i^{obs} - y_i^{fit} \right]^2 \tag{122}$$

The model fitting procedure entailed repetitively running the numerical model with varying values of the adjustable parameters D_{SC} for each airflow correction factor and calculating the residuals for each trial according to Eq. (122). Parameters were optimized by carefully integrating Bevington's (1969) grid search (GRID SEARCH) method with the chosen PDE-solving numerical algorithm of NAG®.

6.4 Representative Results

Selected results and simulated absorption curves are shown in Figures 6.1-6.4. The initial condition is given by a DS mass fraction of BA = 0.0128 and an initial DS thickness of 125.8 μm. The experimental data show a smooth trend of increasing evaporation and decreasing absorption with increasing airflow correction factor (ACF) or k_{evap}. Mean cumulative evaporation of radioactivity ranged from 44.49% of the applied dose for an ACF of 10/42 to 83.31% of applied dose for an ACF of 80/42. Similarly, absorption of radioactivity showed the opposite trend and ranged from 48.24% of applied dose for an ACF of 10/42 to 10.7 % of the applied dose for an ACF of 80/42. Both evaporation and absorption data for all 7 ACFs are used simultaneously for regression analysis. The resulting optimal parameters suggested that the value of airflow velocity and k_{evap} are appropriate and it was left out of subsequent regression analysis. The optimal value of D_{BSC} came out to be 5.41 times D_{WKN} for an f_{dep} = 0.1 and 5.16 times D_{WKN} for an f_{dep} = 0.18. This may be an indication of permeation enhancement in the SC owing

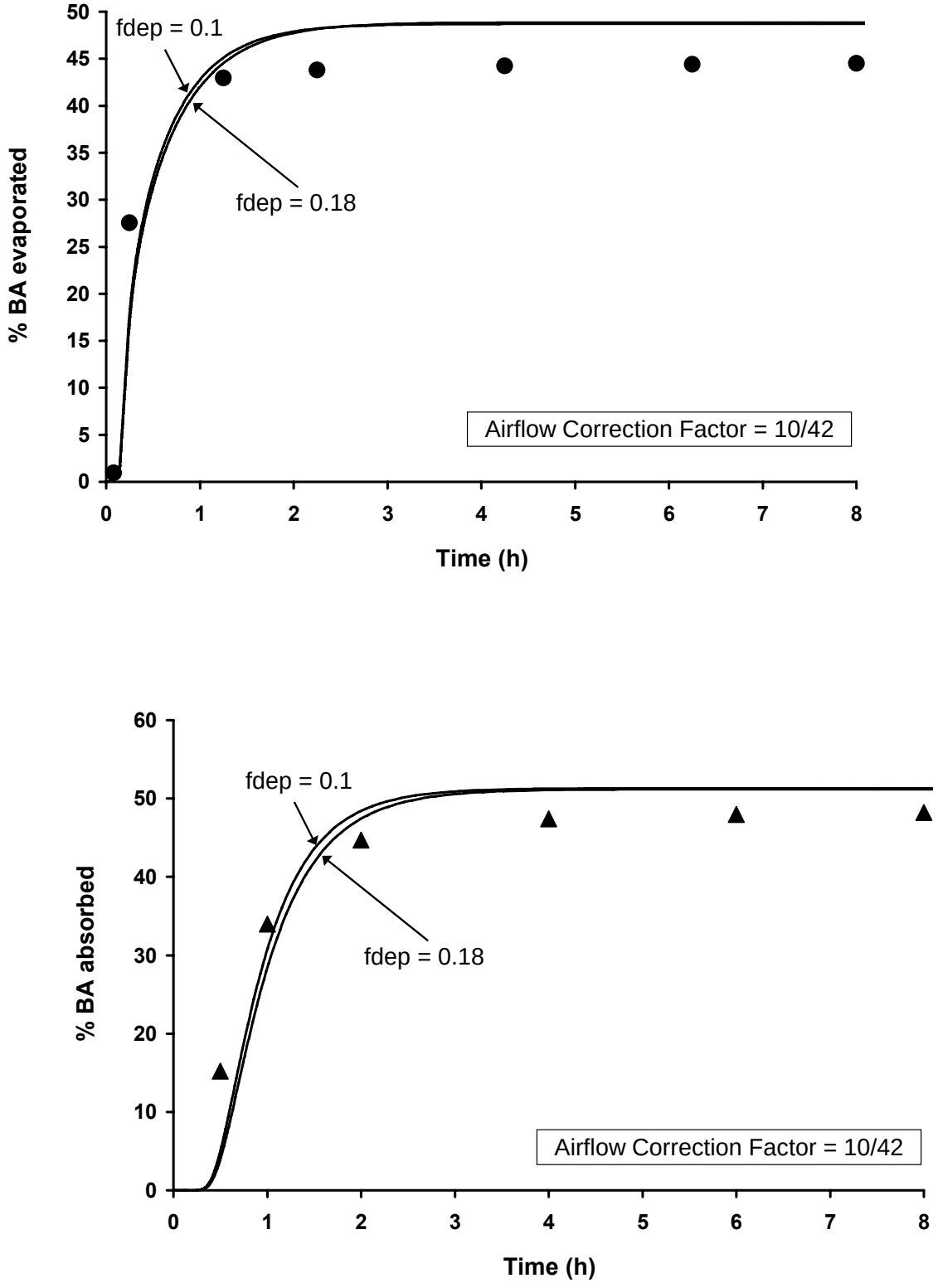


Fig. 6.1. Variation of fraction BA evaporated and absorbed with time: ACF = 10/42.

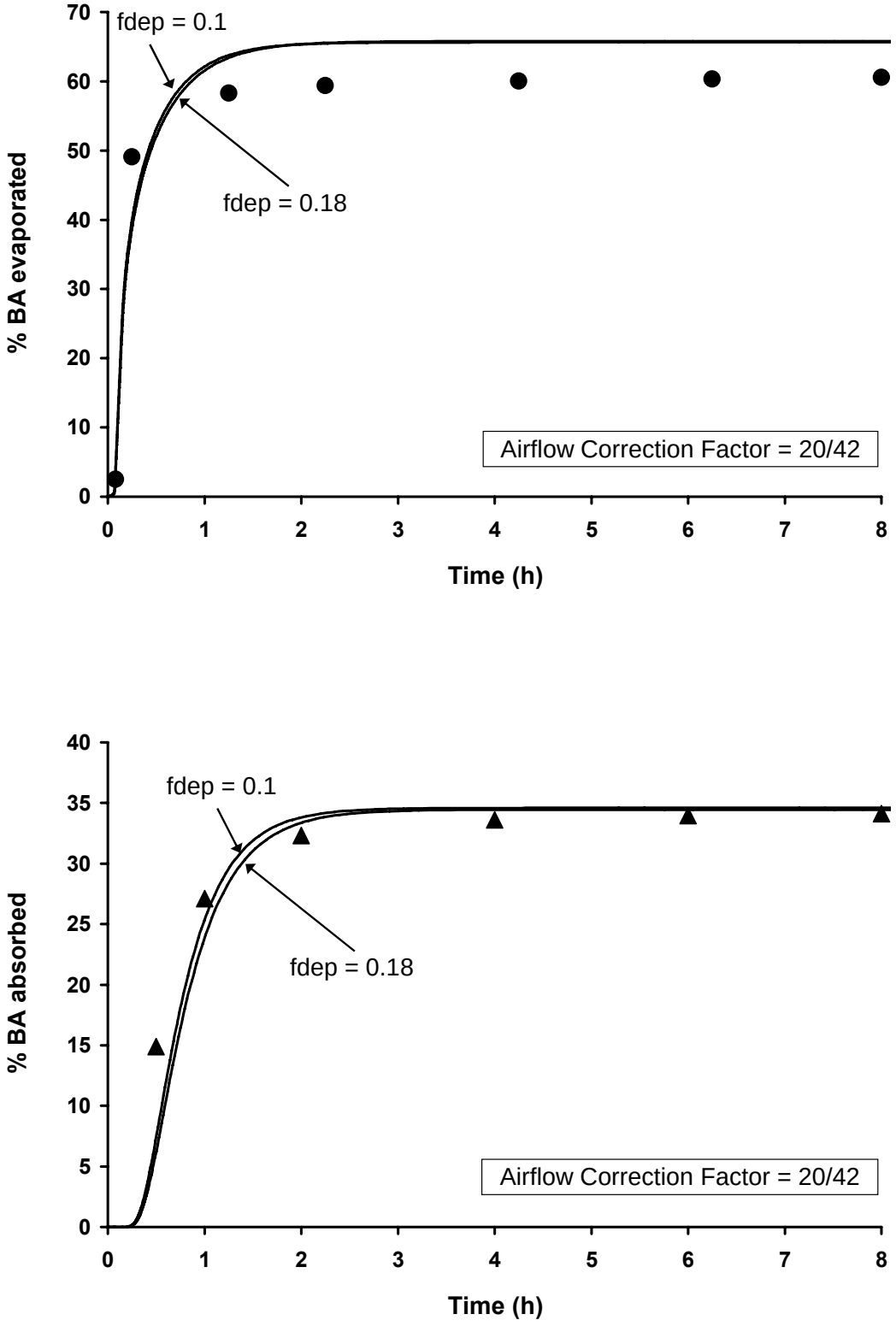
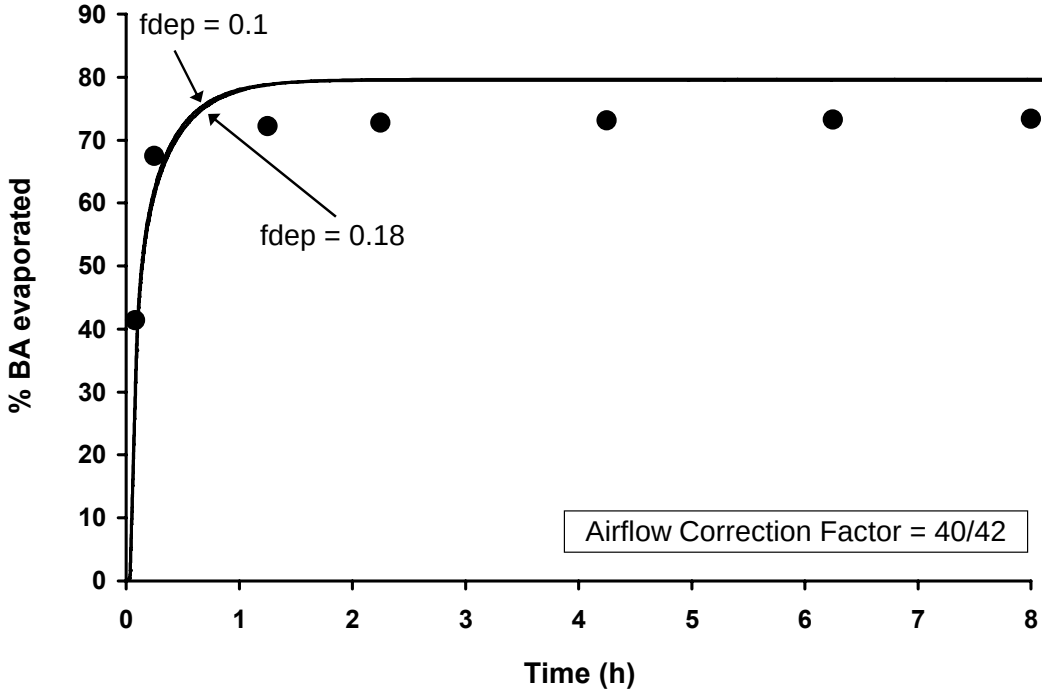


Fig. 6.2. Variation of fraction BA evaporated and absorbed with time: ACF = 20/42.



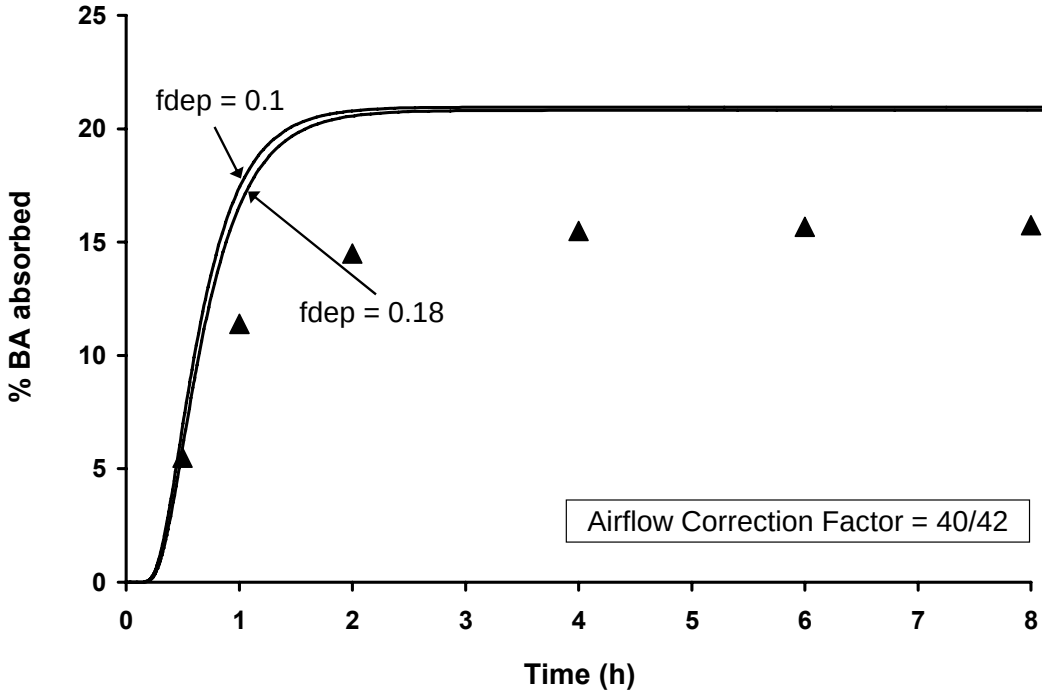


Fig. 6.3. Variation of fraction BA evaporated and absorbed with time: ACF = 40/42.

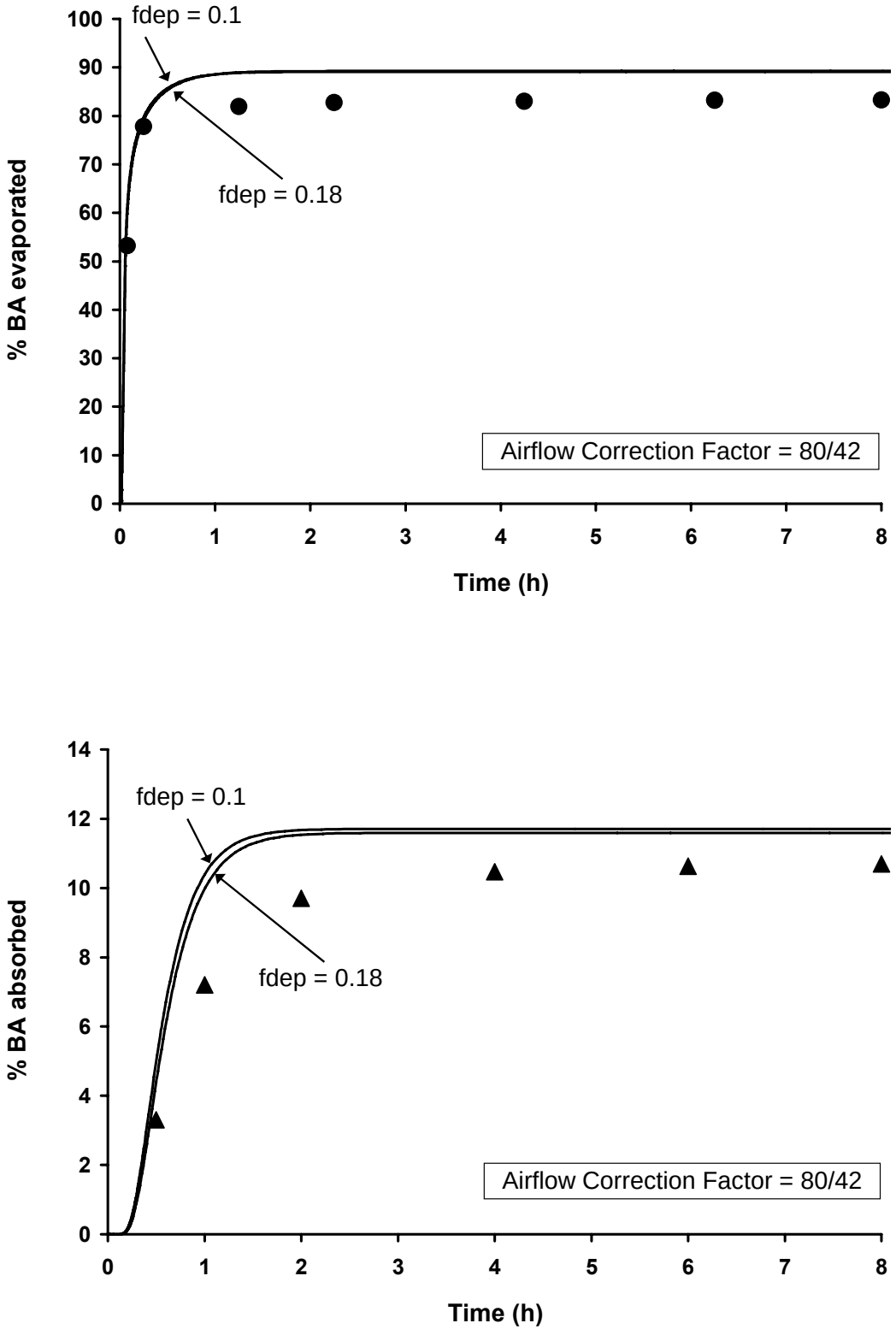


Fig. 6.4. Variation of fraction BA evaporated and absorbed with time: ACF = 80/42.

to sorption of ethanol. Fig. 6.1 shows the simulated and experimental absorption and evaporation profiles for an ACF of 0.1. Simulated profiles for both values of f_{dep} = 0.1 and f_{dep} = 0.18 are shown in this figure. The experimental data are represented through dark circles while the simulated profiles are shown by curves. It should be noted that the simulated absorption profiles for both the values of f_{dep} are very close to each other and the difference is statistically insignificant. Figs. 6.2-6.4 show the same results for ACFs of 20/42, 40/42 and 80/42.

6.5 Conclusions

A mathematical model representing transient mass-transfer pertaining to skin permeation of volatile solutes in a mixture with volatile solvents is presented in this chapter. The experimental data were found to be consistent with an unsteady-state mass transfer model using only one adjustable parameter, namely the diffusivity of BA inside the SC. Several different values of k_{evap} and two different values of f_{dep} were considered. Non-linear regression analyses were performed and the value of SC diffusivity of BA was fitted to the experimental data. Unlike the analysis on low molecular pure volatile liquids (ethanol and benzene), the disposition characteristics of BA seemed to be fairly insensitive to f_{dep} even at a low dose. In, order to use the binary model for predictive purposes, the current analysis should be extended to other representative binary solutions before any claims of predictive capability is made.

References

Bevington, P.R. (1969). Data Reduction and Error Analysis for the Physical Sciences. McGraw-Hill, New York.

Bird, B.R., Stewar,t W.E., Lightfoot, E.N., (1960). Transport Phenomena, John Wiley and Sons Inc., New York.

Cross, S.E., Magnusson, B.M., Winckle, G., Anissimov, Y., Roberts, M.S., (2003). Determination of the effect of lipophilicity on the in vitro permeability and tissue reservoir characteristics of topically applied solutes in human skin layers. Journal of Investigative Dermatology, 120, 759-764.

Johnson, M.E., Blankschtein, D., Langer, R., (1997). Evaluation of solute permeation through the stratum corneum: lateral bilayer diffusion as the primary transport mechanism. Journal of Pharmaceutical Sciences, 86, 1162-1172.

Kasting, G.B., Miller, M.A. (2006). Kinetics of finite dose absorption through skin 2. Volatile compounds. Journal of Pharmaceutical Science, 95, 268–280.

Khalil, E., Kretsos, K., Kasting, G.B., (2006). Glucose partition coefficient and diffusivity in the lower skin layers. Pharmaceutical Research, 23, 1227-1234.

Kretsos, K., Kasting, G.B., Nitsche, J.M., (2004). Distributed diffusion-clearance model for transient drug distribution within the skin. Journal of Pharmaceutical Sciences, 93, 2820-2835.

Lee, H, Ray Chaudhuri, S., Krantz, W.B., Hwang, S.-T., (2006). A model for evaporative casting of polymeric membranes incorporating convection due to density changes. Journal of Membranes Science, 284, 161-172.

Martin, A., (1993). Physical Pharmacy, Lippincott Wiliams & Wilkins, Baltimore, MD.

Miller, M.A., Bhatt, V., Kasting, G.B., (2006). Dose and airflow dependence of benzyl alcohol disposition on skin. Journal of Pharmaceutical Sciences, 95 (2), 281–291.

N'Dri-Stempfer, B., Bunge, A.L., (2005). How much can evaporation of dermally absorbed chemical reduce the systemically absorbed dose? Occupational and Environmental Exposure of Skin to Chemicals, pp. 145-146.

Nitsche, J.M., Wang, T.F., Kasting, G.B., (2006). A two-phase analysis of solute partitioning into the stratum corneum. Journal of Phamaceutical Sciences, 95, 649-666.

Peress, J., (2003). Estimate evaporative losses from spills. Chemical Engineering Progress, 32-34.

Saiyasombati, P., Kasting, G.B., (2003). Disposition of benzyl alcohol after topical application to human skin in vitro. Journal of Pharmaceutical Sciences, 92, 2128-2139.

Santhanam, A., Miller, M.A., Kasting, G.B., (2004). Absorption and evaporation of N, N-diethyl-m-toluamide from human skin in vitro. Toxicology and Applied Pharmacology, 204, 81-90.

Wang, T.-F. et al. (2006). A multiphase microscopic model for stratum corneum permeability. I. Formulation, solution and illustrative results for representative compounds. Journal of Pharmaceutical Sciences, 95, 620-648.

www.nag.com [Accessed 01/11/2007]

Yamazki, K., Kanaoka, M., (2004). Computational Prediction of the Plasma Protein-Binding Percent of Diverse Pharmaceutical Compounds, Journal of Pharmaceutical Sciences, 93, 1480-1494.

Yaws, C.L. (1999). Chemical Properties Handbook. McGraw-Hill, New York.

Wilkinson, S.C., Williams, F.M. (2002). Effects of experimental conditions on absorption of glycol ethers through human skin in vitro. International Archives of Occupational and Environmental Health, 75, 519-527.

Yamashita, F., Hashida, M., (2003). Mechanistic and empirical modeling of skin permeation of drugs. Advanced Drug Delivery Reviews. 55, 1185–1199.

CHAPTER 7

CONCLUSIONS AND RECOMMENDATIONS

7.1 Conclusions

The previous six chapters have described an in–depth investigation of the phenomenon of disposition, that is, simultaneous evaporation and absorption of volatile liquids when applied on the surface of the human skin. The study involved the development of a novel mathematical model describing the process as well as analysis of relevant transient data on skin permeation. The mathematical model was developed using the fundamental principles of material balances and transport phenomena. The model can be used as a predictive tool if the values of all the relevant thermodynamic and transport parameters for the skin-penetrant system are readily available. However, in the current situation, in the absence of such information, some parameters were fitted to the experimental data through non-linear regression analysis. The parameters that were used for regression analysis were chosen carefully. They were parameters that were either critical such that the system behavior was quite sensitive to changes in their values, or the correlations used to predict their values were not completely global.

The research is novel in terms of the mathematical model developed as well as experimental results generated. This has led to some very important conclusions pertaining to the disposition of volatile liquids on human skin. The following conclusions emanate from the pure component analyses (Chapters 2-5):

1. The evaporation and absorption characteristics for volatile liquid are coupled. This characteristic depends on the ratio of the evaporative mass-transfer coefficient (k_{evap}) and the effective diffusivity of the skin sublayers.

2. Immediately after application, low molecular weight volatile compounds (ethanol and benzene) are deposited themselves in and on the top permeable fraction of the human SC, f_{dep}.
3. For a given permeant, the amount absorbed (fraction of applied dose) depends only on system temperature, airflow velocity, dose and f_{dep}.
4. The amount absorbed for low-molecular weight volatile liquids is extremely sensitive to the parameter f_{dep} at small doses.
5. An optimized yet constant value of f_{dep} and D_{SC} are sufficient to appropriately match the experimental disposition characteristics for low molecular weight volatile liquids in their pure state.
6. Even though a constant value of f_{dep} and D_{SC} produce simulations that are sufficient close to the experimental results, relaxing these restrictions by means of a variable diffusivity or a variable deposition depth substantially improves the results.
7. The optimal value for f_{dep} are much higher for individual dose regressions than they are for combined dose regression analyses.
8. The individual dose regression analysis results in a variable optimal f_{dep} which is related to the applied dose through a power-law relationship.
9. The optimal values for fitted parameters depend on the method of fitting (constant D_{SC} - constant f_{dep}, variable D_{SC} - constant f_{dep} and constant D_{SC} - variable f_{dep}). This variation is less prominent for ethanol than it is for benzene.
10. The optimal f_{dep} values for benzene for both the individual dose and combined dose regression analyses are much higher than the corresponding optimal values for ethanol.

This may be indicative of higher disruption of SC lipids by benzene, leading to a larger permeation enhancement.

Other than the pure component analysis on mostly low molecular-weight volatile liquids, a separate more complex analysis was performed on the higher molecular weight volatile liquid, benzyl alcohol, as applied from binary mixtures. The conclusions emanating from this analysis are:

11. The amount absorbed and evaporated is not sensitive to f_{dep} as was in the case of ethanol and benzene.
12. The average airflow velocity used for ethanol and benzene is appropriate for BA.
13. Even though majority of the dosed ethanol evaporates rapidly, there is a small fraction ethanol that penetrates the SC, which may lead to some permeation enhancement. This may be the reason behind a roughly 5-fold increase in the value of D_{SC} as compared to what was predicted by the WKN correlation.

7.2 **Recommendations**

14. The transient mass-transfer model developed should also be tested on permeation of well-behaved solution-diffusion membranes such as silicone rubber. This is necessary in order to test the strength of the model and differentiate between the effect of the limitations of the model and the intricacies of the biological system on the discrepancies between the model predictions and the experimental data.

15. Better correlations for thermodynamic and transport parameters need to be chosen or devised such that the need for optimization or fitting is eventually avoided. One of the most critical parameters in this group is SC diffusivity.
16. The components treated in the system are volatile with appreciable heats of vaporization. Rapid evaporation of volatile substances may result in cooling of the system. This would result in a decrease in the vapor pressure of the evaporating species. A decrease in vapor pressure would result in a decrease in the evaporation rate and also a reduction in the cooling rate. Thus this coupled compensatory behavior may result in a slightly different mass transfer characteristic than the corresponding isothermal system. The current system ignores this effect by assuming isothermal conditions. Non-isothermal conditions may be introduced by adding the appropriate energy balance equation for each layer in the system (liquid and skin). However, the cooling effect may not be substantial because the blood vessels underneath the skin-sublayers are at a substantially higher temperature than ambient and act as a constant heat source. Preliminary experimental findings suggest a maximum cooling of only 2.5 degrees centigrade. This may not be substantial to appreciably alter the vapor pressure and the disposition characteristics of the system.

Appendix A - Derivation of Coordinate Transformation Rules for Skin Layers

The differential for any arbitrary function $\phi = \phi\left(z^{\bullet}, t\right)$ can be written as

$$d\phi = \left.\frac{\partial \phi}{\partial z^{\bullet}}\right|_{t} dz^{\bullet} + \left.\frac{\partial \phi}{\partial t}\right|_{z^{\bullet}} dt \tag{A1}$$

where $\bullet$ can be SC or VT. Differentiating both sides of Eq. (A1) partially with respect to z, we obtain

$$\left.\frac{\partial \phi}{\partial z}\right|_{t} = \left.\frac{\partial \phi}{\partial z^{\bullet}}\right|_{t} \left.\frac{\partial z^{\bullet}}{\partial z}\right|_{t} + \left.\frac{\partial \phi}{\partial t}\right|_{z^{\bullet}} \left.\frac{\partial t}{\partial z}\right|_{t} \text{ or } \left.\frac{\partial \phi}{\partial z}\right|_{t} = \left.\frac{\partial \phi}{\partial z^{\bullet}}\right|_{t} \left.\frac{\partial z^{\bullet}}{\partial z}\right|_{t} \tag{A2}$$

Similarly, the second partial derivative of ϕ with respect to *z* can be written as:

$$\left.\frac{\partial^2 \phi}{\partial z^2}\right|_{t} = \left.\frac{\partial}{\partial z}\right|_{t} \left(\left.\frac{\partial \phi}{\partial z}\right|_{t}\right) \text{ or } \left.\frac{\partial^2 \phi}{\partial z^2}\right|_{t} = \left.\frac{\partial}{\partial z}\right|_{t} \left(\left.\frac{\partial \phi}{\partial z}\right|_{t}\right) \tag{A3}$$

Combining Eqs. (A2) and (A3), we obtain

$$\left.\frac{\partial^2 \phi}{\partial z^2}\right|_{t} = \left.\frac{\partial}{\partial z^{\bullet}}\right|_{t} \left(\left.\frac{\partial \phi}{\partial z^{\bullet}}\right|_{t} \left.\frac{\partial z^{\bullet}}{\partial z}\right|_{t}\right) \left.\frac{\partial z^{\bullet}}{\partial z}\right|_{t} \tag{A4}$$

Similarly, differentiating both sides of Eq. (A1) partially with respect to *t*, we obtain

$$\left.\frac{\partial \phi}{\partial t}\right|_{z} = \left.\frac{\partial \phi}{\partial z^{\bullet}}\right|_{t} \left.\frac{\partial z^{\bullet}}{\partial t}\right|_{z} + \left.\frac{\partial \phi}{\partial t}\right|_{z^{\bullet}} \frac{\partial t}{\partial t} \text{ or } \left.\frac{\partial \phi}{\partial t}\right|_{z} = \left.\frac{\partial \phi}{\partial z^{\bullet}}\right|_{t} \left.\frac{\partial z^{\bullet}}{\partial t}\right|_{z} + \left.\frac{\partial \phi}{\partial t}\right|_{z^{\bullet}} \tag{A5}$$

Thus, the generalized rules for coordinate transformation are given by Eqs. (A2), (A4) and (A5).

Now specifically in the given problem, for SC and VT, we have

$$\left.\frac{\partial z^{SC}}{\partial z}\right|_{t} = \frac{1}{L_{SC}} \tag{A6}$$

$$\left.\frac{\partial z^{VT}}{\partial z}\right|_t = \frac{1}{-L_{VT}} \tag{A7}$$

Therefore, for SC and VT, the partial derivatives in the old coordinate system are related to the new through the following relationships

$$\left.\frac{\partial}{\partial z}\right|_t = \frac{1}{L_{SC}}\left.\frac{\partial}{\partial z^{SC}}\right|_t \tag{A8}$$

$$\left.\frac{\partial}{\partial z}\right|_t = -\frac{1}{L_{VT}}\left.\frac{\partial}{\partial z^{VT}}\right|_t \tag{A9}$$

$$\left.\frac{\partial^2}{\partial z^2}\right|_t = \left.\frac{\partial}{\partial z^{SC}}\right|_t \left(\left.\frac{\partial}{\partial z^{SC}}\right|_t \frac{1}{L_{SC}}\right)\frac{1}{L_{SC}} = \frac{1}{L_{SC}^2}\left.\frac{\partial^2}{\partial z^{SC^2}}\right|_t \tag{A10}$$

$$\left.\frac{\partial^2}{\partial z^2}\right|_t = \left.\frac{\partial}{\partial z^{VT}}\right|_t \left(\left.\frac{\partial}{\partial z^{VT}}\right|_t \frac{1}{-L_{VT}}\right)\frac{1}{-L_{VT}} = \frac{1}{L_{VT}^2}\left.\frac{\partial^2}{\partial z^{VT^2}}\right|_t \tag{A11}$$

Also the time derivatives relating the old and new coordinate systems is written as:

$$\left.\frac{\partial}{\partial t}\right|_z = \left.\frac{\partial}{\partial t}\right|_{z^{SC}} \tag{A12}$$

$$\left.\frac{\partial}{\partial t}\right|_z = \left.\frac{\partial}{\partial t}\right|_{z^{VT}} \tag{A13}$$

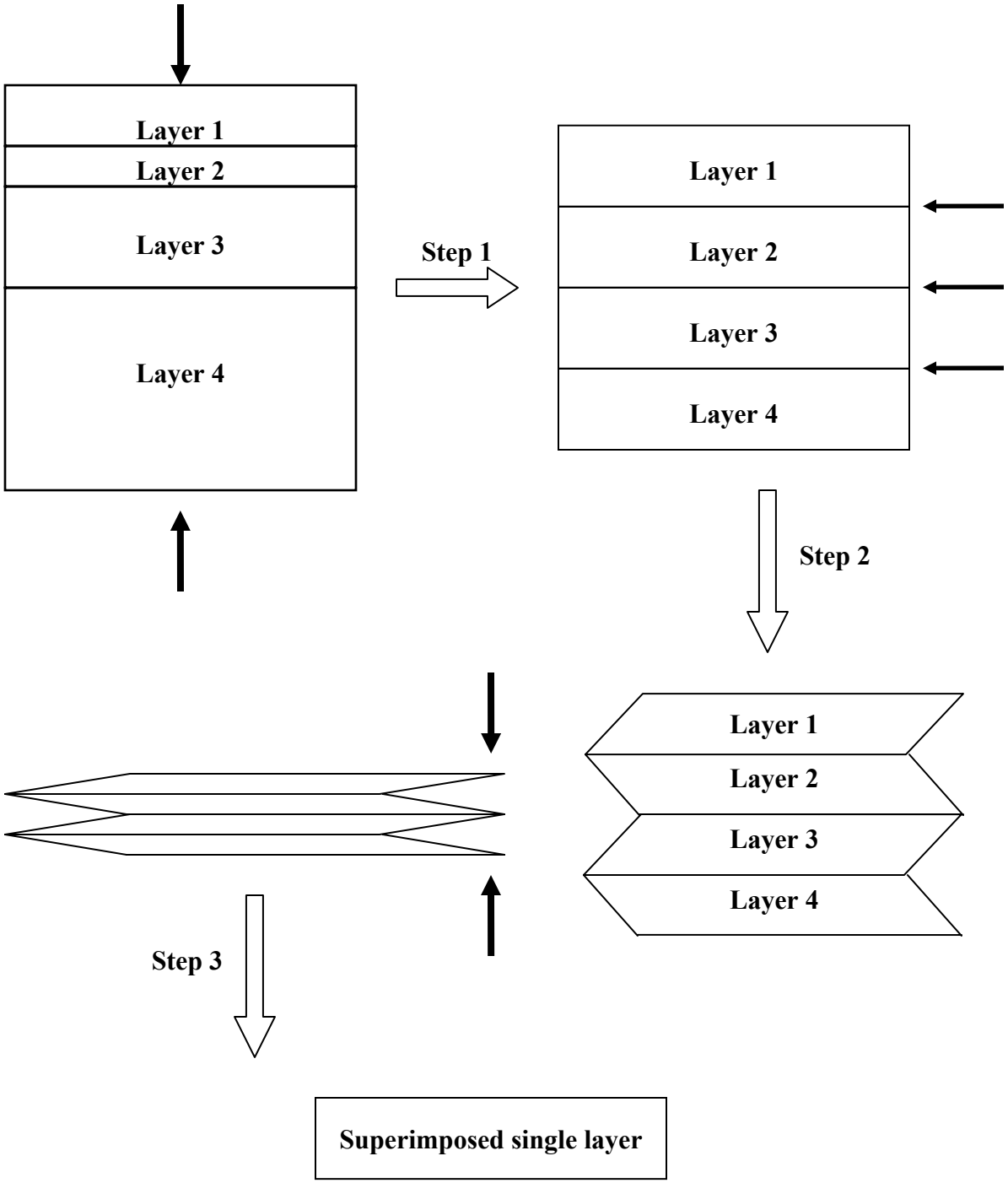


Figure A1. Normalization of system boundaries through compression and superimposition of layers in a multilayered system

Appendix B: Determination of Partition Coefficient of Ethanol in SC

The equilibrium quantity of ethanol and water absorbed into dry SC from a 95% aqueous ethanol solution can be obtained from Fig. 2 of the article by Berner and Mazzenga (1989). Using data extraction software the following are obtained:

Mass ratio of ethanol and SC (M_{etoh}) = 0.3499 (g etoh/ g SC)

Mass ratio of water and SC (M_{H_2O}) = 0.2513 (g water/ g SC)

Thus, the composition of SC at this equilibrium is given as:

$$\omega_{etoh} = \frac{0.3499}{1 + 0.3499 + 0.2513} = 0.2185$$

$$\omega_{H_2O} = \frac{0.2513}{1 + 0.3499 + 0.2513} = 0.1569$$

Now, in this sorbed and equilibrated state, the overall density of the SC can be obtained through an extension of the original SC density equation (Eqn A4) as proposed in [Nitsche, et al. (2006). A two-phase analysis of solute partitioning into the stratum corneum. Journal of Pharmaceutical Sciences, 95(3), 649-666]. Thus, the density of sorbed SC will be given as:

$$\rho_{SC,wet} = \frac{1 + M_{etoh} + M_{H_2O}}{\left(\omega_{lip}/\rho_{lip}\right) + \left(\omega_{pro}/\rho_{pro}\right) + \left(M_{etoh}/\rho_{etoh}\right) + \left(M_{H_2O}/\rho_{H_2O}\right)}$$

$$= \frac{1 + 0.3499 + 0.2513}{\left(0.1/0/9\right) + \left(0.9/1.37\right) + \left(0.3499/0.79\right) + \left(0.2513/1.00\right)} = \frac{1.6012}{1.4622} = 1.0951 \text{ gm/cc}$$

Therefore, concentration of ethanol in SC is given as:

$$\rho_{etoh,SC} = \omega_{etoh}\rho_{SC,wet} = \left(0.2185\right)\left(1.0951\right) = 0.2393 \text{ gm/cc}$$

Similarly, the partition coefficient of ethanol in SC is given as:

$$K^{etoh} = \frac{\rho_{etoh,SC}}{\rho_{etoh}} = \frac{0.2393}{0.79} = 0.3029$$

Appendix C

The boundary condition at the DS – air interface ($z = L$) cannot be obtained by the simple continuity of flux condition, as is the case with stationary boundaries. Since $z = L$ is a moving boundary, the required boundary condition can be obtained by an overall and component mass balance on the entire donor solution and the surrounding gaseous phase. The component mass balance is given as

$$\frac{d}{dt}\left[\int_0^{L(t)} \rho_A^{DS} dz\right]_{DS} + \frac{d}{dt}\left[\int_{L(t)}^{\infty} \rho_A^{DS} dz\right]_G = \left(n_A^{SC}\right)_{z=0} - \left(n_A^G\right)_{z=\infty} = 0 \tag{C123}$$

where the first and second terms in left-hand side (LHS) of Eq. (C1) represent the rate of accumulation/ depletion of the penetrant component A in the DS and in the surrounding gaseous phase ($z = L$ to $z = \infty$), respectively. The first term on the right hand side (RHS) of Eq. (C1) represents the mass flux of component A into the SC from the donor solution. The second term on the RHS of Eq. (C1) represents gas-phase mass flux at $z = \infty$ and is equal to zero since it is, by definition, impenetrable.

The application of a Leibnitz's transformation on the LHS of Eq. (C1) yields

$$\frac{d}{dt}\left[\int_0^{L(t)} \rho_A^{DS} dz\right] + \frac{d}{dt}\left[\int_{L(t)}^{\infty} \rho_A^{G} dz\right]_G = \int_0^{L(t)} \frac{\partial \rho_A^{DS}}{\partial t} dz + \left(\rho_A^{DS}\right)_{z=L} \frac{dL}{dt} + \int_{L(t)}^{\infty} \frac{\partial \rho_A^{G}}{\partial t} dz - \left(\rho_A^{G}\right)_{z=L} \frac{dL}{dt} \tag{C124}$$

However, using the continuity equations on the first and third terms of the RHS of Eq. (C2) we obtain:

$$\frac{d}{dt}\left[\int_0^{L} \rho_A^{DS} dz\right] + \frac{d}{dt}\left[\int_{L}^{\infty} \rho_A^{G} dz\right] = -\int_0^{L} \frac{\partial n_A^{DS}}{\partial z} dz + \left(\rho_A^{DS}\right)_{z=L} \frac{dL}{dt} - \int_{L}^{\infty} \frac{\partial n_A^{G}}{\partial z} dz - \left(\rho_A^{G}\right)_{z=L} \frac{dL}{dt}$$

$$= \left(n_A^{DS}\right)_{z=0} - \left(n_A^{DS}\right)_{z=L} + \left(n_A^G\right)_{z=L} - \cancelto{\mathbf{0}}{\left(n_A^G\right)_{z=\infty}} + \left[\left(\rho_A^{DS}\right)_{z=L} - \left(\rho_A^G\right)_{z=L}\right]\frac{dL}{dt} \tag{C125}$$

Comparing the RHS of Eqs. (C1) and (C3), we obtain

$$\left[\left(\rho_A^{DS}\right)_{z=L} - \left(\rho_A^{G}\right)_{z=L}\right]\frac{dL}{dt} = \left(n_A^{DS}\right)_{z=L} - \left(n_A^{G}\right)_{z=L} \tag{C126}$$

The density of component A in the gaseous phase may easily be neglected with respect to the same in the liquid phase, thereby enabling one to rewrite Eq. (C4) as:

$$\left(\rho_A^{DS}\right)_{z=L}\frac{dL}{dt} = \left(n_A^{DS}\right)_{z=L} - \left(n_A^{G}\right)_{z=L} \tag{C127}$$

The first and second terms on the RHS of Eq. (C4) represent the total flux for the penetrant in the DS and gaseous phase, respectively. The latter has been expressed through evaporative mass-transfer coefficient. Thus, we have:

$$\left(\rho_A^{DS}\right)_{z=L}\frac{dL}{dt} = \left(j_A^{DS}\right)_{z=L} + \left(\rho_A^{DS}W^{DS}\right)_{z=L} - \left(n_A^{G}\right)_{z=L} \text{ or}$$

$$\left(\rho_A^{DS}\right)_{z=L}\left[\frac{dL}{dt} - \left(W^{DS}\right)_{z=L}\right] = \left(j_A^{DS}\right)_{z=L} - \left(n_A^{G}\right)_{z=L} \tag{C128}$$

Similarly, the overall mass balance on the DS and surrounding phase yields

$$\frac{d}{dt}\left[\int_0^{L(t)}\rho^{DS}dz\right] + \frac{d}{dt}\left[\int_{L(t)}^{\infty}\rho^{G}dz\right] = \left(n_T^{SC}\right)_{z=0} \tag{C129}$$

where the first and second terms in left-hand side (LHS) of Eq. (C1) represent the rate of accumulation/ depletion of the total mass in the DS and in the surrounding gaseous phase ($z = L(t)$ to $z = \infty$), respectively. The RHS of Eq. (C7) represents the total mass flux into the SC from the DS. An application of a Leibnitz's transformation yields

$$\frac{d}{dt}\left[\int_0^L \rho^{DS} dz\right] + \frac{d}{dt}\left[\int_L^\infty \rho^G dz\right] = -\int_0^L \frac{\partial\left(\rho^{DS} W^{DS}\right)}{\partial z} dz + \left(\rho^{DS}\right)_{z=L} \frac{dL}{dt} - \int_L^\infty \frac{\partial\left(\rho^G W^G\right)}{\partial z} dz - \left(\rho^G\right)_{z=L} \frac{dL}{dt}$$

$$= \left(n_T^{DS}\right)_{z=0} - \left(n_T^{DS}\right)_{z=L} + \left(n_T^G\right)_{z=L} \cancel{- \left(n_T^G\right)_{z=\infty}}^{\mathbf{0}} + \left[\left(\rho^{DS}\right)_{z=L} - \left(\rho^G\right)_{z=L}\right]\frac{dL}{dt} \tag{C130}$$

Comparing the RHS of Eqs. (C7) and (C8), we get

$$\left[\left(\rho^{DS}\right)_{z=L} - \left(\rho^G\right)_{z=L}\right]\frac{dL}{dt} = \left(n_T^{DS}\right)_{z=L} - \left(n_T^G\right)_{z=L} \tag{C131}$$

Similar to Eq. (C5), neglecting net gaseous density with respect to liquid density, we have

$$\left(\rho^{DS}\right)_{z=L}\left[\frac{dL}{dt} - \left(W^{DS}\right)_{z=L}\right] = -\left(n_T^G\right)_{z=L} \tag{C132}$$

Comparing Eqs. (C6) and (C10), we obtain:

$$\frac{\left(\rho_A^{DS}\right)_{z=L}\left[\frac{dL}{dt} - \left(W^{DS}\right)_{z=L}\right]}{\left(\rho^{DS}\right)_{z=L}\left[\frac{dL}{dt} - \left(W^{DS}\right)_{z=L}\right]} = \left(\omega_A^{DS}\right)_{z=L} = \frac{\left(j_A^{DS}\right)_{z=L} - \left(n_A^G\right)_{z=L}}{-\left(n_T^G\right)_{z=L}} \text{ or}$$

$$\left(j_A^{DS}\right)_{z=L} = \left(n_A^G - \omega_A^{DS} n_T^G\right)_{z=L} \tag{C133}$$

Also, the movement of the boundary $z = L$ is given by Eq. (C10)

Appendix D

Program ETOHDALLSIM simulates the evaporation and absorption of ethanol through skin sub layers. It has the option of using constant or variable SC diffusivity and fractional deposition depth. This program can be used for benzene by changing the physicochemical properties of and transport parameters.

```
PROGRAM ETOHDALLSIM
*
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      IMPLICIT INTEGER (I-N)
*
*     NPDE   = Number of PDEs
*     NCODE  = Number of Coupled ODEs
*     M      = Determines the Coordinate system (0 = Rectangular Cartesian)
*     NPTS   = Number of Mesh Points
*     NXI    = Number of PDE-ODE Coupling Points
*     NXFIX  = Number of Fixed Mesh Points
      PARAMETER (NPDE=2, NCODE=1, M=0, NPTS=201, NXI=0, NXFIX=0)
*
*     DO NOT CHANGE THE LINES BELOW!
************************************************************************
      PARAMETER (NWKRES = NPDE*(3*NPDE+6*NXI+NPTS+15)+ NXI+NCODE
   $                + 7*NPTS+NXFIX+2+9000)
!     Increase the value 9000 if more workspace is needed
      PARAMETER (NEQN=NPDE*NPTS+NCODE)
      PARAMETER (NIW=25*NEQN+25+NXFIX+90000)
!     Dimension of the array IW. Pg. 17. DO NOT CHANGE unless IFAIL 15!
      PARAMETER (LENODE=(6+5)*NEQN+50) ! For theta method, Pg. 17!
      PARAMETER (NW=4*NEQN+11*NEQN/2+1+NWKRES+LENODE+20000)
************************************************************************
*
*     User Defined Parameters
      PARAMETER (NEXDP = 40)
      PARAMETER (NPARAM = 1)
      PARAMETER (MODE = 0)
      PARAMETER (NFDEPVR = 1)
*
*     Variable Declaration
      DOUBLE PRECISION U(NEQN), X(NEQN), RTOL(NEQN), ATOL(NEQN), XI(1),
   +                   XFIX(1), IW(NIW), W(NW), ALGOPT(30), SOL(NPTS,NPDE),
   +                   XDATA(NEXDP), YDATA(NEXDP), YFIT(NEXDP),
SIGMAY(NEXDP)
*     IW(NIW) - Output variable, W(NW) - Workspace variable
      INTEGER ITIME(7)
      LOGICAL REMESH, THETA
      CHARACTER*1 LAOPT, NORM
*
*     External Subroutines
      EXTERNAL UVINIT,ODEDEF,PDEDEF,BNDARY,D03PPF,D03PCL,D03PCK,X05AAF
*     D03PCL replaces MONITF as dummy subroutine!
```

```
*
*       Common Blocks
        COMMON /EXPMT/ TEMP, DOSE, VHTH, ETCSAT, ETSCIN
        COMMON /VHPAR/ RHET, WMET
        COMMON /SCPAR/ RHSC, SCDR, SCTH, DWNK, DASC, FDEP, CNFDEP
        COMMON /VEPAR/ RHVE, VETH, DAVE
        COMMON /INTER/ FLXG, FLSCVH, FLVESC, FBTM, SCVHPC, VESCPC, PTOR
        COMMON /VPCPH/ VPCA, VPCB, VPCC, VPCD, VPCE
        COMMON /CRRCT/ FCOR, DCOR
        COMMON /VRDIF/ SLPD, CTRN, DMINFR, DSATFR
*
*       Initialize the constants specific to the system of interest
        CALL CONSTS
*
*       Experimental Data
        UFLX = 1.0D6*3600.0D0
        AREA = 0.79D0
*
        XDATA(1)  = 0.08D0*3600.0D0
        XDATA(2)  = 0.17D0*3600.0D0
        XDATA(3)  = 0.25D0*3600.0D0
        XDATA(4)  = 0.33D0*3600.0D0
        XDATA(5)  = 0.67D0*3600.0D0
        XDATA(6)  = 1.00D0*3600.0D0
        XDATA(7)  = 2.00D0*3600.0D0
        XDATA(8)  = 4.00D0*3600.0D0
        XDATA(9)  = 8.50D0*3600.0D0
        XDATA(10) = 24.00D0*3600.0D0
*
        DO 101 I = 1,10
        XDATA(I+10) = XDATA(I)
        XDATA(I+20) = XDATA(I)
        XDATA(I+30) = XDATA(I)
101     CONTINUE
*
*
*       40 microliters
        YDATA(1)  = 0.00D0
        YDATA(2)  = 0.01D0
        YDATA(3)  = 0.02D0
        YDATA(4)  = 0.04D0
        YDATA(5)  = 0.10D0
        YDATA(6)  = 0.13D0
        YDATA(7)  = 0.16D0
        YDATA(8)  = 0.17D0
        YDATA(9)  = 0.17D0
        YDATA(10) = 0.18D0
*
*       20 microliters
        YDATA(11)  = 0.01D0
        YDATA(12)  = 0.02D0
        YDATA(13)  = 0.04D0
        YDATA(14)  = 0.06D0
```

```
      YDATA(15)  = 0.13D0
      YDATA(16)  = 0.17D0
      YDATA(17)  = 0.20D0
      YDATA(18)  = 0.22D0
      YDATA(19)  = 0.23D0
      YDATA(20)  = 0.23D0
*
*     10 microliters
      YDATA(21)  = 0.02D0
      YDATA(22)  = 0.06D0
      YDATA(23)  = 0.11D0
      YDATA(24)  = 0.16D0
      YDATA(25)  = 0.27D0
      YDATA(26)  = 0.33D0
      YDATA(27)  = 0.38D0
      YDATA(28)  = 0.39D0
      YDATA(29)  = 0.40D0
      YDATA(30) = 0.41D0
*
*     5 microliters
      YDATA(31)  = 0.02D0
      YDATA(32)  = 0.05D0
      YDATA(33)  = 0.08D0
      YDATA(34)  = 0.11D0
      YDATA(35)  = 0.19D0
      YDATA(36)  = 0.23D0
      YDATA(37)  = 0.27D0
      YDATA(38)  = 0.29D0
      YDATA(39)  = 0.29D0
      YDATA(40)  = 0.29D0
*
*
*     Write the Solution Files
      lsim = 11 ! Description of the Simulation
      lAsc40 = 21 ! Mass Fraction of Solvent in the Stratum Corneum !
      lAsc20 = 22 ! Mass Fraction of Solvent in the Stratum Corneum !
      lAsc10 = 23 ! Mass Fraction of Solvent in the Stratum Corneum !
      lAsc5  = 24 ! Mass Fraction of Solvent in the Stratum Corneum !
      lAve40 = 31 ! Mass Fraction of Solvent in the Viable Epidermis !
      lAve20 = 32 ! Mass Fraction of Solvent in the Viable Epidermis !
      lAve10 = 33 ! Mass Fraction of Solvent in the Viable Epidermis !
      lAve5  = 34 ! Mass Fraction of Solvent in the Viable Epidermis !
      llvt40 = 41 ! Instantaneous Thickness of the Donor Solution !
      llvt20 = 42 ! Instantaneous Thickness of the Donor Solution !
      llvt10 = 43 ! Instantaneous Thickness of the Donor Solution !
      llvt5  = 44 ! Instantaneous Thickness of the Donor Solution !
      lflx40 = 51 ! Fluxes
      lflx20 = 52
      lflx10 = 53
      lflx5  = 54
      lmbc40 = 61 ! Mass balance component
      lmbc20 = 62 ! Mass balance component
      lmbc10 = 63 ! Mass balance component
```

```
      lmbc5  = 64 ! Mass balance component
      lcmf40 = 71 ! Cumulative Fluxes
      lcmf20 = 72 ! Cumulative Fluxes
      lcmf10 = 73 ! Cumulative Fluxes
      lcmf5  = 74 ! Cumulative Fluxes
      lext = 18 ! Anciliary data
*
      itrace = 2 ! See D03PPF Manual, Pg. 18 !
*
      OPEN (lsim, file='sim.dat', status='new')
      OPEN (lAsc40, file='Asc40.dat', status='new')
      OPEN (lAsc20, file='Asc20.dat', status='new')
      OPEN (lAsc10, file='Asc10.dat', status='new')
      OPEN (lAsc5, file='Asc5.dat', status='new')
      OPEN (lAve40, file='Ave40.dat', status='new')
      OPEN (lAve20, file='Ave20.dat', status='new')
      OPEN (lAve10, file='Ave10.dat', status='new')
      OPEN (lAve5, file='Ave5.dat', status='new')
      OPEN (llvt40, file='lvt40.dat', status='new')
      OPEN (llvt20, file='lvt20.dat', status='new')
      OPEN (llvt10, file='lvt10.dat', status='new')
      OPEN (llvt5, file='lvt5.dat', status='new')
      OPEN (lflx40, file='flx40.dat', status='new')
      OPEN (lflx20, file='flx20.dat', status='new')
      OPEN (lflx10, file='flx10.dat', status='new')
      OPEN (lflx5, file='flx5.dat', status='new')
      OPEN (lmbc40, file='mbc40.dat', status='new')
      OPEN (lmbc20, file='mbc20.dat', status='new')
      OPEN (lmbc10, file='mbc10.dat', status='new')
      OPEN (lmbc5, file='mbc5.dat', status='new')
      OPEN (lcmf40, file='cmf40.dat', status='new')
      OPEN (lcmf20, file='cmf20.dat', status='new')
      OPEN (lcmf10, file='cmf10.dat', status='new')
      OPEN (lcmf5, file='cmf5.dat', status='new')
!     OPEN (lext, file='fit.dat', status='new')
*
      NDIFFVR = 1
      IF (NFDEPVR .NE. 0) THEN
      NDIFFVR = 0
      ELSE
      END IF
*
      WRITE(lsim,*) ' 3-Layer Pure Penetrant Isothermal Simulation '
      WRITE(lsim,*) ' The Penetrant in ETHANOL'
      WRITE(lsim,*) 'Number of Mesh Points = ', NPTS
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Molecular Weight of the Penetrant = ', WMET
      WRITE(lsim,*) 'Constant System Temperature = ', TEMP - 273.15D0
      WRITE(lsim,*) 'Density of Penetrant = ', RHET
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Density of SC Phase = ', RHSC
```

```
      WRITE(lsim,*) 'Thickness of Dry SC Phase (in microns) = ', SCDR*1.0D4
      IF (NDIFFVR .EQ. 0) THEN
      WRITE(lsim,*) 'Constant Diffusivity used'
      WRITE(lsim,*) 'Predicted Diffusivity of A in SC = ', DWNK
      WRITE(lsim,*) 'Correction Factor for SC Diffusivity = ', DCOR
      WRITE(lsim,*) 'Corrected Diffusivity of A in SC = ', DWNK*DCOR
      ELSE
      WRITE(lsim,*) 'Concentration-dependent Diffusivity used'
      WRITE(lsim,*) 'Initial Multiplier for Lower Diffusivity = ', DMINFR
      WRITE(lsim,*) 'Initial Multiplier for Higher Diffusivity = ', DMINFR*DSATFR
      WRITE(lsim,*) 'Initial Lower Limit of SC Diffusivity = ', DWNK*DMINFR
      WRITE(lsim,*) 'Initial Upper Limit of SC Diffusivity = ', DWNK*DMINFR*DSATFR
      END IF
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Density of VE Phase = ', RHVE
      WRITE(lsim,*) 'Thickness of VE Phase (in microns) = ', VETH*1.0D4
      WRITE(lsim,*) 'Diffusivity of A in VE = ', DAVE
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Predicted KEVAP = ', FLXG*3600.0D0/(FCOR*RHET)
      WRITE(lsim,*) 'Correction Factor for FLXG = ', FCOR
      WRITE(lsim,*) 'Corrected KEVAP = ', FLXG*3600.0D0/RHET
      WRITE(lsim,*) 'SC-VH partition coeffcient for Ethanol = ', SCVHPC
      WRITE(lsim,*) 'VE-SC partition coeffcient for Ethanol = ', VESCPC
*
*     Initialize Mesh - Uniform Mesh
*
      DO 910 I = 1, NPTS
            X(I) = DBLE(I-1)/(NPTS-1.0D0)
910   CONTINUE
*
*     Set Remesh Parameters
*
      REMESH = .false.
      NRMESH = 0
      DXMESH = 0.0D0
      TRMESH = 1.0D0
      IPMINF = 0
      XRATIO = 1.5D0
      CONST  = 2.0D0/(NPTS-1)
      IND = 0
      ITASK = 2
      NORM = 'A'
      LAOPT= 'S' !use the sparse matrix routines
      THETA = .false.
*
      DO 920 I=1,30
            ALGOPT(I)=0.0D0
920   CONTINUE
      ALGOPT(4)=2.0D0 !Does not perform the Petzold test
      ALGOPT(29)=0.5 !NAG Deafault = 0.1
      ALGOPT(30)=1.0D-300 !NAG Default 1.0D-2!
```

```
*
*       Set Time Step & Convergence Criterion
        DELT = 1.0D-3
        WRITE(lsim,*) 'Time Increament Step Size =', DELT
        DO 930 K = 1, NEQN
        RTOL(K) = 1.0D-8
        ATOL(K) = 1.0D-8
930     CONTINUE
        ITOL = 4
        WRITE(lsim,*) 'RTOL = ', RTOL(1)
        WRITE(lsim,*) 'ATOL = ', ATOL(1)
        WRITE(lsim,*) 'ITOL = ', ITOL
*
*       Write Simulation Date and Time
        CALL X05AAF(ITIME)
        WRITE(lsim,*) 'Month / Day / Year = ', ITIME(2),'/',ITIME(3),'/',ITIME(1)
        WRITE(lsim,*) 'Hour / Minute / Second = ', ITIME(4),'/',ITIME(5),'/',ITIME(6)
*
        NFREE = NEXDP - NPARAM
*
*       40 microliters ******************************************************
*
        IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
        VOL = 40.0D-3
        DOSE = (VOL/AREA)*RHET
        IF (NFDEPVR .EQ. 0) THEN
        FDEP = CNFDEP
        ELSE
        CALL DEPDEP (VOL, FDEP)
        END IF
        CORR = FDEP*SCDR*ETCSAT/RHET
        VHTH = VOL/AREA - CORR
        SCTH = SCDR + CORR
*
        NDPD = ((NPTS-1)*FDEP) + 1
        WRITE(lsim,*) '*****************************************************'
        WRITE(lsim,*) 'Initial Volume of Penetrant (in microlit) = ', VOL*1.0D3
        WRITE(lsim,*) 'Fractional Deposition Depth (%) =', FDEP*100.0D0
        WRITE(lsim,*) 'Number of Permeant Deposited Mesh Points = ', NDPD
        WRITE(lsim,*) 'Initial VH Thickness (in microns) = ', VHTH*1.0D4
        WRITE(lsim,*) 'Thickness of Soaked SC (in microns) = ', SCTH*1.0D4
        WRITE(lsim,*) 'Initial Mass of Penetrant (microg/cm2) = ', DOSE*1.0D6
*
        DO 640 IT640 = 1, 5000
        TOUT = T + DELT
```

```
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 6401 KL = 1,NPDE
            DO 6402 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
6402        CONTINUE
6401  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VETH/DBLE(NPTS-1)
      FCAL = RHVE*DAVE*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
      WRITE(llvt40,1004) THRS, VHTI
*
      ELSE
*
      ETTP = SOL(NPTS,1)
      CALL DENSTY (ETTP, RHET, RHSC, RHAV)
      FEVP = FLXG*ETTP/ETSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
      CALL MASSBL (NPTS, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
      CFLX = CFLX + FTTL*TDIF ! Cumulative Flux
      CHECK = (TMAS + CFLX)/DOSE ! Should be ~ 1
      WRITE(lmbc40,1004) THRS, CHECK
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
```

```
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 6403 NN = 1, 10
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
6403  CONTINUE
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 6409
      ELSE
      END IF
      PPENOLD = PPEN
*
*     Writing the Solutions!
      WRITE(lAsc40,1001) THRS
      WRITE(lAsc40,1002) (SOL(J,1), J=1, NPTS)
      WRITE(lAve40,1001) THRS
      WRITE(lAve40,1002) (SOL(J,2), J=1, NPTS)
      WRITE(lflx40,1004) THRS, ABS(FEVP)*UFLX, ABS(FBTM)*UFLX, ABS(FCAL)*UFLX
      WRITE(lcmf40,1004) THRS, PEVP, PPEN, PTTL
*
640   CONTINUE
*
*     END 40 micoliters ************************************************
*
*     20 microliters ***************************************************
*
6409  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 20.0D-3
      DOSE = (VOL/AREA)*RHET
      IF (NFDEPVR .EQ. 0) THEN
      FDEP = CNFDEP
      ELSE
```

```
      CALL DEPDEP (VOL, FDEP)
      END IF
      CORR = FDEP*SCDR*ETCSAT/RHET
      VHTH = VOL/AREA - CORR
      SCTH = SCDR + CORR
*
      NDPD = ((NPTS-1)*FDEP) + 1
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Initial Volume of Penetrant (in microlit) = ', VOL*1.0D3
      WRITE(lsim,*) 'Fractional Deposition Depth (%) =', FDEP*100.0D0
      WRITE(lsim,*) 'Number of Permeant Deposited Mesh Points = ', NDPD
      WRITE(lsim,*) 'Initial VH Thickness (in microns) = ', VHTH*1.0D4
      WRITE(lsim,*) 'Thickness of Soaked SC (in microns) = ', SCTH*1.0D4
      WRITE(lsim,*) 'Initial Mass of Penetrant (microg/cm2) = ', DOSE*1.0D6
*
      DO 620 IT620 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 6201 KL = 1,NPDE
            DO 6202 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
6202        CONTINUE
6201  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VETH/DBLE(NPTS-1)
      FCAL = RHVE*DAVE*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
      WRITE(llvt20,1004) THRS, VHTI
*
```

```
      ELSE
*
      ETTP = SOL(NPTS,1)
      CALL DENSTY (ETTP, RHET, RHSC, RHAV)
      FEVP = FLXG*ETTP/ETSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
      CALL MASSBL (NPTS, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
      CFLX = CFLX + FTTL*TDIF ! Cumulative Flux
      CHECK = (TMAS + CFLX)/DOSE ! Should be ~ 1
      WRITE(lmbc20,1004) THRS, CHECK
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 6203 NN = 11, 20
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
6203  CONTINUE
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 6209
      ELSE
      END IF
      PPENOLD = PPEN
*
*     Writing the Solutions!
      WRITE(lAsc20,1001) THRS
      WRITE(lAsc20,1002) (SOL(J,1), J=1, NPTS)
      WRITE(lAve20,1001) THRS
      WRITE(lAve20,1002) (SOL(J,2), J=1, NPTS)
      WRITE(lflx20,1004) THRS, ABS(FEVP)*UFLX, ABS(FBTM)*UFLX, ABS(FCAL)*UFLX
      WRITE(lcmf20,1004) THRS, PEVP, PPEN, PTTL
*
620   CONTINUE
*
*     END 20 micoliters ************************************************
```

```
*
*       10 microliters ******************************************************
*
6209    IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
        VOL = 10.0D-3
        DOSE = (VOL/AREA)*RHET
        IF (NFDEPVR .EQ. 0) THEN
        FDEP = CNFDEP
        ELSE
        CALL DEPDEP (VOL, FDEP)
        END IF
        CORR = FDEP*SCDR*ETCSAT/RHET
        VHTH = VOL/AREA - CORR
        SCTH = SCDR + CORR
*
        NDPD = ((NPTS-1)*FDEP) + 1
        WRITE(lsim,*) '******************************************************'
        WRITE(lsim,*) 'Initial Volume of Penetrant (in microlit) = ', VOL*1.0D3
        WRITE(lsim,*) 'Fractional Deposition Depth (%) =', FDEP*100.0D0
        WRITE(lsim,*) 'Number of Permeant Deposited Mesh Points = ', NDPD
        WRITE(lsim,*) 'Initial VH Thickness (in microns) = ', VHTH*1.0D4
        WRITE(lsim,*) 'Thickness of Soaked SC (in microns) = ', SCTH*1.0D4
        WRITE(lsim,*) 'Initial Mass of Penetrant (microg/cm2) = ', DOSE*1.0D6
*
        DO 610 IT610 = 1, 5000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 6101 KL = 1,NPDE
                DO 6102 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
6102            CONTINUE
6101    CONTINUE
*
```

```
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VETH/DBLE(NPTS-1)
      FCAL = RHVE*DAVE*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
      WRITE(llvt10,1004) THRS, VHTI
*
      ELSE
*
      ETTP = SOL(NPTS,1)
      CALL DENSTY (ETTP, RHET, RHSC, RHAV)
      FEVP = FLXG*ETTP/ETSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
      CALL MASSBL (NPTS, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
      CFLX = CFLX + FTTL*TDIF ! Cumulative Flux
      CHECK = (TMAS + CFLX)/DOSE ! Should be ~ 1
      WRITE(lmbc10,1004) THRS, CHECK
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 6103 NN = 21, 30
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
6103  CONTINUE
*
      TOLD = T
```

```
      IF (THRS .GT. 240.0D0) THEN
      GO TO 6109
      ELSE
      END IF
      PPENOLD = PPEN
*
*     Writing the Solutions!
      WRITE(lAsc10,1001) THRS
      WRITE(lAsc10,1002) (SOL(J,1), J=1, NPTS)
      WRITE(lAve10,1001) THRS
      WRITE(lAve10,1002) (SOL(J,2), J=1, NPTS)
      WRITE(lflx10,1004) THRS, ABS(FEVP)*UFLX, ABS(FBTM)*UFLX, ABS(FCAL)*UFLX
      WRITE(lcmf10,1004) THRS, PEVP, PPEN, PTTL
*
610   CONTINUE
*
*     END 10 micoliters ***************************************************
*
*     5 microliters *******************************************************
*
6109  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 5.0D-3
      DOSE = (VOL/AREA)*RHET
      IF (NFDEPVR .EQ. 0) THEN
      FDEP = CNFDEP
      ELSE
      CALL DEPDEP (VOL, FDEP)
      END IF
      CORR = FDEP*SCDR*ETCSAT/RHET
      VHTH = VOL/AREA - CORR
      SCTH = SCDR + CORR
*
      NDPD = ((NPTS-1)*FDEP) + 1
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Initial Volume of Penetrant (in microlit) = ', VOL*1.0D3
      WRITE(lsim,*) 'Fractional Deposition Depth (%) =', FDEP*100.0D0
      WRITE(lsim,*) 'Number of Permeant Deposited Mesh Points = ', NDPD
      WRITE(lsim,*) 'Initial VH Thickness (in microns) = ', VHTH*1.0D4
      WRITE(lsim,*) 'Thickness of Soaked SC (in microns) = ', SCTH*1.0D4
      WRITE(lsim,*) 'Initial Mass of Penetrant (microg/cm2) = ', DOSE*1.0D6
*
      DO 65 IT65 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
```

```
*
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 651 KL = 1,NPDE
                DO 652 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
652             CONTINUE
651     CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VETH/DBLE(NPTS-1)
        FCAL = RHVE*DAVE*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHET
        WRITE(llvt5,1004) THRS, VHTI
*
        ELSE
*
        ETTP = SOL(NPTS,1)
        CALL DENSTY (ETTP, RHET, RHSC, RHAV)
        FEVP = FLXG*ETTP/ETSCIN
!       FEVP = FLXG*SOL(NPTS,1)
        TMVH = 0.0D0
        END IF
*
        CALL MASSBL (NPTS, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
        CFLX = CFLX + FTTL*TDIF ! Cumulative Flux
        CHECK = (TMAS + CFLX)/DOSE ! Should be ~ 1
        WRITE(lmbc5,1004) THRS, CHECK
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
```

```
      PTTL = PEVP + PPEN
*
            DO 653 NN = 31, 40
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
653   CONTINUE
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      CHISQR = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
      write (lsim,*) 'NFREE =', NFREE
      WRITE (lsim,*) 'FINAL CHISQ (*10^5) =', CHISQR*1.0D5
      WRITE (lsim,*) 'FINAL SSR =', CHISQR*DBLE(NFREE)
      GO TO 6509
      ELSE
      END IF
      PPENOLD = PPEN
*
*     Writing the Solutions!
      WRITE(lAsc5,1001) THRS
      WRITE(lAsc5,1002) (SOL(J,1), J=1, NPTS)
      WRITE(lAve5,1001) THRS
      WRITE(lAve5,1002) (SOL(J,2), J=1, NPTS)
      WRITE(lflx5,1004) THRS, ABS(FEVP)*UFLX, ABS(FBTM)*UFLX, ABS(FCAL)*UFLX
      WRITE(lcmf5,1004) THRS, PEVP, PPEN, PTTL
*
65          CONTINUE
************************************************************************
*     END Final Call to D03PPF
************************************************************************
*
*
1001  FORMAT('Time (s) ', F24.12)
1002  FORMAT(5(F18.12,1X))
1003  FORMAT(F36.12, 2X, F18.12, 2X, F18.12)
1004  FORMAT(F36.12, 2X, 6(F36.12))
*
6509  CLOSE(lsim, STATUS='KEEP')
            CLOSE(lAsc40, STATUS='KEEP')
            CLOSE(lAsc20, STATUS='KEEP')
            CLOSE(lAsc10, STATUS='KEEP')
            CLOSE(lAsc5, STATUS='KEEP')
            CLOSE(lAve40, STATUS='KEEP')
            CLOSE(lAve20, STATUS='KEEP')
```

```
      CLOSE(lAve10, STATUS='KEEP')
      CLOSE(lAve5, STATUS='KEEP')
      CLOSE(llvt40, STATUS='KEEP')
      CLOSE(llvt20, STATUS='KEEP')
      CLOSE(llvt10, STATUS='KEEP')
      CLOSE(llvt5, STATUS='KEEP')
      CLOSE(lflx40, STATUS='KEEP')
      CLOSE(lflx20, STATUS='KEEP')
      CLOSE(lflx10, STATUS='KEEP')
      CLOSE(lflx5, STATUS='KEEP')
      CLOSE(lmbc40, STATUS='KEEP')
      CLOSE(lmbc20, STATUS='KEEP')
      CLOSE(lmbc10, STATUS='KEEP')
      CLOSE(lmbc5, STATUS='KEEP')
      CLOSE(lcmf40, STATUS='KEEP')
      CLOSE(lcmf20, STATUS='KEEP')
      CLOSE(lcmf10, STATUS='KEEP')
      CLOSE(lcmf5, STATUS='KEEP')
!     CLOSE(lext, STATUS='KEEP')
*
      STOP
      END
*
*
*
```

Program BENZGRIDALLDOSE simulates the evaporation and absorption of benzene through skin sub layers and employs a GRID SEARCH optimization algorithm that fits the simulated absorption profile with the experimental data on skin permeation benzene. It fits absorption data for all doses and results in a optimal value for constant SC diffusivity. This program can be used for ethanol by changing the physicochemical properties of and transport parameters.

```
PROGRAM BENZGRIDALLDOSE
*
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      IMPLICIT INTEGER (I-N)
*
*     NPDE   = Number of PDEs
*     NCODE  = Number of Coupled ODEs
*     M      = Determines the Coordinate system (0 = Rectangular Cartesian)
*     NPTS   = Number of Mesh Points
*     NXI    = Number of PDE-ODE Coupling Points
*     NXFIX  = Number of Fixed Mesh Points
      PARAMETER (NPDE=2, NCODE=1, M=0, NPTS=201, NXI=0, NXFIX=0)
*
*     DO NOT CHANGE THE LINES BELOW!
      PARAMETER (NWKRES = NPDE*(3*NPDE+6*NXI+NPTS+15)+ NXI+NCODE
     $                  + 7*NPTS+NXFIX+2+9000)
!     Increase the value 9000 if more workspace is needed
      PARAMETER (NEQN=NPDE*NPTS+NCODE)
      PARAMETER (NIW=25*NEQN+25+NXFIX+90000)
!     Dimension of the array IW. Pg. 17. DO NOT CHANGE unless IFAIL 15!
      PARAMETER (LENODE=(6+5)*NEQN+50) ! For theta method, Pg. 17!
      PARAMETER (NW=4*NEQN+11*NEQN/2+1+NWKRES+LENODE+20000)
!     DO NOT CHANGE unless IFAIL 15 !
*
      PARAMETER (NFIT = 0) ! Keep a non-zero value for optimization
      PARAMETER (NEXDP = 40)
      PARAMETER (NPARAM = 1)
      PARAMETER (NDEP = 0)
      PARAMETER (NDIF = 1)
      PARAMETER (MODE = 0)
*
      DOUBLE PRECISION U(NEQN), X(NEQN), RTOL(NEQN), ATOL(NEQN), XI(1),
     +                 XFIX(1), IW(NIW), W(NW), ALGOPT(30), SOL(NPTS,NPDE),
     +                 XDATA(NEXDP), YDATA(NEXDP), YFIT(NEXDP),
     +                 SIGMAY(NEXDP), A(10), DELTAA(10), SIGMAA(10)
*     IW(NIW) - Output variable, W(NW) - Workspace variable
      INTEGER ITIME(7)
*
      LOGICAL REMESH, THETA
      CHARACTER*1 LAOPT, NORM
*
*     External Subroutines
      EXTERNAL UVINIT,ODEDEF,PDEDEF,BNDARY,D03PPF,D03PCL,D03PCK,X05AAF
!     D03PCL replaces MONITF as dummy subroutine!
*
```

```
*       Common Blocks
        COMMON /EXPMT/ TEMP, DOSE, VHTH, BZSCIN
        COMMON /VHPAR/ RHBZ, WMBZ
        COMMON /SCPAR/ RHSC, SCTH, DASC, FDEP, CORR
        COMMON /VEPAR/ RHVT, VTTH, DAVT
        COMMON /INTER/ FLXG, FLSCVH, FLVESC, FBTM, SCVHPC, VTSCPC, PTOR
        COMMON /VPCPH/ VPCA, VPCB, VPCC, VPCD, VPCE
        COMMON /CRRCT/ FCOR, DCOR
        COMMON /VALUE/ DAVE, VESCPC, DFREE, FUBZ, SCWRPC, VTWRPC
*
*       Initialize the constants specific to the system of interest
        CALL CONSTS
*
*       Write the Solution Files
        lsim = 11 ! Description of the Simulation
        lAsc40 = 21 ! Mass Fraction of Solvent in the Stratum Corneum !
        lAsc20 = 22 ! Mass Fraction of Solvent in the Stratum Corneum !
        lAsc10 = 23 ! Mass Fraction of Solvent in the Stratum Corneum !
        lAsc5  = 24 ! Mass Fraction of Solvent in the Stratum Corneum !
        lAve40 = 31 ! Mass Fraction of Solvent in the Viable Epidermis !
        lAve20 = 32 ! Mass Fraction of Solvent in the Viable Epidermis !
        lAve10 = 33 ! Mass Fraction of Solvent in the Viable Epidermis !
        lAve5  = 34 ! Mass Fraction of Solvent in the Viable Epidermis !
        llvt40 = 41 ! Instantaneous Thickness of the Donor Solution !
        llvt20 = 42 ! Instantaneous Thickness of the Donor Solution !
        llvt10 = 43 ! Instantaneous Thickness of the Donor Solution !
        llvt5  = 44 ! Instantaneous Thickness of the Donor Solution !
        lflx40 = 51 ! Fluxes
        lflx20 = 52
        lflx10 = 53
        lflx5  = 54
        lmbc40 = 61 ! Mass balance component
        lmbc20 = 62 ! Mass balance component
        lmbc10 = 63 ! Mass balance component
        lmbc5  = 64 ! Mass balance component
        lcmf40 = 71 ! Cumulative Fluxes
        lcmf20 = 72 ! Cumulative Fluxes
        lcmf10 = 73 ! Cumulative Fluxes
        lcmf5  = 74 ! Cumulative Fluxes
        lext = 18 ! Anciliary data
*
        itrace = 2 ! See D03PPF Manual, Pg. 18 !
*
        OPEN (lsim, file='sim.dat', status='new')
        OPEN (lAsc40, file='Asc40.dat', status='new')
        OPEN (lAsc20, file='Asc20.dat', status='new')
        OPEN (lAsc10, file='Asc10.dat', status='new')
        OPEN (lAsc5, file='Asc5.dat', status='new')
        OPEN (lAve40, file='Ave40.dat', status='new')
        OPEN (lAve20, file='Ave20.dat', status='new')
        OPEN (lAve10, file='Ave10.dat', status='new')
        OPEN (lAve5, file='Ave5.dat', status='new')
        OPEN (llvt40, file='lvt40.dat', status='new')
```

```
      OPEN (llvt20, file='lvt20.dat', status='new')
      OPEN (llvt10, file='lvt10.dat', status='new')
      OPEN (llvt5, file='lvt5.dat', status='new')
      OPEN (lflx40, file='flx40.dat', status='new')
      OPEN (lflx20, file='flx20.dat', status='new')
      OPEN (lflx10, file='flx10.dat', status='new')
      OPEN (lflx5, file='flx5.dat', status='new')
      OPEN (lmbc40, file='mbc40.dat', status='new')
      OPEN (lmbc20, file='mbc20.dat', status='new')
      OPEN (lmbc10, file='mbc10.dat', status='new')
      OPEN (lmbc5, file='mbc5.dat', status='new')
      OPEN (lcmf40, file='cmf40.dat', status='new')
      OPEN (lcmf20, file='cmf20.dat', status='new')
      OPEN (lcmf10, file='cmf10.dat', status='new')
      OPEN (lcmf5, file='cmf5.dat', status='new')
!     OPEN (lext, file='fit.dat', status='new')
*
*
*     Experimental Data
      UFLX = 1.0D6*3600.0D0
      AREA = 0.79D0
*
      XDATA(1)  = 0.08D0*3600.0D0
      XDATA(2)  = 0.17D0*3600.0D0
      XDATA(3)  = 0.25D0*3600.0D0
      XDATA(4)  = 0.33D0*3600.0D0
      XDATA(5)  = 0.67D0*3600.0D0
      XDATA(6)  = 1.00D0*3600.0D0
      XDATA(7)  = 2.00D0*3600.0D0
      XDATA(8)  = 4.00D0*3600.0D0
      XDATA(9)  = 8.50D0*3600.0D0
      XDATA(10) = 24.00D0*3600.0D0
*
      DO 101 I = 1,10
      XDATA(I+10) = XDATA(I)
      XDATA(I+20) = XDATA(I)
      XDATA(I+30) = XDATA(I)
101   CONTINUE
*
*
*     40 microliters
      YDATA(1)  = 0.00D0
      YDATA(2)  = 0.01D0
      YDATA(3)  = 0.02D0
      YDATA(4)  = 0.02D0
      YDATA(5)  = 0.03D0
      YDATA(6)  = 0.04D0
      YDATA(7)  = 0.04D0
      YDATA(8)  = 0.04D0
      YDATA(9)  = 0.04D0
      YDATA(10) = 0.05D0
*
*     20 microliters
```

```
      YDATA(11) = 0.00D0
      YDATA(12) = 0.01D0
      YDATA(13) = 0.02D0
      YDATA(14) = 0.02D0
      YDATA(15) = 0.03D0
      YDATA(16) = 0.04D0
      YDATA(17) = 0.05D0
      YDATA(18) = 0.05D0
      YDATA(19) = 0.05D0
      YDATA(20) = 0.05D0
*
*     10 microliters
      YDATA(21) = 0.00D0
      YDATA(22) = 0.01D0
      YDATA(23) = 0.02D0
      YDATA(24) = 0.02D0
      YDATA(25) = 0.03D0
      YDATA(26) = 0.04D0
      YDATA(27) = 0.04D0
      YDATA(28) = 0.05D0
      YDATA(29) = 0.05D0
      YDATA(30) = 0.05D0
*
*     5 microliters
      YDATA(31) = 0.01D0
      YDATA(32) = 0.02D0
      YDATA(33) = 0.02D0
      YDATA(34) = 0.03D0
      YDATA(35) = 0.04D0
      YDATA(36) = 0.05D0
      YDATA(37) = 0.05D0
      YDATA(38) = 0.06D0
      YDATA(39) = 0.06D0
      YDATA(40) = 0.06D0
*
*
      WRITE(lsim,*) ' 3-Layer Pure Penetrant Isothermal Simulation '
      WRITE(lsim,*) ' The Penetrant in BENZENE'
*
      WRITE(lsim,*) 'Number of Mesh Points = ', NPTS
      NDPD = ((NPTS-1)*FDEP) + 1
      WRITE(lsim,*) 'Number of Permeant Deposited Mesh Points = ', NDPD
*
      WRITE(lsim,*) '*************************************************'
      WRITE(lsim,*) 'Molecular Weight of the Penetrant = ', WMBZ
      WRITE(lsim,*) 'Constant System Temperature = ', TEMP - 273.15D0
      WRITE(lsim,*) 'Density of Penetrant = ', RHBZ
*
      WRITE(lsim,*) '*************************************************'
      WRITE(lsim,*) 'Density of SC Phase = ', RHSC
      WRITE(lsim,*) 'Thickness of SC Phase (in microns) = ', SCTH*1.0D4
      WRITE(lsim,*) 'Corrected Diffusivity of A in SC = ', DASC
      WRITE(lsim,*) 'Initial Correction Factor for DGBK = ', DCOR
```

```
      WRITE(lsim,*) 'Predicted Diffusivity of A in SC = ', DASC/DCOR
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Density of VT Phase = ', RHVT
      WRITE(lsim,*) 'Thickness of VT Phase (in microns) = ', VTTH*1.0D4
      WRITE(lsim,*) 'Free Diffusivity in VT = ', DFREE
      WRITE(lsim,*) 'PROTEIN BINDING FACTOR (FSUBU) = ', FUBZ
      WRITE(lsim,*) 'Diffusivity of A in VT = ', DAVT
!     WRITE(lsim,*) 'Diffusivity of A in VT (old) = ', DAVE
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Vapor Pressure in Torr = ', PTOR
      WRITE(lsim,*) 'Corrected KEVAP = ', FLXG*3600.0D0/RHBZ
      WRITE(lsim,*) 'Initial Correction Factor for FLXG = ', FCOR
      WRITE(lsim,*) 'Predicted KEVAP = ', FLXG*3600.0D0/(FCOR*RHBZ)
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Fractional Deposition Depth (%)= ', FDEP*100.0D0
      WRITE(lsim,*) 'Thickness Correction due to Solvent Deposition = ', CORR*1.0D4
      WRITE(lsim,*) 'SC-VH Partition Coefficient of Benzene = ', SCVHPC
      WRITE(lsim,*) 'VT-SC Partition Coefficient of Benzene = ', VTSCPC
      WRITE(lsim,*) 'SC-Water Partition Coefficient of Benzene = ', SCWRPC
      WRITE(lsim,*) 'VT-Water Partition Coefficient of Benzene = ', VTWRPC
!     WRITE(lsim,*) 'VT-SC Partition Coefficient of Benzene (old) = ', VESCPC
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Number of Fitted Parameters = ', NPARAM
      WRITE(lsim,*) 'Weighting Coefficient = ', MODE
      WRITE(lsim,*) 'Fitting the Deposition Depth = ', NDEP
      WRITE(lsim,*) 'Fitting the SC Diffusivity = ', NDIF
*
*     Initialize Mesh - Uniform Mesh
*
      DO 910 I = 1, NPTS
            X(I) = DBLE(I-1)/(NPTS-1.0D0)
910   CONTINUE
*
*     Set Remesh Parameters
*
      REMESH = .false.
      NRMESH = 0
      DXMESH = 0.0D0
      TRMESH = 1.0D0
      IPMINF = 0
      XRATIO = 1.5D0
      CONST  = 2.0D0/(NPTS-1)
      IND = 0
      ITASK = 2
      NORM = 'A'
      LAOPT= 'S' !use the sparse matrix routines
      THETA = .false.
*
      DO 920 I=1,30
```

```
            ALGOPT(I)=0.0D0
920     CONTINUE
        ALGOPT(4)=2.0D0 !Does not perform the Petzold test
        ALGOPT(29)=0.5 !NAG Deafault = 0.1
        ALGOPT(30)=1.0D-300 !NAG Default 1.0D-2!
*
*       Set Time Step & Convergence Criterion
*
        DELT = 1.0D-3
        WRITE(lsim,*) '*****************************************************'
        WRITE(lsim,*) 'Time Increament Step Size =', DELT
        DO 930 K = 1, NEQN
            RTOL(K) = 1.0D-8
            ATOL(K) = 1.0D-8
930     CONTINUE
        ITOL = 4
        WRITE(lsim,*) 'RTOL = ', RTOL(1)
        WRITE(lsim,*) 'ATOL = ', ATOL(1)
        WRITE(lsim,*) 'ITOL = ', ITOL
*
*       Write Simulation Date and Time
        CALL X05AAF(ITIME)
        WRITE(lsim,*) 'Month / Day / Year = ', ITIME(2),'/',ITIME(3),'/',ITIME(1)
        WRITE(lsim,*) 'Hour / Minute / Second = ', ITIME(4),'/',ITIME(5),'/',ITIME(6)
*
*
        FLXGOLD = FLXG
        DASCOLD = DASC
        FDEPOLD = FDEP
*
        NFREE = NEXDP - NPARAM
*
        IF (NFIT .EQ. 0) THEN
        GO TO 990
        ELSE
        WRITE (lsim,*) 'RUNNING THE GRID SEARCH OPTIMIZER'
        END IF
*
        IF (NPARAM .EQ. 3) THEN
            IF (NDEP .EQ. 1) THEN
            A(1) = FDEP
                IF (NDIF .EQ. 2) THEN
                A(2) = DASC
                A(3) = FLXG
                ELSE
                A(2) = FLXG
                A(3) = DASC
                END IF
            ELSE IF (NDEP .EQ. 2) THEN
            A(2) = FDEP
                IF (NDIF .EQ. 1) THEN
                A(1) = DASC
                A(3) = FLXG
```

```
                ELSE
                A(1) = FLXG
                A(3) = DASC
                END IF
        ELSE IF (NDEP .EQ. 3) THEN
        A(3) = FDEP
                IF (NDIF .EQ. 1) THEN
                A(1) = DASC
                A(2) = FLXG
                ELSE
                A(1) = FLXG
                A(2) = DASC
                END IF
        END IF
    ELSE IF (NPARAM .EQ. 2) THEN
        IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                A(1) = DASC
                A(2) = FLXG
                ELSE
                A(1) = FLXG
                A(2) = DASC
                END IF
        ELSE IF (NDEP .EQ. 1) THEN
                A(1) = FDEP
                IF (NDIF .EQ. 2) THEN
                A(2) = DASC
                ELSE
                A(2) = FLXG
                END IF
        ELSE IF (NDEP .EQ. 2) THEN
                IF (NDIF .EQ. 1) THEN
                A(1) = DASC
                ELSE
                A(1) = FLXG
                END IF
                A(2) = FDEP
        END IF
    ELSE IF (NPARAM .EQ. 1) THEN
        IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                A(1) = DASC
                ELSE
                A(1) = FLXG
                END IF
        ELSE
                A(1) = FDEP
        END IF
    END IF
*
    DO 940 LL = 1,NPARAM
    DELTAA(LL) = 0.1D0*A(LL)
940 CONTINUE
```

```
      DO 941 LM = 1,NEXDP
      SIGMAY(LM) = 1.0D0
941   CONTINUE
*
*
*     Optimization with GRID SEARCH
      CHITOL = 1.0D-1
      write(lsim,*) 'CHITOL =', CHITOL
      CHIOLD = 0.0D0
*
      DO 998 NOPT = 1, 1000
      write(lsim,*) 'NOPT =', NOPT
*
      DO 90 J = 1, NPARAM
*
*     EVALUATE CHI SQUARE AT FIRST TWO SEARCH POINTS
*
*************************************************************************
*     1st Call to D03PPF
*************************************************************************
*
      IF (NPARAM .EQ. 3) THEN
            IF (NDEP .EQ. 1) THEN
            FDEP = A(1)
                  IF (NDIF .EQ. 2) THEN
                  DASC = A(2)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(2)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 2) THEN
            FDEP = A(2)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(1)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 3) THEN
            FDEP = A(3)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(2)
                  ELSE
                  FLXG = A(1)
                  DASC = A(2)
                  END IF
            END IF
      ELSE IF (NPARAM .EQ. 2) THEN
            IF (NDEP .EQ. 0) THEN
                  IF (NDIF .EQ. 1) THEN
```

```
                DASC = A(1)
                FLXG = A(2)
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
        ELSE IF (NDEP .EQ. 1) THEN
                FDEP = A(1)
                IF (NDIF .EQ. 2) THEN
                DASC = A(2)
                ELSE
                FLXG = A(2)
                END IF
        ELSE IF (NDEP .EQ. 2) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
                FDEP = A(2)
        END IF
    ELSE IF (NPARAM .EQ. 1) THEN
        IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
        ELSE
                FDEP = A(1)
        END IF
    END IF
*
*       40 microliters ******************************************************
*
    IND = 0
    T = 0.0D0
    TOUT = 0.0D0
    TOLD = 0.0D0
    CFLX = 0.0D0
    CEVP = 0.0D0
    CPEN = 0.0D0
    PPENOLD = 0.0D0
*
    VOL = 40.0D-3
    VHTH = VOL/AREA - CORR
    DOSE = VOL*RHBZ/AREA
*
    DO 140 IT140 = 1, 5000
    TOUT = T + DELT
*
    IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
```

```
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 1401 KL = 1,NPDE
                DO 1402 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
1402            CONTINUE
1401    CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHBZ
*
        ELSE
*
        BZTP = SOL(NPTS,1)
        CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
        FEVP = FLXG*BZTP/BZSCIN
!       FEVP = FLXG*SOL(NPTS,1)
        TMVH = 0.0D0
        END IF
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 1403 NN = 1, 10
        IF (T .LT. XDATA(NN)) THEN
        ELSE IF (T .EQ. XDATA(NN)) THEN
        YFIT(NN) = PPEN
        ELSE
                IF (TOLD .LT. XDATA(NN)) THEN
```

```
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
1403  CONTINUE
*
!     write(lsim,*) 'yfit(10), 1st pass =', yfit(10), 'ydata(10), 1st pass =', ydata(10)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 1409
      ELSE
      END IF
      PPENOLD = PPEN
*
140   CONTINUE
*
*     END 40 micoliters ***************************************************
*
*     20 microliters ******************************************************
*
1409  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 20.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 120 IT120 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 1201 KL = 1,NPDE
            DO 1202 KM = 1,NPTS
```

```
                    SOL(KM,KL) = U(NPDE*(KM-1)+KL)
1202          CONTINUE
1201  CONTINUE
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 1203 NN = 11, 20
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
1203  CONTINUE
*
!     write(lsim,*) 'yfit(20), 1st pass =', yfit(20), 'ydata(20), 1st pass =', ydata(20)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 1209
```

```
      ELSE
      END IF
      PPENOLD = PPEN
*
120   CONTINUE
*
*     END 20 micoliters ************************************************
*
*     10 microliters ***************************************************
*
1209  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 10.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 110 IT110 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 1101 KL = 1,NPDE
            DO 1102 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
1102        CONTINUE
1101  CONTINUE
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
```

```
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 1103 NN = 21, 30
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
1103  CONTINUE
*
!     write(lsim,*) 'yfit(30), 1st pass =', yfit(30), 'ydata(30), 1st pass =', ydata(30)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 1109
      ELSE
      END IF
      PPENOLD = PPEN
*
110   CONTINUE
*
*     END 10 micoliters ************************************************
*
*     5 microliters ****************************************************
*
1109  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
```

```
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 5.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 15 IT15 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 151 KL = 1,NPDE
            DO 152 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
152         CONTINUE
151   CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
```

```
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 153 NN = 31, 40
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
153   CONTINUE
*
!     write(lsim,*) 'yfit(40), 1st pass =', yfit(40), 'ydata(40), 1st pass =', ydata(40)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 159
      ELSE
      END IF
      PPENOLD = PPEN
*
*
15    CONTINUE
**********************************************************************
*     END 1st Call to D03PPF
**********************************************************************
*
159   CHISQ1 = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!     write(lsim,*) 'end 1st NAG time looping'
      FN  = 0.0D0
      DELTA = DELTAA(J)
41    A(J) = A(J) + DELTA
*
**********************************************************************
*     2nd Call to D03PPF
**********************************************************************
*
            DO 2020 MM = 1,NEXDP
            YFIT(MM) = 0.0D0
2020  CONTINUE
*
```

```
*
      IF (NPARAM .EQ. 3) THEN
            IF (NDEP .EQ. 1) THEN
            FDEP = A(1)
                  IF (NDIF .EQ. 2) THEN
                  DASC = A(2)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(2)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 2) THEN
            FDEP = A(2)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(1)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 3) THEN
            FDEP = A(3)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(2)
                  ELSE
                  FLXG = A(1)
                  DASC = A(2)
                  END IF
            END IF
      ELSE IF (NPARAM .EQ. 2) THEN
            IF (NDEP .EQ. 0) THEN
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(2)
                  ELSE
                  FLXG = A(1)
                  DASC = A(2)
                  END IF
            ELSE IF (NDEP .EQ. 1) THEN
                  FDEP = A(1)
                  IF (NDIF .EQ. 2) THEN
                  DASC = A(2)
                  ELSE
                  FLXG = A(2)
                  END IF
            ELSE IF (NDEP .EQ. 2) THEN
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  ELSE
                  FLXG = A(1)
                  END IF
                  FDEP = A(2)
```

```
            END IF
        ELSE IF (NPARAM .EQ. 1) THEN
            IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
            ELSE
                FDEP = A(1)
            END IF
        END IF
*
*       40 microliters ******************************************************
*
        IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
        VOL = 40.0D-3
        VHTH = VOL/AREA - CORR
        DOSE = VOL*RHBZ/AREA
*
        DO 240 IT240 = 1, 5000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $              ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $              REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $              CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 2401 KL = 1,NPDE
            DO 2402 KM = 1,NPTS
                SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2402        CONTINUE
2401    CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
```

```
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 2403 NN = 1, 10
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
2403  CONTINUE
*
!     write(lsim,*) 'yfit(10), 2nd pass =', yfit(10), 'ydata(4), 2nd pass =', ydata(10)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 2409
      ELSE
      END IF
      PPENOLD = PPEN
*
240   CONTINUE
*
```

```
*       END 40 micoliters ***************************************************
*
*       20 microliters ******************************************************
*
2409    IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
        VOL = 20.0D-3
        VHTH = VOL/AREA - CORR
        DOSE = VOL*RHBZ/AREA
*
        DO 220 IT220 = 1, 5000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 2201 KL = 1,NPDE
                DO 2202 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2202            CONTINUE
2201    CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHBZ
```

```
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 2203 NN = 11, 20
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
2203  CONTINUE
*
!     write(lsim,*) 'yfit(20), 2nd pass =', yfit(20), 'ydata(20), 2nd pass =', ydata(20)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 2209
      ELSE
      END IF
      PPENOLD = PPEN
*
220   CONTINUE
*
*     END 20 micoliters ************************************************
*
*     10 microliters ***************************************************
*
2209  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
```

```
      PPENOLD = 0.0D0
*
      VOL = 10.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 210 IT210 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 2101 KL = 1,NPDE
            DO 2102 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2102        CONTINUE
2101  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
```

```
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 2103 NN = 21, 30
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
2103  CONTINUE
*
!     write(lsim,*) 'yfit(30), 2nd pass =', yfit(30), 'ydata(30), 2nd pass =', ydata(30)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 2109
      ELSE
      END IF
      PPENOLD = PPEN
*
210   CONTINUE
*
*     END 10 micoliters ****************************************************
*
*     5 microliters ********************************************************
*
2109  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 5.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 25 IT25 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
```

```
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 251 KL = 1,NPDE
                DO 252 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
252             CONTINUE
251     CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHBZ
*
        ELSE
*
        BZTP = SOL(NPTS,1)
        CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
        FEVP = FLXG*BZTP/BZSCIN
!       FEVP = FLXG*SOL(NPTS,1)
        TMVH = 0.0D0
        END IF
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 253 NN = 31, 40
        IF (T .LT. XDATA(NN)) THEN
        ELSE IF (T .EQ. XDATA(NN)) THEN
        YFIT(NN) = PPEN
        ELSE
```

```
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
253   CONTINUE
*
!     write(lsim,*) 'yfit(40), 2nd pass =', yfit(40), 'ydata(40), 2nd pass =', ydata(40)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 259
      ELSE
      END IF
      PPENOLD = PPEN
*
*
25    CONTINUE
************************************************************************
*     END 2nd Call to D03PPF
************************************************************************
*
259   CHISQ2 = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!     write(lsim,*) 'end 2nd NAG time looping'
      write(lsim,*) 'chisq2, 2nd pass', chisq2, 'j =', j
*
      IF (CHISQ1 .LT. CHISQ2) THEN
*
*     CHI SQUARE IS INCREASING, REVERSE DIRECTION OF SEARCH
*
      DELTA = -DELTA
      A(J) = A(J) + DELTA
*
************************************************************************
*     3rd Call to D03PPF
************************************************************************
*
            DO 2030 MM = 1,NEXDP
            YFIT(MM) = 0.0D0
2030  CONTINUE
*
*
      IF (NPARAM .EQ. 3) THEN
            IF (NDEP .EQ. 1) THEN
            FDEP = A(1)
                  IF (NDIF .EQ. 2) THEN
                  DASC = A(2)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(2)
                  DASC = A(3)
```

```
            END IF
        ELSE IF (NDEP .EQ. 2) THEN
        FDEP = A(2)
            IF (NDIF .EQ. 1) THEN
            DASC = A(1)
            FLXG = A(3)
            ELSE
            FLXG = A(1)
            DASC = A(3)
            END IF
        ELSE IF (NDEP .EQ. 3) THEN
        FDEP = A(3)
            IF (NDIF .EQ. 1) THEN
            DASC = A(1)
            FLXG = A(2)
            ELSE
            FLXG = A(1)
            DASC = A(2)
            END IF
        END IF
ELSE IF (NPARAM .EQ. 2) THEN
        IF (NDEP .EQ. 0) THEN
            IF (NDIF .EQ. 1) THEN
            DASC = A(1)
            FLXG = A(2)
            ELSE
            FLXG = A(1)
            DASC = A(2)
            END IF
        ELSE IF (NDEP .EQ. 1) THEN
            FDEP = A(1)
            IF (NDIF .EQ. 2) THEN
            DASC = A(2)
            ELSE
            FLXG = A(2)
            END IF
        ELSE IF (NDEP .EQ. 2) THEN
            IF (NDIF .EQ. 1) THEN
            DASC = A(1)
            ELSE
            FLXG = A(1)
            END IF
            FDEP = A(2)
        END IF
ELSE IF (NPARAM .EQ. 1) THEN
        IF (NDEP .EQ. 0) THEN
            IF (NDIF .EQ. 1) THEN
            DASC = A(1)
            ELSE
            FLXG = A(1)
            END IF
        ELSE
            FDEP = A(1)
```

```
            END IF
        END IF
*
*       40 microliters ******************************************************
*
        IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
        VOL = 40.0D-3
        VHTH = VOL/AREA - CORR
        DOSE = VOL*RHBZ/AREA
*
        DO 340 IT340 = 1, 5000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 3401 KL = 1,NPDE
                DO 3402 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
3402            CONTINUE
3401    CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
```

```
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 3403 NN = 1, 10
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
3403  CONTINUE
*
!     write(lsim,*) 'yfit(10), 3rd pass =', yfit(10), 'ydata(10), 3rd pass =', ydata(10)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 3409
      ELSE
      END IF
      PPENOLD = PPEN
*
340   CONTINUE
*
*     END 40 micoliters ************************************************
*
*     20 microliters ***************************************************
*
3409  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
```

```
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 20.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 320 IT320 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 3201 KL = 1,NPDE
            DO 3202 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
3202        CONTINUE
3201  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
```

```
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
               DO 3203 NN = 11, 20
        IF (T .LT. XDATA(NN)) THEN
        ELSE IF (T .EQ. XDATA(NN)) THEN
        YFIT(NN) = PPEN
        ELSE
               IF (TOLD .LT. XDATA(NN)) THEN
               SLP = (PPEN - PPENOLD)/TDIF
               YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
               ELSE IF (TOLD .EQ. XDATA(NN)) THEN
               YFIT(NN) = PPENOLD
               END IF
        END IF
3203    CONTINUE
*
!       write(lsim,*) 'yfit(20), 3rd pass =', yfit(20), 'ydata(20), 3rd pass =', ydata(20)
*
        TOLD = T
        IF (THRS .GT. 240.0D0) THEN
        GO TO 3209
        ELSE
        END IF
        PPENOLD = PPEN
*
320     CONTINUE
*
*       END 20 micoliters ***************************************************
*
*       10 microliters ******************************************************
*
3209    IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
        VOL = 10.0D-3
        VHTH = VOL/AREA - CORR
        DOSE = VOL*RHBZ/AREA
*
        DO 310 IT310 = 1, 5000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
```

```
*
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 3101 KL = 1,NPDE
                DO 3102 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
3102            CONTINUE
3101    CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHBZ
*
        ELSE
*
        BZTP = SOL(NPTS,1)
        CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
        FEVP = FLXG*BZTP/BZSCIN
!       FEVP = FLXG*SOL(NPTS,1)
        TMVH = 0.0D0
        END IF
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 3103 NN = 21, 30
        IF (T .LT. XDATA(NN)) THEN
        ELSE IF (T .EQ. XDATA(NN)) THEN
        YFIT(NN) = PPEN
```

```
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
3103  CONTINUE
*
!     write(lsim,*) 'yfit(30), 3rd pass =', yfit(30), 'ydata(30), 3rd pass =', ydata(30)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 3109
      ELSE
      END IF
      PPENOLD = PPEN
*
310   CONTINUE
*
*     END 10 micoliters ****************************************************
*
*     5 microliters ********************************************************
*
3109  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 5.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 35 IT35 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
```

```
      DO 351 KL = 1,NPDE
            DO 352 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
352         CONTINUE
351   CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 353 NN = 31, 40
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
353   CONTINUE
*
!     write(lsim,*) 'yfit(40), 3rd pass =', yfit(40), 'ydata(40), 3rd pass =', ydata(40)
```

```
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 359
      ELSE
      END IF
      PPENOLD = PPEN
*
*
35    CONTINUE
*************************************************************************
*     END 3rd Call to D03PPF
*************************************************************************
*
359   SAVE = CHISQ1
      CHISQ1 = CHISQ2
      SAVE = CHISQ2
      ELSE IF (CHISQ1 .EQ. CHISQ2) THEN
      GO TO 41
      ELSE
      END IF
!     write(lsim,*) 'end 3rd NAG time looping'
*
*     INCREASE A(J) UNTIL CHI SQUARE INCREASES
*
61    FN = FN + 1.0D0
      A(J) = A(J) + DELTA
*
*************************************************************************
*     4th Call to D03PPF
*************************************************************************
*
            DO 2040 MM = 1,NEXDP
            YFIT(MM) = 0.0D0
2040  CONTINUE
*
*
      IF (NPARAM .EQ. 3) THEN
            IF (NDEP .EQ. 1) THEN
            FDEP = A(1)
                  IF (NDIF .EQ. 2) THEN
                  DASC = A(2)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(2)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 2) THEN
            FDEP = A(2)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(3)
                  ELSE
```

```
                FLXG = A(1)
                DASC = A(3)
                END IF
        ELSE IF (NDEP .EQ. 3) THEN
        FDEP = A(3)
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                FLXG = A(2)
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
        END IF
    ELSE IF (NPARAM .EQ. 2) THEN
        IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                FLXG = A(2)
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
        ELSE IF (NDEP .EQ. 1) THEN
                FDEP = A(1)
                IF (NDIF .EQ. 2) THEN
                DASC = A(2)
                ELSE
                FLXG = A(2)
                END IF
        ELSE IF (NDEP .EQ. 2) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
                FDEP = A(2)
        END IF
    ELSE IF (NPARAM .EQ. 1) THEN
        IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
        ELSE
                FDEP = A(1)
        END IF
    END IF
*
*       40 microliters ******************************************************
*
      IND = 0
      T = 0.0D0
```

```
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 40.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 440 IT440 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 4401 KL = 1,NPDE
            DO 4402 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
4402        CONTINUE
4401  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
```

```
!       FEVP = FLXG*SOL(NPTS,1)
        TMVH = 0.0D0
        END IF
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 4403 NN = 1, 10
                IF (T .LT. XDATA(NN)) THEN
                ELSE IF (T .EQ. XDATA(NN)) THEN
                YFIT(NN) = PPEN
                ELSE
                        IF (TOLD .LT. XDATA(NN)) THEN
                        SLP = (PPEN - PPENOLD)/TDIF
                        YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
                        ELSE IF (TOLD .EQ. XDATA(NN)) THEN
                        YFIT(NN) = PPENOLD
                        END IF
                END IF
4403    CONTINUE
*
!       write(lsim,*) 'yfit(10), 4th pass =', yfit(10), 'ydata(10), 4th pass =', ydata(10)
*
        TOLD = T
        IF (THRS .GT. 240.0D0) THEN
        GO TO 4409
        ELSE
        END IF
        PPENOLD = PPEN
*
440     CONTINUE
*
*       END 40 micoliters ************************************************
*
*       20 microliters ***************************************************
*
4409    IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
        VOL = 20.0D-3
        VHTH = VOL/AREA - CORR
        DOSE = VOL*RHBZ/AREA
*
```

```
      DO 420 IT420 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 4201 KL = 1,NPDE
            DO 4202 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
4202        CONTINUE
4201  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
```

```
            DO 4203 NN = 11, 20
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
4203  CONTINUE
*
!     write(lsim,*) 'yfit(20), 4th pass =', yfit(20), 'ydata(20), 4th pass =', ydata(20)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 4209
      ELSE
      END IF
      PPENOLD = PPEN
*
420   CONTINUE
*
*     END 20 micoliters ************************************************
*
*     10 microliters ***************************************************
*
4209  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 10.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 410 IT410 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
```

```
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 4101 KL = 1,NPDE
                DO 4102 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
4102            CONTINUE
4101    CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHBZ
*
        ELSE
*
        BZTP = SOL(NPTS,1)
        CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
        FEVP = FLXG*BZTP/BZSCIN
!       FEVP = FLXG*SOL(NPTS,1)
        TMVH = 0.0D0
        END IF
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 4103 NN = 21, 30
        IF (T .LT. XDATA(NN)) THEN
        ELSE IF (T .EQ. XDATA(NN)) THEN
        YFIT(NN) = PPEN
        ELSE
                IF (TOLD .LT. XDATA(NN)) THEN
                SLP = (PPEN - PPENOLD)/TDIF
                YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
                ELSE IF (TOLD .EQ. XDATA(NN)) THEN
                YFIT(NN) = PPENOLD
                END IF
```

```
      END IF
4103  CONTINUE
*
!     write(lsim,*) 'yfit(30), 4th pass =', yfit(30), 'ydata(30), 4th pass =', ydata(30)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 4109
      ELSE
      END IF
      PPENOLD = PPEN
*
410   CONTINUE
*
*     END 10 micoliters ************************************************
*
*     5 microliters ****************************************************
*
4109  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 5.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 45 IT45 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 451 KL = 1,NPDE
            DO 452 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
452         CONTINUE
451   CONTINUE
*
*
```

```
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 453 NN = 31, 40
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
453   CONTINUE
*
!     write(lsim,*) 'yfit(40), 4th pass =', yfit(40), 'ydata(40), 4th pass =', ydata(40)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 459
      ELSE
      END IF
      PPENOLD = PPEN
```

```
*
*
45            CONTINUE
*************************************************************************
*       END 4th Call to D03PPF
*************************************************************************
*
459     CHISQ3 = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!       WRITE(lsim,*) 'end 4th NAG time looping'
        write(lsim,*) 'chisq3, 4th pass', chisq3, 'j =', j
        IF (CHISQ3 .LT. CHISQ2) THEN
        CHISQ1 = CHISQ2
        CHISQ2 = CHISQ3
        GO TO 61
        ELSE
        END IF
*
*       FIND MINIMUM OF PARABOLA DEFINED BY LAST TWO POINTS
*
        DELTA = DELTA*(1.0D0/(1.0D0 + (CHISQ1-CHISQ2)/(CHISQ3-CHISQ2)) + 0.5D0)
        A(J) = A(J) - DELTA
        SIGMAA(J) = DELTAA(J)*SQRT(2.0D0/(DBLE(NFREE)*(CHISQ3 - 2.0D0*CHISQ2
    +                + CHISQ1)))
        DELTAA(J) = DELTAA(J)*FN/3.0D0
90      CONTINUE
*
*************************************************************************
*       5th Call to D03PPF - CHECK
*************************************************************************
*
              DO 2050 MM = 1,NEXDP
              YFIT(MM) = 0.0D0
2050    CONTINUE
*
*
        IF (NPARAM .EQ. 3) THEN
              IF (NDEP .EQ. 1) THEN
              FDEP = A(1)
                    IF (NDIF .EQ. 2) THEN
                    DASC = A(2)
                    FLXG = A(3)
                    ELSE
                    FLXG = A(2)
                    DASC = A(3)
                    END IF
              ELSE IF (NDEP .EQ. 2) THEN
              FDEP = A(2)
                    IF (NDIF .EQ. 1) THEN
                    DASC = A(1)
                    FLXG = A(3)
                    ELSE
                    FLXG = A(1)
                    DASC = A(3)
```

```
                END IF
            ELSE IF (NDEP .EQ. 3) THEN
            FDEP = A(3)
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                FLXG = A(2)
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
            END IF
        ELSE IF (NPARAM .EQ. 2) THEN
            IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                FLXG = A(2)
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
            ELSE IF (NDEP .EQ. 1) THEN
                FDEP = A(1)
                IF (NDIF .EQ. 2) THEN
                DASC = A(2)
                ELSE
                FLXG = A(2)
                END IF
            ELSE IF (NDEP .EQ. 2) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
                FDEP = A(2)
            END IF
        ELSE IF (NPARAM .EQ. 1) THEN
            IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
            ELSE
                FDEP = A(1)
            END IF
        END IF
*
*       40 microliters ******************************************************
*
        IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
```

```
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 40.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 540 IT540 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 5401 KL = 1,NPDE
            DO 5402 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
5402        CONTINUE
5401  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
```

```
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 5403 NN = 1, 10
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
5403  CONTINUE
*
!     write(lsim,*) 'yfit(10), 5th pass =', yfit(10), 'ydata(10), 5th pass =', ydata(10)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 5409
      ELSE
      END IF
      PPENOLD = PPEN
*
540   CONTINUE
*
*     END 40 micoliters ***************************************************
*
*     20 microliters ******************************************************
*
5409  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 20.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 520 IT520 = 1, 5000
      TOUT = T + DELT
```

```
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 5201 KL = 1,NPDE
            DO 5202 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
5202        CONTINUE
5201  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 5203 NN = 11, 20
      IF (T .LT. XDATA(NN)) THEN
```

```
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
5203  CONTINUE
*
!     write(lsim,*) 'yfit(20), 5th pass =', yfit(20), 'ydata(20), 5th pass =', ydata(20)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 5209
      ELSE
      END IF
      PPENOLD = PPEN
*
520   CONTINUE
*
*     END 20 micoliters ************************************************
*
*     10 microliters ***************************************************
*
5209  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 10.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      DO 510 IT510 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
```

```
*
*       Separating the Temperature and Concentration Solutions
        DO 5101 KL = 1,NPDE
                DO 5102 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
5102            CONTINUE
5101    CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHBZ
*
        ELSE
*
        BZTP = SOL(NPTS,1)
        CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
        FEVP = FLXG*BZTP/BZSCIN
!       FEVP = FLXG*SOL(NPTS,1)
        TMVH = 0.0D0
        END IF
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 5103 NN = 21, 30
        IF (T .LT. XDATA(NN)) THEN
        ELSE IF (T .EQ. XDATA(NN)) THEN
        YFIT(NN) = PPEN
        ELSE
                IF (TOLD .LT. XDATA(NN)) THEN
                SLP = (PPEN - PPENOLD)/TDIF
                YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
                ELSE IF (TOLD .EQ. XDATA(NN)) THEN
                YFIT(NN) = PPENOLD
                END IF
        END IF
5103    CONTINUE
```

```
*
!       write(lsim,*) 'yfit(30), 5th pass =', yfit(30), 'ydata(30), 5th pass =', ydata(30)
*
        TOLD = T
        IF (THRS .GT. 240.0D0) THEN
        GO TO 5109
        ELSE
        END IF
        PPENOLD = PPEN
*
510     CONTINUE
*
*       END 10 micoliters ***************************************************
*
*       5 microliters *******************************************************
*
5109    IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
        VOL = 5.0D-3
        VHTH = VOL/AREA - CORR
        DOSE = VOL*RHBZ/AREA
*
        DO 55 IT55 = 1, 5000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 551 KL = 1,NPDE
                DO 552 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
552             CONTINUE
551     CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
```

```
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 553 NN = 31, 40
            IF (T .LT. XDATA(NN)) THEN
            ELSE IF (T .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPEN
            ELSE
                  IF (TOLD .LT. XDATA(NN)) THEN
                  SLP = (PPEN - PPENOLD)/TDIF
                  YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
                  ELSE IF (TOLD .EQ. XDATA(NN)) THEN
                  YFIT(NN) = PPENOLD
                  END IF
            END IF
553   CONTINUE
*
!     write(lsim,*) 'yfit(40), 5th pass =', yfit(40), 'ydata(40), 5th pass =', ydata(40)
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 559
      ELSE
      END IF
      PPENOLD = PPEN
*
*
```

```
55     CONTINUE
*************************************************************************
*      END 5th Call to D03PPF
*************************************************************************
*
559    CHISQR = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
       EPSCHI = ABS(100.0D0*(CHISQR-CHIOLD)/CHISQR)
       write(lsim,*) 'EPSCHI =', EPSCHI
       IF (EPSCHI .LE. CHITOL) THEN
       GO TO 999
       ELSE
       END IF
       CHIOLD = CHISQR
998    CONTINUE
*
*************************************************************************
*      Final Call to D03PPF
*************************************************************************
*
999    IF (NPARAM .EQ. 3) THEN
              IF (NDEP .EQ. 1) THEN
              FDEP = A(1)
                     IF (NDIF .EQ. 2) THEN
                     DASC = A(2)
                     FLXG = A(3)
                     ELSE
                     FLXG = A(2)
                     DASC = A(3)
                     END IF
              ELSE IF (NDEP .EQ. 2) THEN
              FDEP = A(2)
                     IF (NDIF .EQ. 1) THEN
                     DASC = A(1)
                     FLXG = A(3)
                     ELSE
                     FLXG = A(1)
                     DASC = A(3)
                     END IF
              ELSE IF (NDEP .EQ. 3) THEN
              FDEP = A(3)
                     IF (NDIF .EQ. 1) THEN
                     DASC = A(1)
                     FLXG = A(2)
                     ELSE
                     FLXG = A(1)
                     DASC = A(2)
                     END IF
              END IF
       ELSE IF (NPARAM .EQ. 2) THEN
              IF (NDEP .EQ. 0) THEN
                     IF (NDIF .EQ. 1) THEN
                     DASC = A(1)
                     FLXG = A(2)
```

```
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
        ELSE IF (NDEP .EQ. 1) THEN
                FDEP = A(1)
                IF (NDIF .EQ. 2) THEN
                DASC = A(2)
                ELSE
                FLXG = A(2)
                END IF
        ELSE IF (NDEP .EQ. 2) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
                FDEP = A(2)
        END IF
    ELSE IF (NPARAM .EQ. 1) THEN
        IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
        ELSE
                FDEP = A(1)
        END IF
    END IF
*
    WRITE(lsim,*) '*****************************************************'
    write (lsim,*) 'change in flxg =', flxg*fcor/flxgold
    write (lsim,*) 'change in dasc =', dasc*dcor/dascold
    write (lsim,*) 'change in fdep =', FDEP/FDEPOLD
*
    DO 102 I = 1,NPARAM
    IF (A(I) .LE. 0.0D0) THEN
    GO TO 659
    ELSE
    END IF
102 CONTINUE
*
990 write (lsim,*) 'NO CHANGE IN PARAMETERS'
*
    WRITE(lsim,*) '*****************************************************'
    WRITE (lsim,*) 'FINAL FLXG =', FLXG*1.0D6
    WRITE (lsim,*) 'FINAL DASC =', DASC
    WRITE (lsim,*) 'FINAL FDEP =', FDEP
*
*
*   40 microliters ******************************************************
*
```

```
      IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 40.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Initial Volume of Penetrant (in microlit)= ', VOL*1.0D3
      WRITE(lsim,*) 'Initial VH Thickness (in microns) = ', VHTH*1.0D4
      WRITE(lsim,*) 'Initial Mass of Penetrant (microg/cm2)= ', DOSE*1.0D6
*
      DO 640 IT640 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 6401 KL = 1,NPDE
            DO 6402 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
6402        CONTINUE
6401  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
```

```
      TMVH = U(NEQN)*RHBZ
      WRITE(llvt40,1004) THRS, VHTI
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
      CALL MASSBL (NPTS, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
      CFLX = CFLX + FTTL*TDIF ! Cumulative Flux
      CHECK = (TMAS + CFLX)/DOSE ! Should be ~ 1
      WRITE(lmbc40,1004) THRS, CHECK
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 6403 NN = 1, 10
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
6403  CONTINUE
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 6409
      ELSE
      END IF
      PPENOLD = PPEN
*
*     Writing the Solutions!
      WRITE(lAsc40,1001) THRS
      WRITE(lAsc40,1002) (SOL(J,1), J=1, NPTS)
      WRITE(lAve40,1001) THRS
      WRITE(lAve40,1002) (SOL(J,2), J=1, NPTS)
      WRITE(lflx40,1004) THRS, ABS(FEVP)*UFLX, ABS(FBTM)*UFLX, ABS(FCAL)*UFLX
      WRITE(lcmf40,1004) THRS, PEVP, PPEN, PTTL
*
```

```
640   CONTINUE
*
*     END 40 micoliters ************************************************
*
*     20 microliters ***************************************************
*
6409  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 20.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Initial Volume of Penetrant (in microlit)= ', VOL*1.0D3
      WRITE(lsim,*) 'Initial VH Thickness (in microns) = ', VHTH*1.0D4
      WRITE(lsim,*) 'Initial Mass of Penetrant (microg/cm2)= ', DOSE*1.0D6
*
      DO 620 IT620 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 6201 KL = 1,NPDE
            DO 6202 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
6202        CONTINUE
6201  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
```

```
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
      WRITE(llvt20,1004) THRS, VHTI
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
      CALL MASSBL (NPTS, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
      CFLX = CFLX + FTTL*TDIF ! Cumulative Flux
      CHECK = (TMAS + CFLX)/DOSE ! Should be ~ 1
      WRITE(lmbc20,1004) THRS, CHECK
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 6203 NN = 11, 20
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
6203  CONTINUE
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 6209
      ELSE
      END IF
      PPENOLD = PPEN
*
*     Writing the Solutions!
      WRITE(lAsc20,1001) THRS
```

```
      WRITE(lAsc20,1002) (SOL(J,1), J=1, NPTS)
      WRITE(lAve20,1001) THRS
      WRITE(lAve20,1002) (SOL(J,2), J=1, NPTS)
      WRITE(lflx20,1004) THRS, ABS(FEVP)*UFLX, ABS(FBTM)*UFLX, ABS(FCAL)*UFLX
      WRITE(lcmf20,1004) THRS, PEVP, PPEN, PTTL
*
620   CONTINUE
*
*     END 20 micoliters ************************************************
*
*     10 microliters ***************************************************
*
6209  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 10.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      WRITE(lsim,*) '***************************************************'
      WRITE(lsim,*) 'Initial Volume of Penetrant (in microlit)= ', VOL*1.0D3
      WRITE(lsim,*) 'Initial VH Thickness (in microns) = ', VHTH*1.0D4
      WRITE(lsim,*) 'Initial Mass of Penetrant (microg/cm2)= ', DOSE*1.0D6
*
      DO 610 IT610 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 6101 KL = 1,NPDE
            DO 6102 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
6102        CONTINUE
6101  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
```

```
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
      WRITE(llvt10,1004) THRS, VHTI
*
      ELSE
*
      BZTP = SOL(NPTS,1)
      CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
      FEVP = FLXG*BZTP/BZSCIN
!     FEVP = FLXG*SOL(NPTS,1)
      TMVH = 0.0D0
      END IF
*
      CALL MASSBL (NPTS, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
      CFLX = CFLX + FTTL*TDIF ! Cumulative Flux
      CHECK = (TMAS + CFLX)/DOSE ! Should be ~ 1
      WRITE(lmbc10,1004) THRS, CHECK
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 6103 NN = 21, 30
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
6103  CONTINUE
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      GO TO 6109
```

```
      ELSE
      END IF
      PPENOLD = PPEN
*
*     Writing the Solutions!
      WRITE(lAsc10,1001) THRS
      WRITE(lAsc10,1002) (SOL(J,1), J=1, NPTS)
      WRITE(lAve10,1001) THRS
      WRITE(lAve10,1002) (SOL(J,2), J=1, NPTS)
      WRITE(lflx10,1004) THRS, ABS(FEVP)*UFLX, ABS(FBTM)*UFLX, ABS(FCAL)*UFLX
      WRITE(lcmf10,1004) THRS, PEVP, PPEN, PTTL
*
610   CONTINUE
*
*     END 10 micoliters ************************************************
*
*     5 microliters ****************************************************
*
6109  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 5.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHBZ/AREA
*
      WRITE(lsim,*) '****************************************************'
      WRITE(lsim,*) 'Initial Volume of Penetrant (in microlit)= ', VOL*1.0D3
      WRITE(lsim,*) 'Initial VH Thickness (in microns) = ', VHTH*1.0D4
      WRITE(lsim,*) 'Initial Mass of Penetrant (microg/cm2)= ', DOSE*1.0D6
*
      DO 65 IT65 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 651 KL = 1,NPDE
            DO 652 KM = 1,NPTS
```

```
                    SOL(KM,KL) = U(NPDE*(KM-1)+KL)
652           CONTINUE
651     CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHBZ
        WRITE(llvt5,1004) THRS, VHTI
*
        ELSE
*
        BZTP = SOL(NPTS,1)
        CALL DENSTY (BZTP, RHBZ, RHSC, RHAV)
        FEVP = FLXG*BZTP/BZSCIN
!       FEVP = FLXG*SOL(NPTS,1)
        TMVH = 0.0D0
        END IF
*
        CALL MASSBL (NPTS, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
        CFLX = CFLX + FTTL*TDIF ! Cumulative Flux
        CHECK = (TMAS + CFLX)/DOSE ! Should be ~ 1
        WRITE(lmbc5,1004) THRS, CHECK
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 653 NN = 31, 40
        IF (T .LT. XDATA(NN)) THEN
        ELSE IF (T .EQ. XDATA(NN)) THEN
        YFIT(NN) = PPEN
        ELSE
                IF (TOLD .LT. XDATA(NN)) THEN
                SLP = (PPEN - PPENOLD)/TDIF
                YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
                ELSE IF (TOLD .EQ. XDATA(NN)) THEN
                YFIT(NN) = PPENOLD
                END IF
```

```
      END IF
653   CONTINUE
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      CHISQR = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
      WRITE(lsim,*) '************************************************'
      write (lsim,*) 'NFREE =', NFREE
      WRITE (lsim,*) 'FINAL CHISQ (*10^5) =', CHISQR*1.0D5
      WRITE (lsim,*) 'FINAL SSR =', CHISQR*DBLE(NFREE)
      GO TO 659
      ELSE
      END IF
      PPENOLD = PPEN
*
*     Writing the Solutions!
      WRITE(lAsc5,1001) THRS
      WRITE(lAsc5,1002) (SOL(J,1), J=1, NPTS)
      WRITE(lAve5,1001) THRS
      WRITE(lAve5,1002) (SOL(J,2), J=1, NPTS)
      WRITE(lflx5,1004) THRS, ABS(FEVP)*UFLX, ABS(FBTM)*UFLX, ABS(FCAL)*UFLX
      WRITE(lcmf5,1004) THRS, PEVP, PPEN, PTTL
*
65          CONTINUE
*************************************************************************
*     END Final Call to D03PPF
*************************************************************************
*
*
1001  FORMAT('Time (s) ', F24.12)
1002  FORMAT(5(F18.12,1X))
1003  FORMAT(F36.12, 2X, F18.12, 2X, F18.12)
1004  FORMAT(F36.12, 2X, 6(F36.12))
*
659         CLOSE(lAsc, STATUS='KEEP')
            CLOSE(lAve, STATUS='KEEP')
            CLOSE(llvt, STATUS='KEEP')
            CLOSE(lsim, STATUS='KEEP')
            CLOSE(lflx, STATUS='KEEP')
            CLOSE(lmbc, STATUS='KEEP')
            CLOSE(lcmf, STATUS='KEEP')
!           CLOSE(lext, STATUS='KEEP')
*
            STOP
            END
*
*
```

Program ETOHGRIDVARD simulates the evaporation and absorption of ethanol through skin sub layers. It uses a concentration-dependent diffusivity and has the option of fitting several different components of the variable diffusivity in random orders. This program can be used for benzene by changing the physicochemical properties of and transport parameters.

```
      PROGRAM ETOHGRIDVARD
*
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      IMPLICIT INTEGER (I-N)
*
*     NPDE   = Number of PDEs
*     NCODE  = Number of Coupled ODEs
*     M      = Determines the Coordinate system (0 = Rectangular Cartesian)
*     NPTS   = Number of Mesh Points
*     NXI    = Number of PDE-ODE Coupling Points
*     NXFIX  = Number of Fixed Mesh Points
      PARAMETER (NPDE=2, NCODE=1, M=0, NPTS=201, NXI=0, NXFIX=0)
*
*     DO NOT CHANGE THE LINE BELOW!
      PARAMETER (NWKRES = NPDE*(3*NPDE+6*NXI+NPTS+15)+ NXI+NCODE
     $                  + 7*NPTS+NXFIX+2+9000)
!     Increase the value 9000 if more workspace is needed
      PARAMETER (NEQN=NPDE*NPTS+NCODE)
      PARAMETER (NIW=25*NEQN+25+NXFIX+90000)
!     Dimension of the array IW. Pg. 17. DO NOT CHANGE unless IFAIL 15!
      PARAMETER (LENODE=(6+5)*NEQN+50) ! For theta method, Pg. 17!
      PARAMETER (NW=4*NEQN+11*NEQN/2+1+NWKRES+LENODE+20000)
!     DO NOT CHANGE unless IFAIL 15 !
*
      PARAMETER (NFIT = 0) ! Keep a non-zero value for optimization
      PARAMETER (NEXDP = 30)
      PARAMETER (NPARAM = 1)
      PARAMETER (NFLX = 0)
      PARAMETER (NSAT = 1)
      PARAMETER (MODE = 0)
*
      DOUBLE PRECISION U(NEQN), X(NEQN), RTOL(NEQN), ATOL(NEQN), XI(1),
     +                 XFIX(1), IW(NIW), W(NW), ALGOPT(30), SOL(NPTS,NPDE),
     +                 XDATA(NEXDP), YDATA(NEXDP), YFIT(NEXDP),
     +                 SIGMAY(NEXDP), A(10), DELTAA(10), SIGMAA(10)
*     IW(NIW) - Output variable, W(NW) - Workspace variable
      INTEGER ITIME(7)
*
      LOGICAL REMESH, THETA
      CHARACTER*1 LAOPT, NORM
*
*     External Subroutines
      EXTERNAL UVINIT,ODEDEF,PDEDEF,BNDARY,D03PPF,D03PCL,D03PCK,X05AAF
!     D03PCL replaces MONITF as dummy subroutine!
*
*     Common Blocks
```

```
      COMMON /EXPMT/ TEMP, DOSE, VHTH, ETSCIN
      COMMON /VHPAR/ RHET, WMET
      COMMON /SCPAR/ RHSC, SCTH, DWNK, FDEP, CORR
      COMMON /VEPAR/ RHVT, VTTH, DAVT
      COMMON /INTER/ FLXG, FLSCVH, FLVESC, FBTM, SCVHPC, VESCPC, PTOR
      COMMON /VPCPH/ VPCA, VPCB, VPCC, VPCD, VPCE
      COMMON /CRRCT/ FCOR, DCORR1, DCORR2, SLPD, CTRN
*
*     Initialize the constants specific to the system of interest
      CALL CONSTS
*
*     Write the Solution Files
      lsim = 11 ! Description of the Simulation
      lAsc40 = 21 ! Mass Fraction of Solvent in the Stratum Corneum !
      lAsc20 = 22 ! Mass Fraction of Solvent in the Stratum Corneum !
      lAsc10 = 23 ! Mass Fraction of Solvent in the Stratum Corneum !
      lAsc5  = 24 ! Mass Fraction of Solvent in the Stratum Corneum !
      lAve40 = 31 ! Mass Fraction of Solvent in the Viable Epidermis !
      lAve20 = 32 ! Mass Fraction of Solvent in the Viable Epidermis !
      lAve10 = 33 ! Mass Fraction of Solvent in the Viable Epidermis !
      lAve5  = 34 ! Mass Fraction of Solvent in the Viable Epidermis !
      llvt40 = 41 ! Instantaneous Thickness of the Donor Solution !
      llvt20 = 42 ! Instantaneous Thickness of the Donor Solution !
      llvt10 = 43 ! Instantaneous Thickness of the Donor Solution !
      llvt5  = 44 ! Instantaneous Thickness of the Donor Solution !
      lflx40 = 51 ! Fluxes
      lflx20 = 52
      lflx10 = 53
      lflx5  = 54
      lmbc40 = 61 ! Mass balance component
      lmbc20 = 62 ! Mass balance component
      lmbc10 = 63 ! Mass balance component
      lmbc5  = 64 ! Mass balance component
      lcmf40 = 71 ! Cumulative Fluxes
      lcmf20 = 72 ! Cumulative Fluxes
      lcmf10 = 73 ! Cumulative Fluxes
      lcmf5  = 74 ! Cumulative Fluxes
      lext = 18 ! Anciliary data
*
      itrace = 2 ! See D03PPF Manual, Pg. 18 !
*
      OPEN (lsim, file='sim.dat', status='new')
      OPEN (lAsc40, file='Asc40.dat', status='new')
      OPEN (lAsc20, file='Asc20.dat', status='new')
!     OPEN (lAsc10, file='Asc10.dat', status='new')
      OPEN (lAsc5, file='Asc5.dat', status='new')
      OPEN (lAve40, file='Ave40.dat', status='new')
      OPEN (lAve20, file='Ave20.dat', status='new')
!     OPEN (lAve10, file='Ave10.dat', status='new')
      OPEN (lAve5, file='Ave5.dat', status='new')
      OPEN (llvt40, file='lvt40.dat', status='new')
      OPEN (llvt20, file='lvt20.dat', status='new')
!     OPEN (llvt10, file='lvt10.dat', status='new')
```

```
      OPEN (llvt5, file='lvt5.dat', status='new')
      OPEN (lflx40, file='flx40.dat', status='new')
      OPEN (lflx20, file='flx20.dat', status='new')
!     OPEN (lflx10, file='flx10.dat', status='new')
      OPEN (lflx5, file='flx5.dat', status='new')
      OPEN (lmbc40, file='mbc40.dat', status='new')
      OPEN (lmbc20, file='mbc20.dat', status='new')
!     OPEN (lmbc10, file='mbc10.dat', status='new')
      OPEN (lmbc5, file='mbc5.dat', status='new')
      OPEN (lcmf40, file='cmf40.dat', status='new')
      OPEN (lcmf20, file='cmf20.dat', status='new')
!     OPEN (lcmf10, file='cmf10.dat', status='new')
      OPEN (lcmf5, file='cmf5.dat', status='new')
!     OPEN (lext, file='fit.dat', status='new')
*
*
*     Experimental Data
      UFLX = 1.0D6*3600.0D0
      AREA = 0.79D0
*
      XDATA(1)  = 0.08D0*3600.0D0
      XDATA(2)  = 0.17D0*3600.0D0
      XDATA(3)  = 0.25D0*3600.0D0
      XDATA(4)  = 0.33D0*3600.0D0
      XDATA(5)  = 0.67D0*3600.0D0
      XDATA(6)  = 1.00D0*3600.0D0
      XDATA(7)  = 2.00D0*3600.0D0
      XDATA(8)  = 4.00D0*3600.0D0
      XDATA(9)  = 8.50D0*3600.0D0
      XDATA(10) = 24.00D0*3600.0D0
*
      DO 101 I = 1,10
      XDATA(I+10) = XDATA(I)
      XDATA(I+20) = XDATA(I)
!     XDATA(I+30) = XDATA(I)
101   CONTINUE
*
*
*     40 microliters
      YDATA(1)  = 0.00D0
      YDATA(2)  = 0.01D0
      YDATA(3)  = 0.02D0
      YDATA(4)  = 0.04D0
      YDATA(5)  = 0.10D0
      YDATA(6)  = 0.13D0
      YDATA(7)  = 0.16D0
      YDATA(8)  = 0.17D0
      YDATA(9)  = 0.17D0
      YDATA(10) = 0.18D0
*
*     20 microliters
      YDATA(11)  = 0.01D0
      YDATA(12)  = 0.02D0
```

```
      YDATA(13)  = 0.04D0
      YDATA(14)  = 0.06D0
      YDATA(15)  = 0.13D0
      YDATA(16)  = 0.17D0
      YDATA(17)  = 0.20D0
      YDATA(18)  = 0.22D0
      YDATA(19)  = 0.23D0
      YDATA(20)  = 0.23D0
*
*     5 microliters
      YDATA(21)  = 0.02D0
      YDATA(22)  = 0.05D0
      YDATA(23)  = 0.08D0
      YDATA(24)  = 0.11D0
      YDATA(25)  = 0.19D0
      YDATA(26)  = 0.23D0
      YDATA(27)  = 0.27D0
      YDATA(28)  = 0.29D0
      YDATA(29)  = 0.29D0
      YDATA(30)  = 0.29D0
*
*     Spreadsheet Constants
      RKEVAPM = 0.64202D0
      DASCM = 6.87D-10
      SCVHPCM = 0.38D0
      DAVEM = 4.31D-6
      PTORM = 87.93
*
*
      WRITE(lsim,*) ' 3-Layer Pure Penetrant Isothermal Simulation '
      WRITE(lsim,*) ' The Penetrant in ETHANOL'
*
      WRITE(lsim,*) 'Number of Mesh Points = ', NPTS
      NDPD = ((NPTS-1)*FDEP) + 1
      WRITE(lsim,*) 'Number of Permeant Deposited Mesh Points = ', NDPD
*
      WRITE(lsim,*) '*************************************************'
      WRITE(lsim,*) 'Molecular Weight of the Penetrant = ', WMET
      WRITE(lsim,*) 'Constant System Temperature = ', TEMP - 273.15D0
      WRITE(lsim,*) 'Density of Penetrant = ', RHET
*
      WRITE(lsim,*) '*************************************************'
      WRITE(lsim,*) 'Density of SC Phase = ', RHSC
      WRITE(lsim,*) 'Thickness of SC Phase (in microns) = ', SCTH*1.0D4
      WRITE(lsim,*) 'Predicted Diffusivity of A in SC = ', DWNK
      WRITE(lsim,*) 'Initial Multiplier for Lower Diffusivity = ', DCORR1
      WRITE(lsim,*) 'Initial Multiplier for Higher Diffusivity = ', DCORR1*DCORR2
      WRITE(lsim,*) 'Initial Lower Limit of SC Diffusivity = ', DWNK*DCORR1
      WRITE(lsim,*) 'Initial Upper Limit of SC Diffusivity = ', DWNK*DCORR1*DCORR2
      WRITE(lsim,*) 'Diffusivity of A in SC (Spreadsheet) = ', DASCM
*
      WRITE(lsim,*) '*************************************************'
      WRITE(lsim,*) 'Density of VE Phase = ', RHVT
```

```
      WRITE(lsim,*) 'Thickness of VE Phase (in microns) = ', VTTH*1.0D4
      WRITE(lsim,*) 'Diffusivity of A in VE = ', DAVT
      WRITE(lsim,*) 'Diffusivity of A in VE (Spreadsheet) = ', DAVEM
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Vapor Pressure in Torr = ', PTOR
      WRITE(lsim,*) 'Vapor Pressure in Torr (Spreadsheet) = ', PTORM
      WRITE(lsim,*) 'Corrected KEVAP = ', FLXG*3600.0D0/RHET
      WRITE(lsim,*) 'Initial Correction Factor for FLXG = ', FCOR
      WRITE(lsim,*) 'Predicted KEVAP = ', FLXG*3600.0D0/(FCOR*RHET)
      WRITE(lsim,*) 'Spreadsheet KEVAP = ', RKEVAPM
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Fractional Deposition Depth (%)= ', FDEP*100.0D0
      WRITE(lsim,*) 'Thickness Correction due to Solvent Deposition = ', CORR*1.0D4
      WRITE(lsim,*) 'SC-VH Partition Coefficient of Ethanol = ', SCVHPC
      WRITE(lsim,*) 'SC-VH Partition Coefficient of Ethanol (Spreadsheet) = ', SCVHPCM
      WRITE(lsim,*) 'VE-SC Partition Coefficient of Ethanol = ', VESCPC
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Multiplying Factor for Saturated Diffusivity = ', DCORR2
      WRITE(lsim,*) 'Slope of the Inflexion Line = ', SLPD
      WRITE(lsim,*) 'Concentration corresponding to Inflexion Point = ', CTRN
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Number of Fitted Parameters = ', NPARAM
      WRITE(lsim,*) 'Weighting Coefficient = ', MODE
      WRITE(lsim,*) 'Fitting EVAP = ', NFLX
      WRITE(lsim,*) 'Fitting DSAT = ', NSAT
*
*     Initialize Mesh - Uniform Mesh
*
      DO 910 I = 1, NPTS
            X(I) = DBLE(I-1)/(NPTS-1.0D0)
910   CONTINUE
*
*     Set Remesh Parameters
*
      REMESH = .false.
      NRMESH = 0
      DXMESH = 0.0D0
      TRMESH = 1.0D0
      IPMINF = 0
      XRATIO = 1.5D0
      CONST  = 2.0D0/(NPTS-1)
      IND = 0
      ITASK = 2
      NORM = 'A'
      LAOPT= 'S' !use the sparse matrix routines
      THETA = .false.
*
      DO 920 I=1,30
            ALGOPT(I)=0.0D0
```

```
920   CONTINUE
      ALGOPT(4)=2.0D0 !Does not perform the Petzold test
      ALGOPT(29)=0.5 !NAG Deafault = 0.1
      ALGOPT(30)=1.0D-300 !NAG Default 1.0D-2!
*
*     Set Time Step & Convergence Criterion
      DELT = 1.0D-3
      WRITE(lsim,*) 'Time Increament Step Size =', DELT
      DO 930 K = 1, NEQN
      RTOL(K) = 1.0D-8
      ATOL(K) = 1.0D-8
930   CONTINUE
      ITOL = 4
      WRITE(lsim,*) 'RTOL = ', RTOL(1)
      WRITE(lsim,*) 'ATOL = ', ATOL(1)
      WRITE(lsim,*) 'ITOL = ', ITOL
*
*     Write Simulation Date and Time
      CALL X05AAF(ITIME)
      WRITE(lsim,*) 'Month / Day / Year = ', ITIME(2),'/',ITIME(3),'/',ITIME(1)
      WRITE(lsim,*) 'Hour / Minute / Second = ', ITIME(4),'/',ITIME(5),'/',ITIME(6)
*
*
      FLXGOLD = FLXG
      DCORR2OLD = DCORR2
      DCORR1OLD = DCORR1
*
      IF (NPARAM .EQ. 3) THEN
            IF (NFLX .EQ. 1) THEN
            A(1) = FLXG
                  IF (NSAT .EQ. 2) THEN
                  A(2) = DCORR2
                  A(3) = DCORR1
                  ELSE
                  A(2) = DCORR1
                  A(3) = DCORR2
                  END IF
            ELSE IF (NFLX .EQ. 2) THEN
            A(2) = FLXG
                  IF (NSAT .EQ. 1) THEN
                  A(1) = DCORR2
                  A(3) = DCORR1
                  ELSE
                  A(1) = DCORR1
                  A(3) = DCORR2
                  END IF
            ELSE IF (NFLX .EQ. 3) THEN
            A(3) = FLXG
                  IF (NSAT .EQ. 1) THEN
                  A(1) = DCORR2
                  A(2) = DCORR1
                  ELSE
                  A(1) = DCORR2
```

```
                A(2) = DCORR1
                END IF
        END IF
    ELSE IF (NPARAM .EQ. 2) THEN
        IF (NFLX .EQ. 0) THEN
                IF (NSAT .EQ. 1) THEN
                A(1) = DCORR2
                A(2) = DCORR1
                ELSE
                A(1) = DCORR1
                A(2) = DCORR2
                END IF
        ELSE IF (NFLX .EQ. 1) THEN
                A(1) = FLXG
                IF (NSAT .EQ. 2) THEN
                A(2) = DCORR2
                ELSE
                A(2) = DCORR1
                END IF
        ELSE IF (NFLX .EQ. 2) THEN
                IF (NSAT .EQ. 1) THEN
                A(1) = DCORR2
                ELSE
                A(1) = DCORR1
                END IF
                A(2) = FLXG
        END IF
    ELSE IF (NPARAM .EQ. 1) THEN
        IF (NFLX .EQ. 0) THEN
                IF (NSAT .EQ. 1) THEN
                A(1) = DCORR2
                ELSE
                A(1) = DCORR1
                DCORR2 = 1.0D0/A(1)
                END IF
        ELSE
                A(1) = FLXG
        END IF
    END IF
*
    DO 940 LL = 1,NPARAM
    DELTAA(LL) = 0.1D0*A(LL)
940 CONTINUE
    DO 941 LM = 1,NPARAM
    SIGMAY(LM) = 1.0D0
941 CONTINUE
*
*
    NFREE = NEXDP - NPARAM
*
    IF (NFIT .EQ. 0) THEN
!   NFREE = 1
    GO TO 999
```

```
      ELSE
      WRITE (lsim,*) 'RUNNING THE GRID SEARCH OPTIMIZER'
      END IF
*
*
*     Optimization with GRID SEARCH
      CHITOL = 0.10D0
      CHIOLD = 0.0D0
*
      DO 998 NOPT = 1, 1000
      write(lsim,*) 'NOPT =', NOPT
*
      DO 90 J = 1, NPARAM
*
*     EVALUATE CHI SQUARE AT FIRST TWO SEARCH POINTS
*
************************************************************************
*     1st Call to D03PPF
************************************************************************
*
      IF (NPARAM .EQ. 3) THEN
            IF (NFLX .EQ. 1) THEN
            FLXG = A(1)
                  IF (NSAT .EQ. 2) THEN
                  DCORR2 = A(2)
                  DCORR1 = A(3)
                  ELSE
                  DCORR1 = A(2)
                  DCORR2 = A(3)
                  END IF
            ELSE IF (NFLX .EQ. 2) THEN
            FLXG = A(2)
                  IF (NSAT .EQ. 1) THEN
                  DCORR2 = A(1)
                  DCORR1 = A(3)
                  ELSE
                  DCORR1 = A(1)
                  DCORR2 = A(3)
                  END IF
            ELSE IF (NFLX .EQ. 3) THEN
            FLXG = A(3)
                  IF (NSAT .EQ. 1) THEN
                  DCORR1 = A(1)
                  DCORR2 = A(2)
                  ELSE
                  DCORR2 = A(1)
                  DCORR1 = A(2)
                  END IF
            END IF
      ELSE IF (NPARAM .EQ. 2) THEN
            IF (NFLX .EQ. 0) THEN
                  IF (NSAT .EQ. 1) THEN
                  DCORR2 = A(1)
```

```
                DCORR1 = A(2)
                ELSE
                DCORR1 = A(1)
                DCORR2 = A(2)
                END IF
        ELSE IF (NFLX .EQ. 1) THEN
                FLXG = A(1)
                IF (NSAT .EQ. 2) THEN
                DCORR2 = A(2)
                ELSE
                DCORR1 = A(2)
                END IF
        ELSE IF (NFLX .EQ. 2) THEN
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                ELSE
                DCORR1 = A(1)
                END IF
                FLXG = A(2)
        END IF
    ELSE IF (NPARAM .EQ. 1) THEN
        IF (NFLX .EQ. 0) THEN
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                ELSE
                DCORR1 = A(1)
                DCORR2 = 1.0D0/DCORR1
                END IF
        ELSE
                FLXG = A(1)
        END IF
    END IF
*
*       40 microliters ******************************************************
*
    IND = 0
    T = 0.0D0
    TOUT = 0.0D0
    TOLD = 0.0D0
    CFLX = 0.0D0
    CEVP = 0.0D0
    CPEN = 0.0D0
    PPENOLD = 0.0D0
*
    VOL = 40.0D-3
    VHTH = VOL/AREA - CORR
    DOSE = VOL*RHET/AREA
*
    DO 140 IT140 = 1, 5000
    TOUT = T + DELT
*
    IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
```

```
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 1401 KL = 1,NPDE
                DO 1402 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
1402            CONTINUE
1401    CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHET
*
        ELSE
*
        FEVP = FLXG*SOL(NPTS,1)/ETSCIN
        TMVH = 0.0D0
        END IF
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 1403 NN = 1, 10
        IF (T .LT. XDATA(NN)) THEN
        ELSE IF (T .EQ. XDATA(NN)) THEN
        YFIT(NN) = PPEN
        ELSE
                IF (TOLD .LT. XDATA(NN)) THEN
                SLP = (PPEN - PPENOLD)/TDIF
                YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
                ELSE IF (TOLD .EQ. XDATA(NN)) THEN
```

```
            YFIT(NN) = PPENOLD
            END IF
      END IF
1403  CONTINUE
*
!     write(lsim,*) 'yfit(10), 1st pass =', yfit(10), 'ydata(10), 1st pass =', ydata(10)
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
      GO TO 1409
      ELSE
      END IF
      PPENOLD = PPEN
*
140   CONTINUE
*
*     END 40 micoliters ****************************************************
*
*     20 microliters *******************************************************
*
1409  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 20.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      DO 120 IT120 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 1201 KL = 1,NPDE
            DO 1202 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
1202        CONTINUE
1201  CONTINUE
```

```
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 1203 NN = 11, 20
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
1203  CONTINUE
*
!     write(lsim,*) 'yfit(20), 1st pass =', yfit(20), 'ydata(20), 1st pass =', ydata(20)
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
      GO TO 1209
      ELSE
      END IF
      PPENOLD = PPEN
*
120   CONTINUE
*
```

```
*       END 20 micoliters ***************************************************
*
*       5 microliters *******************************************************
*
1209    IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
        VOL = 5.0D-3
        VHTH = VOL/AREA - CORR
        DOSE = VOL*RHET/AREA
*
        DO 15 IT15 = 1, 5000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 151 KL = 1,NPDE
                DO 152 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
152             CONTINUE
151     CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHET
```

```
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 153 NN = 21, 30
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
153   CONTINUE
*
!     write(lsim,*) 'yfit(40), 1st pass =', yfit(40), 'ydata(40), 1st pass =', ydata(40)
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
      GO TO 159
      ELSE
      END IF
      PPENOLD = PPEN
*
*
15    CONTINUE
************************************************************************
*     END 1st Call to D03PPF
************************************************************************
*
159   CHISQ1 = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!     write(lsim,*) 'end 1st NAG time looping'
      FN  = 0.0D0
      DELTA = DELTAA(J)
41    A(J) = A(J) + DELTA
*
************************************************************************
*     2nd Call to D03PPF
************************************************************************
*
```

```
            DO 2020 MM = 1,NEXDP
            YFIT(MM) = 0.0D0
2020  CONTINUE
*
*
      IF (NPARAM .EQ. 3) THEN
            IF (NFLX .EQ. 1) THEN
            FLXG = A(1)
                  IF (NSAT .EQ. 2) THEN
                  DCORR2 = A(2)
                  DCORR1 = A(3)
                  ELSE
                  DCORR1 = A(2)
                  DCORR2 = A(3)
                  END IF
            ELSE IF (NFLX .EQ. 2) THEN
            FLXG = A(2)
                  IF (NSAT .EQ. 1) THEN
                  DCORR2 = A(1)
                  DCORR1 = A(3)
                  ELSE
                  DCORR1 = A(1)
                  DCORR2 = A(3)
                  END IF
            ELSE IF (NFLX .EQ. 3) THEN
            FLXG = A(3)
                  IF (NSAT .EQ. 1) THEN
                  DCORR1 = A(1)
                  DCORR2 = A(2)
                  ELSE
                  DCORR2 = A(1)
                  DCORR1 = A(2)
                  END IF
            END IF
      ELSE IF (NPARAM .EQ. 2) THEN
            IF (NFLX .EQ. 0) THEN
                  IF (NSAT .EQ. 1) THEN
                  DCORR2 = A(1)
                  DCORR1 = A(2)
                  ELSE
                  DCORR1 = A(1)
                  DCORR2 = A(2)
                  END IF
            ELSE IF (NFLX .EQ. 1) THEN
                  FLXG = A(1)
                  IF (NSAT .EQ. 2) THEN
                  DCORR2 = A(2)
                  ELSE
                  DCORR1 = A(2)
                  END IF
            ELSE IF (NFLX .EQ. 2) THEN
                  IF (NSAT .EQ. 1) THEN
                  DCORR2 = A(1)
```

```
                ELSE
                DCORR1 = A(1)
                END IF
                FLXG = A(2)
            END IF
        ELSE IF (NPARAM .EQ. 1) THEN
            IF (NFLX .EQ. 0) THEN
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                ELSE
                DCORR1 = A(1)
                DCORR2 = 1.0D0/DCORR1
                END IF
            ELSE
                FLXG = A(1)
            END IF
        END IF
*
*       40 microliters ******************************************************
*
        IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
        VOL = 40.0D-3
        VHTH = VOL/AREA - CORR
        DOSE = VOL*RHET/AREA
*
        DO 240 IT240 = 1, 5000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $              ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $              REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $              CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 2401 KL = 1,NPDE
            DO 2402 KM = 1,NPTS
                SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2402        CONTINUE
2401    CONTINUE
```

```
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 2403 NN = 1, 10
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
2403  CONTINUE
*
!     write(lsim,*) 'yfit(10), 2nd pass =', yfit(10), 'ydata(4), 2nd pass =', ydata(10)
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
      GO TO 2409
      ELSE
      END IF
      PPENOLD = PPEN
*
```

```
240   CONTINUE
*
*     END 40 micoliters ************************************************
*
*     20 microliters ***************************************************
*
2409  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 20.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      DO 220 IT220 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 2201 KL = 1,NPDE
            DO 2202 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2202        CONTINUE
2201  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
```

```
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 2203 NN = 11, 20
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
2203  CONTINUE
*
!     write(lsim,*) 'yfit(20), 2nd pass =', yfit(20), 'ydata(20), 2nd pass =', ydata(20)
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
      GO TO 2209
      ELSE
      END IF
      PPENOLD = PPEN
*
220   CONTINUE
*
*     END 20 micoliters ************************************************
*
*     5 microliters ****************************************************
*
2209  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
```

```
*
      VOL = 5.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      DO 25 IT25 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 251 KL = 1,NPDE
            DO 252 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
252         CONTINUE
251   CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
```

```
      PTTL = PEVP + PPEN
*
            DO 253 NN = 21, 30
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
253   CONTINUE
*
!     write(lsim,*) 'yfit(40), 2nd pass =', yfit(40), 'ydata(40), 2nd pass =', ydata(40)
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
      GO TO 259
      ELSE
      END IF
      PPENOLD = PPEN
*
*
25    CONTINUE
************************************************************************
*     END 2nd Call to D03PPF
************************************************************************
*
259   CHISQ2 = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!     write(lsim,*) 'end 2nd NAG time looping'
      write(lsim,*) 'chisq2, 2nd pass', chisq2, 'j =', j
*
      IF (CHISQ1 .LT. CHISQ2) THEN
*
*     CHI SQUARE IS INCREASING, REVERSE DIRECTION OF SEARCH
*
      DELTA = -DELTA
      A(J) = A(J) + DELTA
*
************************************************************************
*     3rd Call to D03PPF
************************************************************************
*
            DO 2030 MM = 1,NEXDP
            YFIT(MM) = 0.0D0
2030  CONTINUE
*
*
      IF (NPARAM .EQ. 3) THEN
            IF (NFLX .EQ. 1) THEN
```

```
        FLXG = A(1)
               IF (NSAT .EQ. 2) THEN
               DCORR2 = A(2)
               DCORR1 = A(3)
               ELSE
               DCORR1 = A(2)
               DCORR2 = A(3)
               END IF
        ELSE IF (NFLX .EQ. 2) THEN
        FLXG = A(2)
               IF (NSAT .EQ. 1) THEN
               DCORR2 = A(1)
               DCORR1 = A(3)
               ELSE
               DCORR1 = A(1)
               DCORR2 = A(3)
               END IF
        ELSE IF (NFLX .EQ. 3) THEN
        FLXG = A(3)
               IF (NSAT .EQ. 1) THEN
               DCORR1 = A(1)
               DCORR2 = A(2)
               ELSE
               DCORR2 = A(1)
               DCORR1 = A(2)
               END IF
        END IF
ELSE IF (NPARAM .EQ. 2) THEN
        IF (NFLX .EQ. 0) THEN
               IF (NSAT .EQ. 1) THEN
               DCORR2 = A(1)
               DCORR1 = A(2)
               ELSE
               DCORR1 = A(1)
               DCORR2 = A(2)
               END IF
        ELSE IF (NFLX .EQ. 1) THEN
               FLXG = A(1)
               IF (NSAT .EQ. 2) THEN
               DCORR2 = A(2)
               ELSE
               DCORR1 = A(2)
               END IF
        ELSE IF (NFLX .EQ. 2) THEN
               IF (NSAT .EQ. 1) THEN
               DCORR2 = A(1)
               ELSE
               DCORR1 = A(1)
               END IF
               FLXG = A(2)
        END IF
ELSE IF (NPARAM .EQ. 1) THEN
        IF (NFLX .EQ. 0) THEN
```

```
                  IF (NSAT .EQ. 1) THEN
                  DCORR2 = A(1)
                  ELSE
                  DCORR1 = A(1)
                  DCORR2 = 1.0D0/DCORR1
                  END IF
            ELSE
                  FLXG = A(1)
            END IF
      END IF
*
*     40 microliters ******************************************************
*
      IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 40.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      DO 340 IT340 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 3401 KL = 1,NPDE
            DO 3402 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
3402        CONTINUE
3401  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
```

```
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHET
*
        ELSE
*
        FEVP = FLXG*SOL(NPTS,1)/ETSCIN
        TMVH = 0.0D0
        END IF
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 3403 NN = 1, 10
        IF (T .LT. XDATA(NN)) THEN
        ELSE IF (T .EQ. XDATA(NN)) THEN
        YFIT(NN) = PPEN
        ELSE
                IF (TOLD .LT. XDATA(NN)) THEN
                SLP = (PPEN - PPENOLD)/TDIF
                YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
                ELSE IF (TOLD .EQ. XDATA(NN)) THEN
                YFIT(NN) = PPENOLD
                END IF
        END IF
3403    CONTINUE
*
!       write(lsim,*) 'yfit(10), 3rd pass =', yfit(10), 'ydata(10), 3rd pass =', ydata(10)
*
        TOLD = T
        IF (THRS .GT. 72.0D0) THEN
        GO TO 3409
        ELSE
        END IF
        PPENOLD = PPEN
*
340     CONTINUE
*
*       END 40 micoliters ***************************************************
*
*       20 microliters ******************************************************
*
3409    IND = 0
```

```
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 20.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      DO 320 IT320 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 3201 KL = 1,NPDE
            DO 3202 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
3202        CONTINUE
3201  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
```

```
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 3203 NN = 11, 20
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
3203  CONTINUE
*
!     write(lsim,*) 'yfit(20), 3rd pass =', yfit(20), 'ydata(20), 3rd pass =', ydata(20)
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
      GO TO 3209
      ELSE
      END IF
      PPENOLD = PPEN
*
320   CONTINUE
*
*     END 20 micoliters ***************************************************
*
*     5 microliters *******************************************************
*
3209  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 5.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      DO 35 IT35 = 1, 5000
      TOUT = T + DELT
```

```
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 351 KL = 1,NPDE
            DO 352 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
352         CONTINUE
351   CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 353 NN = 21, 30
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
```

```
              IF (TOLD .LT. XDATA(NN)) THEN
              SLP = (PPEN - PPENOLD)/TDIF
              YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
              ELSE IF (TOLD .EQ. XDATA(NN)) THEN
              YFIT(NN) = PPENOLD
              END IF
       END IF
353    CONTINUE
*
!      write(lsim,*) 'yfit(40), 3rd pass =', yfit(40), 'ydata(40), 3rd pass =', ydata(40)
*
       TOLD = T
       IF (THRS .GT. 72.0D0) THEN
       GO TO 359
       ELSE
       END IF
       PPENOLD = PPEN
*
*
35     CONTINUE
************************************************************************
*      END 3rd Call to D03PPF
************************************************************************
*
359    SAVE = CHISQ1
       CHISQ1 = CHISQ2
       SAVE = CHISQ2
       ELSE IF (CHISQ1 .EQ. CHISQ2) THEN
       GO TO 41
       ELSE
       END IF
!      write(lsim,*) 'end 3rd NAG time looping'
*
*      INCREASE A(J) UNTIL CHI SQUARE INCREASES
*
61     FN = FN + 1.0D0
       A(J) = A(J) + DELTA
*
************************************************************************
*      4th Call to D03PPF
************************************************************************
*
              DO 2040 MM = 1,NEXDP
              YFIT(MM) = 0.0D0
2040   CONTINUE
*
*
       IF (NPARAM .EQ. 3) THEN
              IF (NFLX .EQ. 1) THEN
              FLXG = A(1)
                     IF (NSAT .EQ. 2) THEN
                     DCORR2 = A(2)
                     DCORR1 = A(3)
```

```
                ELSE
                DCORR1 = A(2)
                DCORR2 = A(3)
                END IF
        ELSE IF (NFLX .EQ. 2) THEN
        FLXG = A(2)
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                DCORR1 = A(3)
                ELSE
                DCORR1 = A(1)
                DCORR2 = A(3)
                END IF
        ELSE IF (NFLX .EQ. 3) THEN
        FLXG = A(3)
                IF (NSAT .EQ. 1) THEN
                DCORR1 = A(1)
                DCORR2 = A(2)
                ELSE
                DCORR2 = A(1)
                DCORR1 = A(2)
                END IF
        END IF
ELSE IF (NPARAM .EQ. 2) THEN
        IF (NFLX .EQ. 0) THEN
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                DCORR1 = A(2)
                ELSE
                DCORR1 = A(1)
                DCORR2 = A(2)
                END IF
        ELSE IF (NFLX .EQ. 1) THEN
                FLXG = A(1)
                IF (NSAT .EQ. 2) THEN
                DCORR2 = A(2)
                ELSE
                DCORR1 = A(2)
                END IF
        ELSE IF (NFLX .EQ. 2) THEN
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                ELSE
                DCORR1 = A(1)
                END IF
                FLXG = A(2)
        END IF
ELSE IF (NPARAM .EQ. 1) THEN
        IF (NFLX .EQ. 0) THEN
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                ELSE
                DCORR1 = A(1)
```

```
                DCORR2 = 1.0D0/DCORR1
                END IF
            ELSE
                FLXG = A(1)
            END IF
        END IF
*
*       40 microliters ****************************************************
*
        IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
        VOL = 40.0D-3
        VHTH = VOL/AREA - CORR
        DOSE = VOL*RHET/AREA
*
        DO 440 IT440 = 1, 5000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $              ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $              REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $              CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 4401 KL = 1,NPDE
            DO 4402 KM = 1,NPTS
                SOL(KM,KL) = U(NPDE*(KM-1)+KL)
4402        CONTINUE
4401    CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
```

```
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 4403 NN = 1, 10
            IF (T .LT. XDATA(NN)) THEN
            ELSE IF (T .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPEN
            ELSE
                  IF (TOLD .LT. XDATA(NN)) THEN
                  SLP = (PPEN - PPENOLD)/TDIF
                  YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
                  ELSE IF (TOLD .EQ. XDATA(NN)) THEN
                  YFIT(NN) = PPENOLD
                  END IF
            END IF
4403  CONTINUE
*
!     write(lsim,*) 'yfit(10), 4th pass =', yfit(10), 'ydata(10), 4th pass =', ydata(10)
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
      GO TO 4409
      ELSE
      END IF
      PPENOLD = PPEN
*
440   CONTINUE
*
*     END 40 micoliters ************************************************
*
*     20 microliters ***************************************************
*
4409  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
```

```
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 20.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      DO 420 IT420 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 4201 KL = 1,NPDE
            DO 4202 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
4202        CONTINUE
4201  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
```

```
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 4203 NN = 11, 20
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
4203  CONTINUE
*
!     write(lsim,*) 'yfit(20), 4th pass =', yfit(20), 'ydata(20), 4th pass =', ydata(20)
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
      GO TO 4209
      ELSE
      END IF
      PPENOLD = PPEN
*
420   CONTINUE
*
*     END 20 micoliters ************************************************
*
*     5 microliters ****************************************************
*
4209  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 5.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      DO 45 IT45 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
```

```
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 451 KL = 1,NPDE
            DO 452 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
452         CONTINUE
451   CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 453 NN = 21, 30
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
```

```
              YFIT(NN) = PPENOLD
              END IF
       END IF
453    CONTINUE
*
!      write(lsim,*) 'yfit(40), 4th pass =', yfit(40), 'ydata(40), 4th pass =', ydata(40)
*
       TOLD = T
       IF (THRS .GT. 72.0D0) THEN
       GO TO 459
       ELSE
       END IF
       PPENOLD = PPEN
*
*
45            CONTINUE
************************************************************************
*      END 4th Call to D03PPF
************************************************************************
*
459    CHISQ3 = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!      WRITE(lsim,*) 'end 4th NAG time looping'
       write(lsim,*) 'chisq3, 4th pass', chisq3, 'j =', j
       IF (CHISQ3 .LT. CHISQ2) THEN
       CHISQ1 = CHISQ2
       CHISQ2 = CHISQ3
       GO TO 61
       ELSE
       END IF
*
*      FIND MINIMUM OF PARABOLA DEFINED BY LAST TWO POINTS
*
       DELTA = DELTA*(1.0D0/(1.0D0 + (CHISQ1-CHISQ2)/(CHISQ3-CHISQ2)) + 0.5D0)
       A(J) = A(J) - DELTA
       SIGMAA(J) = DELTAA(J)*SQRT(2.0D0/(DBLE(NFREE)*(CHISQ3 - 2.0D0*CHISQ2
   +                + CHISQ1)))
       DELTAA(J) = DELTAA(J)*FN/3.0D0
90     CONTINUE
*
************************************************************************
*      5th Call to D03PPF - CHECK
************************************************************************
*
              DO 2050 MM = 1,NEXDP
              YFIT(MM) = 0.0D0
2050   CONTINUE
*
*
       IF (NPARAM .EQ. 3) THEN
              IF (NFLX .EQ. 1) THEN
              FLXG = A(1)
                     IF (NSAT .EQ. 2) THEN
                     DCORR2 = A(2)
```

```
                DCORR1 = A(3)
                ELSE
                DCORR1 = A(2)
                DCORR2 = A(3)
                END IF
        ELSE IF (NFLX .EQ. 2) THEN
        FLXG = A(2)
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                DCORR1 = A(3)
                ELSE
                DCORR1 = A(1)
                DCORR2 = A(3)
                END IF
        ELSE IF (NFLX .EQ. 3) THEN
        FLXG = A(3)
                IF (NSAT .EQ. 1) THEN
                DCORR1 = A(1)
                DCORR2 = A(2)
                ELSE
                DCORR2 = A(1)
                DCORR1 = A(2)
                END IF
        END IF
ELSE IF (NPARAM .EQ. 2) THEN
        IF (NFLX .EQ. 0) THEN
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                DCORR1 = A(2)
                ELSE
                DCORR1 = A(1)
                DCORR2 = A(2)
                END IF
        ELSE IF (NFLX .EQ. 1) THEN
                FLXG = A(1)
                IF (NSAT .EQ. 2) THEN
                DCORR2 = A(2)
                ELSE
                DCORR1 = A(2)
                END IF
        ELSE IF (NFLX .EQ. 2) THEN
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                ELSE
                DCORR1 = A(1)
                END IF
                FLXG = A(2)
        END IF
ELSE IF (NPARAM .EQ. 1) THEN
        IF (NFLX .EQ. 0) THEN
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                ELSE
```

```
                  DCORR1 = A(1)
                  DCORR2 = 1.0D0/DCORR1
                  END IF
            ELSE
                  FLXG = A(1)
            END IF
      END IF
*
*     40 microliters ******************************************************
*
      IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 40.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      DO 540 IT540 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 5401 KL = 1,NPDE
            DO 5402 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
5402        CONTINUE
5401  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
```

```
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 5403 NN = 1, 10
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
5403  CONTINUE
*
!     write(lsim,*) 'yfit(10), 5th pass =', yfit(10), 'ydata(10), 5th pass =', ydata(10)
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
      GO TO 5409
      ELSE
      END IF
      PPENOLD = PPEN
*
540   CONTINUE
*
*     END 40 micoliters ************************************************
*
*     20 microliters ***************************************************
*
5409  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
```

```
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 20.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      DO 520 IT520 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 5201 KL = 1,NPDE
            DO 5202 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
5202        CONTINUE
5201  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
```

```
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 5203 NN = 11, 20
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
5203  CONTINUE
*
!     write(lsim,*) 'yfit(20), 5th pass =', yfit(20), 'ydata(20), 5th pass =', ydata(20)
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
      GO TO 5209
      ELSE
      END IF
      PPENOLD = PPEN
*
520   CONTINUE
*
*     END 20 micoliters ****************************************************
*
*     5 microliters ********************************************************
*
5209  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 5.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      DO 55 IT55 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
```

```
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                  REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                  CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 551 KL = 1,NPDE
                DO 552 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
552             CONTINUE
551     CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHET
*
        ELSE
*
        FEVP = FLXG*SOL(NPTS,1)/ETSCIN
        TMVH = 0.0D0
        END IF
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 553 NN = 21, 30
                IF (T .LT. XDATA(NN)) THEN
                ELSE IF (T .EQ. XDATA(NN)) THEN
                YFIT(NN) = PPEN
                ELSE
                        IF (TOLD .LT. XDATA(NN)) THEN
                        SLP = (PPEN - PPENOLD)/TDIF
                        YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
```

```
                    ELSE IF (TOLD .EQ. XDATA(NN)) THEN
                    YFIT(NN) = PPENOLD
                    END IF
               END IF
553    CONTINUE
*
!      write(lsim,*) 'yfit(40), 5th pass =', yfit(40), 'ydata(40), 5th pass =', ydata(40)
*
       TOLD = T
       IF (THRS .GT. 72.0D0) THEN
       GO TO 559
       ELSE
       END IF
       PPENOLD = PPEN
*
*
55     CONTINUE
************************************************************************
*      END 5th Call to D03PPF
************************************************************************
*
559    CHISQR = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
       EPSCHI = ABS(100.0D0*(CHISQR-CHIOLD)/CHISQR)
       write(lsim,*) 'EPSCHI =', EPSCHI
       WRITE(lsim,*) 'chitol check', chitol
       IF (EPSCHI .LE. CHITOL) THEN
       GO TO 999
       ELSE
       END IF
       CHIOLD = CHISQR
998    CONTINUE
*
************************************************************************
*      Final Call to D03PPF
************************************************************************
*
999    IF (NPARAM .EQ. 3) THEN
              IF (NFLX .EQ. 1) THEN
              FLXG = A(1)
                     IF (NSAT .EQ. 2) THEN
                     DCORR2 = A(2)
                     DCORR1 = A(3)
                     ELSE
                     DCORR1 = A(2)
                     DCORR2 = A(3)
                     END IF
              ELSE IF (NFLX .EQ. 2) THEN
              FLXG = A(2)
                     IF (NSAT .EQ. 1) THEN
                     DCORR2 = A(1)
                     DCORR1 = A(3)
                     ELSE
                     DCORR1 = A(1)
```

```
                DCORR2 = A(3)
                END IF
        ELSE IF (NFLX .EQ. 3) THEN
        FLXG = A(3)
                IF (NSAT .EQ. 1) THEN
                DCORR1 = A(1)
                DCORR2 = A(2)
                ELSE
                DCORR2 = A(1)
                DCORR1 = A(2)
                END IF
        END IF
    ELSE IF (NPARAM .EQ. 2) THEN
        IF (NFLX .EQ. 0) THEN
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                DCORR1 = A(2)
                ELSE
                DCORR1 = A(1)
                DCORR2 = A(2)
                END IF
        ELSE IF (NFLX .EQ. 1) THEN
                FLXG = A(1)
                IF (NSAT .EQ. 2) THEN
                DCORR2 = A(2)
                ELSE
                DCORR1 = A(2)
                END IF
        ELSE IF (NFLX .EQ. 2) THEN
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                ELSE
                DCORR1 = A(1)
                END IF
                FLXG = A(2)
        END IF
    ELSE IF (NPARAM .EQ. 1) THEN
        IF (NFLX .EQ. 0) THEN
                IF (NSAT .EQ. 1) THEN
                DCORR2 = A(1)
                ELSE
                DCORR1 = A(1)
                DCORR2 = 1.0D0/DCORR1
                END IF
        ELSE
                FLXG = A(1)
        END IF
    END IF
*
    write (lsim,*) 'CHANGE IN FLXG =', FLXG*FCOR/FLXGOLD
    write (lsim,*) 'CHANGE IN DCORR1 =', DCORR1/DCORR1OLD
    write (lsim,*) 'CHANGE IN DCORR2 =', DCORR2/DCORR2OLD
    WRITE (lsim,*) 'FINAL FLXG =', FLXG*1.0D6
```

```
      WRITE (lsim,*) 'FINAL DCORR1 =', DCORR1
      WRITE (lsim,*) 'FINAL DCORR2 =', DCORR2
*
      WRITE (lsim,*) 'FINAL PREDICTIVE DIFFUSIVITY =', DWNK*DCORR1
      WRITE (lsim,*) 'FINAL SATURATION DIFFUSIVITY =', DWNK*DCORR1*DCORR2
*
      DO 102 I = 1,NPARAM
      IF (A(I) .LE. 0.0D0) THEN
      GO TO 659
      ELSE
      END IF
102   CONTINUE
*
*
*     40 microliters ******************************************************
*
      IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 40.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Initial Volume of Penetrant (in microlit)= ', VOL*1.0D3
      WRITE(lsim,*) 'Initial VH Thickness (in microns) = ', VHTH*1.0D4
      WRITE(lsim,*) 'Initial Mass of Penetrant (microg/cm2)= ', DOSE*1.0D6
*
      DO 640 IT640 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 6401 KL = 1,NPDE
            DO 6402 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
6402        CONTINUE
```

```
6401  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
      WRITE(llvt40,1004) THRS, VHTI
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
      CALL MASSBL (NPTS, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
      CFLX = CFLX + FTTL*TDIF ! Cumulative Flux
      CHECK = (TMAS + CFLX)/DOSE ! Should be ~ 1
      WRITE(lmbc40,1004) THRS, CHECK
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 6403 NN = 1, 10
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
6403  CONTINUE
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
```

```
      GO TO 6409
      ELSE
      END IF
      PPENOLD = PPEN
*
*     Writing the Solutions!
      WRITE(lAsc40,1001) THRS
      WRITE(lAsc40,1002) (SOL(J,1), J=1, NPTS)
      WRITE(lAve40,1001) THRS
      WRITE(lAve40,1002) (SOL(J,2), J=1, NPTS)
      WRITE(lflx40,1004) THRS, ABS(FEVP)*UFLX, ABS(FBTM)*UFLX, ABS(FCAL)*UFLX
      WRITE(lcmf40,1004) THRS, PEVP, PPEN, PTTL
*
640   CONTINUE
*
*     END 40 micoliters ***************************************************
*
*     20 microliters ******************************************************
*
6409  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 20.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Initial Volume of Penetrant (in microlit)= ', VOL*1.0D3
      WRITE(lsim,*) 'Initial VH Thickness (in microns) = ', VHTH*1.0D4
      WRITE(lsim,*) 'Initial Mass of Penetrant (microg/cm2)= ', DOSE*1.0D6
*
      DO 620 IT620 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 6201 KL = 1,NPDE
```

```
            DO 6202 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
6202        CONTINUE
6201  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHET
      WRITE(llvt20,1004) THRS, VHTI
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/ETSCIN
      TMVH = 0.0D0
      END IF
*
      CALL MASSBL (NPTS, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
      CFLX = CFLX + FTTL*TDIF ! Cumulative Flux
      CHECK = (TMAS + CFLX)/DOSE ! Should be ~ 1
      WRITE(lmbc20,1004) THRS, CHECK
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 6203 NN = 11, 20
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
6203  CONTINUE
```

```
*
      TOLD = T
      IF (THRS .GT. 72.0D0) THEN
      GO TO 6209
      ELSE
      END IF
      PPENOLD = PPEN
*
*     Writing the Solutions!
      WRITE(lAsc20,1001) THRS
      WRITE(lAsc20,1002) (SOL(J,1), J=1, NPTS)
      WRITE(lAve20,1001) THRS
      WRITE(lAve20,1002) (SOL(J,2), J=1, NPTS)
      WRITE(lflx20,1004) THRS, ABS(FEVP)*UFLX, ABS(FBTM)*UFLX, ABS(FCAL)*UFLX
      WRITE(lcmf20,1004) THRS, PEVP, PPEN, PTTL
*
620   CONTINUE
*
*     END 20 micoliters ************************************************
*
*     5 microliters ****************************************************
*
6209  IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      VOL = 5.0D-3
      VHTH = VOL/AREA - CORR
      DOSE = VOL*RHET/AREA
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Initial Volume of Penetrant (in microlit)= ', VOL*1.0D3
      WRITE(lsim,*) 'Initial VH Thickness (in microns) = ', VHTH*1.0D4
      WRITE(lsim,*) 'Initial Mass of Penetrant (microg/cm2)= ', DOSE*1.0D6
*
      DO 65 IT65 = 1, 5000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF
*
      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
```

```
*
*       Separating the Temperature and Concentration Solutions
        DO 651 KL = 1,NPDE
                DO 652 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
652             CONTINUE
651     CONTINUE
*
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!       FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHET
        WRITE(llvt5,1004) THRS, VHTI
*
        ELSE
*
        FEVP = FLXG*SOL(NPTS,1)/ETSCIN
        TMVH = 0.0D0
        END IF
*
        CALL MASSBL (NPTS, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
        CFLX = CFLX + FTTL*TDIF ! Cumulative Flux
        CHECK = (TMAS + CFLX)/DOSE ! Should be ~ 1
        WRITE(lmbc5,1004) THRS, CHECK
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 653 NN = 21, 30
        IF (T .LT. XDATA(NN)) THEN
        ELSE IF (T .EQ. XDATA(NN)) THEN
        YFIT(NN) = PPEN
        ELSE
                IF (TOLD .LT. XDATA(NN)) THEN
                SLP = (PPEN - PPENOLD)/TDIF
                YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
                ELSE IF (TOLD .EQ. XDATA(NN)) THEN
                YFIT(NN) = PPENOLD
```

```
            END IF
        END IF
653     CONTINUE
*
        TOLD = T
        IF (THRS .GT. 72.0D0) THEN
        CHISQR = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
        WRITE(lsim,*) '************************************************'
        write (lsim,*) 'NFREE =', NFREE
        WRITE (lsim,*) 'FINAL CHISQ (*10^5) =', CHISQR*1.0D5
        WRITE (lsim,*) 'FINAL SSR =', CHISQR*DBLE(NFREE)
        GO TO 659
        ELSE
        END IF
        PPENOLD = PPEN
*
*       Writing the Solutions!
        WRITE(lAsc5,1001) THRS
        WRITE(lAsc5,1002) (SOL(J,1), J=1, NPTS)
        WRITE(lAve5,1001) THRS
        WRITE(lAve5,1002) (SOL(J,2), J=1, NPTS)
        WRITE(lflx5,1004) THRS, ABS(FEVP)*UFLX, ABS(FBTM)*UFLX, ABS(FCAL)*UFLX
        WRITE(lcmf5,1004) THRS, PEVP, PPEN, PTTL
*
65          CONTINUE
*************************************************************************
*       END Final Call to D03PPF
*************************************************************************
*
*
1001    FORMAT('Time (s) ', F24.12)
1002    FORMAT(5(F18.12,1X))
1003    FORMAT(F36.12, 2X, F18.12, 2X, F18.12)
1004    FORMAT(F36.12, 2X, 6(F36.12))
*
659         CLOSE(lsim, STATUS='KEEP')
            CLOSE(lAsc40, STATUS='KEEP')
            CLOSE(lAsc20, STATUS='KEEP')
!           CLOSE(lAsc10, STATUS='KEEP')
            CLOSE(lAsc5, STATUS='KEEP')
            CLOSE(lAve40, STATUS='KEEP')
            CLOSE(lAve20, STATUS='KEEP')
!           CLOSE(lAve10, STATUS='KEEP')
            CLOSE(lAve5, STATUS='KEEP')
            CLOSE(llvt40, STATUS='KEEP')
            CLOSE(llvt20, STATUS='KEEP')
!           CLOSE(llvt10, STATUS='KEEP')
            CLOSE(llvt5, STATUS='KEEP')
            CLOSE(lflx40, STATUS='KEEP')
            CLOSE(lflx20, STATUS='KEEP')
!           CLOSE(lflx10, STATUS='KEEP')
            CLOSE(lflx5, STATUS='KEEP')
            CLOSE(lmbc40, STATUS='KEEP')
```

```
      CLOSE(lmbc20, STATUS='KEEP')
!     CLOSE(lmbc10, STATUS='KEEP')
      CLOSE(lmbc5, STATUS='KEEP')
      CLOSE(lcmf40, STATUS='KEEP')
      CLOSE(lcmf20, STATUS='KEEP')
!     CLOSE(lcmf10, STATUS='KEEP')
      CLOSE(lcmf5, STATUS='KEEP')
!     CLOSE(lext, STATUS='KEEP')
*
      STOP
      END
```

Program BZDINDCHIF simulates the evaporation and absorption of ethanol through skin sub layers for individual doses and fits them to the corresponding experimental data on skin permeation of benzene. This results in a dose-dependent variable fractional deposition depth. This program can be used for ethanol by changing the physicochemical properties of and transport parameters.

```
      PROGRAM BZDINDCHIF
*
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      IMPLICIT INTEGER (I-N)
*
*     NPDE   = Number of PDEs
*     NCODE  = Number of Coupled ODEs
*     M      = Determines the Coordinate system (0 = Rectangular Cartesian)
*     NPTS   = Number of Mesh Points
*     NXI    = Number of PDE-ODE Coupling Points
*     NXFIX  = Number of Fixed Mesh Points
      PARAMETER (NPDE=2, NCODE=1, M=0, NPTS=201, NXI=0, NXFIX=0)
*
*     DO NOT CHANGE THE LINES BELOW!
      PARAMETER (NWKRES = NPDE*(3*NPDE+6*NXI+NPTS+15)+ NXI+NCODE
     $                  + 7*NPTS+NXFIX+2+9000)
!     Increase the value 9000 if more workspace is needed
      PARAMETER (NEQN=NPDE*NPTS+NCODE)
      PARAMETER (NIW=25*NEQN+25+NXFIX+90000)
!     Dimension of the array IW. Pg. 17. DO NOT CHANGE unless IFAIL 15!
      PARAMETER (LENODE=(6+5)*NEQN+50) ! For theta method, Pg. 17!
      PARAMETER (NW=4*NEQN+11*NEQN/2+1+NWKRES+LENODE+20000)
!     DO NOT CHANGE unless IFAIL 15 !
*
      PARAMETER (NFIT = 1) ! Keep a non-zero value for optimization
      PARAMETER (NEXDP = 10)
      PARAMETER (NPARAM = 1)
      PARAMETER (NDEP = 0)
      PARAMETER (NDIF = 1)
      PARAMETER (MODE = 0)
*
      DOUBLE PRECISION U(NEQN), X(NEQN), RTOL(NEQN), ATOL(NEQN), XI(1),
     +                 XFIX(1), IW(NIW), W(NW), ALGOPT(30),
     +                 XDATA(NEXDP), YDATA(NEXDP), YFIT(NEXDP),
     +                 SIGMAY (NEXDP), A(NEXDP), DELTAA(NPARAM),
SIGMAA(NEXDP),
     +                 DA(NEXDP), ALPHA(NEXDP,NEXDP), BETA(NEXDP),
SOL(NPTS,NPDE)
*     IW(NIW) - Output variable, W(NW) - Workspace variable
      INTEGER ITIME(7)
*
      LOGICAL REMESH, THETA
      CHARACTER*1 LAOPT, NORM
*
*     External Subroutines
      EXTERNAL UVINIT,ODEDEF,PDEDEF,BNDARY,D03PPF,D03PCL,D03PCK,X05AAF
```

```
!       D03PCL replaces MONITF as dummy subroutine!
*
*       Common Blocks
        COMMON /EXPMT/ TEMP, VOL, DOSE, VHTH, BZSCIN
        COMMON /VHPAR/ RHBZ, WMBZ
        COMMON /SCPAR/ RHSC, SCTH, DASC, FDEP
        COMMON /VEPAR/ RHVT, VTTH, DAVT
        COMMON /INTER/ FLXG, FLSCVH, FLVESC, FBTM, SCVHPC, VTSCPC, PTOR
        COMMON /VPCPH/ VPCA, VPCB, VPCC, VPCD, VPCE
        COMMON /CRRCT/ FCOR, DCOR, FUBZ, SCWRPC, VTWRPC
*
*       Initialize the constants specific to the system of interest
        CALL CONSTS
*
*       Write the Solution Files
        lsim = 11 ! Description of the Simulation
        lAsc = 12 ! Mass Fraction of Solvent in the Stratum Corneum !
        lAve = 13 ! Mass Fraction of Solvent in the Viable Epidermis !
        llvt = 14 ! Instantaneous Thickness of the Donor Solution !
        lflx = 15 ! Fluxes
        lmbc = 16 ! Mass balance component
        lcmf = 17 ! Cumulative Fluxes
        lext = 18 ! Anciliary data
*
        itrace = 2 ! See D03PPF Manual, Pg. 18 !
*
        OPEN (lsim, file='sim.dat', status='new')
        OPEN (lAsc, file='Asc.dat', status='new')
        OPEN (lAve, file='Ave.dat', status='new')
        OPEN (llvt, file='lvt.dat', status='new')
        OPEN (lflx, file='flx.dat', status='new')
        OPEN (lmbc, file='mbc.dat', status='new')
        OPEN (lcmf, file='cmf.dat', status='new')
!       OPEN (lext, file='fit.dat', status='new')
*
*
*       Experimental Data
*
        XDATA(1)  = 0.08D0*3600.0D0
        XDATA(2)  = 0.17D0*3600.0D0
        XDATA(3)  = 0.25D0*3600.0D0
        XDATA(4)  = 0.33D0*3600.0D0
        XDATA(5)  = 0.67D0*3600.0D0
        XDATA(6)  = 1.00D0*3600.0D0
        XDATA(7)  = 2.00D0*3600.0D0
        XDATA(8)  = 4.00D0*3600.0D0
        XDATA(9)  = 8.50D0*3600.0D0
        XDATA(10) = 24.00D0*3600.0D0
*
        IF (VOL .EQ. 5.0D-3) THEN
        write (lsim,*) 'using experimental data for 5 microliters'
        YDATA(1)  = 0.01D0
        YDATA(2)  = 0.02D0
```

```
      YDATA(3)  = 0.02D0
      YDATA(4)  = 0.03D0
      YDATA(5)  = 0.04D0
      YDATA(6)  = 0.05D0
      YDATA(7)  = 0.05D0
      YDATA(8)  = 0.06D0
      YDATA(9)  = 0.06D0
      YDATA(10) = 0.06D0
*
      ELSE IF (VOL .EQ. 10.0D-3) THEN
      write (lsim,*) 'using experimental data for 10 microliters'
      YDATA(1)  = 0.00D0
      YDATA(2)  = 0.01D0
      YDATA(3)  = 0.02D0
      YDATA(4)  = 0.02D0
      YDATA(5)  = 0.03D0
      YDATA(6)  = 0.04D0
      YDATA(7)  = 0.04D0
      YDATA(8)  = 0.05D0
      YDATA(9) = 0.05D0
      YDATA(10) = 0.05D0
*
      ELSE IF (VOL .EQ. 20.0D-3) THEN
      write (lsim,*) 'using experimental data for 20 microliters'
      YDATA(1)  = 0.00D0
      YDATA(2)  = 0.01D0
      YDATA(3)  = 0.02D0
      YDATA(4)  = 0.02D0
      YDATA(5)  = 0.03D0
      YDATA(6)  = 0.04D0
      YDATA(7)  = 0.05D0
      YDATA(8)  = 0.05D0
      YDATA(9)  = 0.05D0
      YDATA(10) = 0.05D0
*
      ELSE IF (VOL .EQ. 40.0D-3) THEN
      write (lsim,*) 'using experimental data for 40 microliters'
      YDATA(1)  = 0.00D0
      YDATA(2)  = 0.01D0
      YDATA(3)  = 0.02D0
      YDATA(4)  = 0.02D0
      YDATA(5)  = 0.03D0
      YDATA(6)  = 0.04D0
      YDATA(7)  = 0.04D0
      YDATA(8)  = 0.04D0
      YDATA(9)  = 0.04D0
      YDATA(10) = 0.05D0
*
      ELSE
      END IF
*
      WRITE(lsim,*) ' 3-Layer Pure Penetrant Isothermal Simulation '
      WRITE(lsim,*) ' The Penetrant in BENZENE'
```

```
*
      WRITE(lsim,*) 'Number of Mesh Points = ', NPTS
      NDPD = ((NPTS-1)*FDEP) + 1
      WRITE(lsim,*) 'Number of Permeant Deposited Mesh Points = ', NDPD
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Molecular Weight of the Penetrant = ', WMBZ
      WRITE(lsim,*) 'Constant System Temperature = ', TEMP - 273.15D0
      WRITE(lsim,*) 'Density of Penetrant = ', RHBZ
      WRITE(lsim,*) 'Initial Volume of Penetrant (in microlit)= ', VOL*1.0D3
      WRITE(lsim,*) 'Initial Mass of Penetrant (microg/cm2)= ', DOSE*1.0D6
      WRITE(lsim,*) 'Initial VH Thickness (in microns) = ', VHTH*1.0D4
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Density of SC Phase = ', RHSC
      WRITE(lsim,*) 'Thickness of SC Phase (in microns) = ', SCTH*1.0D4
      WRITE(lsim,*) 'Corrected Diffusivity of A in SC = ', DASC
      WRITE(lsim,*) 'Initial Correction Factor for DASC = ', DCOR
      WRITE(lsim,*) 'Predicted Diffusivity of A in SC = ', DASC/DCOR
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Density of VT Phase = ', RHVT
      WRITE(lsim,*) 'Thickness of VT Phase (in microns) = ', VTTH*1.0D4
      WRITE(lsim,*) 'Diffusivity of A in VT = ', DAVT
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Vapor Pressure in Torr = ', PTOR
      WRITE(lsim,*) 'Corrected KEVAP = ', FLXG*3600.0D0/RHBZ
      WRITE(lsim,*) 'Initial Correction Factor for FLXG = ', FCOR
      WRITE(lsim,*) 'Predicted KEVAP = ', FLXG*3600.0D0/(FCOR*RHBZ)
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Fractional Deposition Depth (%)= ', FDEP*100.0D0
      WRITE(lsim,*) 'SC-VH Partition Coefficient of Ethanol = ', SCVHPC
      WRITE(lsim,*) 'VT-SC Partition Coefficient of Ethanol = ', VTSCPC
      WRITE(lsim,*) 'SC-Water Partition Coefficient of Benzene = ', SCWRPC
      WRITE(lsim,*) 'VT-Water Partition Coefficient of Benzene = ', VTWRPC
*
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Number of Fitted Parameters = ', NPARAM
      WRITE(lsim,*) 'Weighting Coefficient = ', MODE
      WRITE(lsim,*) 'Fitting the Deposition Depth = ', NDEP
      WRITE(lsim,*) 'Fitting the SC Diffusivity = ', NDIF
*
*     Initialize Mesh - Uniform Mesh
*
      DO 910 I = 1, NPTS
            X(I) = DBLE(I-1)/(NPTS-1.0D0)
910   CONTINUE
*
*     Set Remesh Parameters
*
      REMESH = .false.
```

```
      NRMESH = 0
      DXMESH = 0.0D0
      TRMESH = 1.0D0
      IPMINF = 0
      XRATIO = 1.5D0
      CONST  = 2.0D0/(NPTS-1)
      IND = 0
      ITASK = 2
      NORM = 'A'
      LAOPT= 'S' !use the sparse matrix routines
      THETA = .false.
*
      DO 920 I=1,30
            ALGOPT(I)=0.0D0
920   CONTINUE
      ALGOPT(4)=2.0D0 !Does not perform the Petzold test
      ALGOPT(29)=0.5 !NAG Deafault = 0.1
      ALGOPT(30)=1.0D-300 !NAG Default 1.0D-2!
*
*     Set Time Step & Convergence Criterion
*
      DELT = 1.0D-3
      WRITE(lsim,*) '*****************************************************'
      WRITE(lsim,*) 'Time Increament Step Size =', DELT
      DO 930 K = 1, NEQN
      RTOL(K) = 1.0D-8
      ATOL(K) = 1.0D-8
930   CONTINUE
      ITOL = 4
      WRITE(lsim,*) 'RTOL = ', RTOL(1)
      WRITE(lsim,*) 'ATOL = ', ATOL(1)
      WRITE(lsim,*) 'ITOL = ', ITOL
*
*     Write Simulation Date and Time
      CALL X05AAF(ITIME)
      WRITE(lsim,*) 'Month / Day / Year = ', ITIME(2),'/',ITIME(3),'/',ITIME(1)
      WRITE(lsim,*) 'Hour / Minute / Second = ', ITIME(4),'/',ITIME(5),'/',ITIME(6)
*
*
      FLXGOLD = FLXG
      DASCOLD = DASC
      FDEPOLD = FDEP
*
      IF (NPARAM .EQ. 3) THEN
            IF (NDEP .EQ. 1) THEN
            A(1) = FDEP
                  IF (NDIF .EQ. 2) THEN
                  A(2) = DASC
                  A(3) = FLXG
                  ELSE
                  A(2) = FLXG
                  A(3) = DASC
                  END IF
```

```
        ELSE IF (NDEP .EQ. 2) THEN
        A(2) = FDEP
                IF (NDIF .EQ. 1) THEN
                A(1) = DASC
                A(3) = FLXG
                ELSE
                A(1) = FLXG
                A(3) = DASC
                END IF
        ELSE IF (NDEP .EQ. 3) THEN
        A(3) = FDEP
                IF (NDIF .EQ. 1) THEN
                A(1) = DASC
                A(2) = FLXG
                ELSE
                A(1) = FLXG
                A(2) = DASC
                END IF
        END IF
ELSE IF (NPARAM .EQ. 2) THEN
        IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                A(1) = DASC
                A(2) = FLXG
                ELSE
                A(1) = FLXG
                A(2) = DASC
                END IF
        ELSE IF (NDEP .EQ. 1) THEN
                A(1) = FDEP
                IF (NDIF .EQ. 2) THEN
                A(2) = DASC
                ELSE
                A(2) = FLXG
                END IF
        ELSE IF (NDEP .EQ. 2) THEN
                IF (NDIF .EQ. 1) THEN
                A(1) = DASC
                ELSE
                A(1) = FLXG
                END IF
                A(2) = FDEP
        END IF
ELSE IF (NPARAM .EQ. 1) THEN
        IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                A(1) = DASC
                ELSE
                A(1) = FLXG
                END IF
        ELSE
                A(1) = FDEP
        END IF
```

```
      END IF
*
      DO 940 LL = 1,NPARAM
      DELTAA(LL) = 0.1D0*A(LL)
940   CONTINUE
      DO 941 LM = 1,NEXDP
      SIGMAY(LM) = 1.0D0
941   CONTINUE
*
*
      NFREE = NEXDP - NPARAM
*
      IF (NFIT .EQ. 0) THEN
      GO TO 999
      ELSE
      WRITE (lsim,*) 'RUNNING THE CHIFIT OPTIMIZER'
      END IF
*
*
*     Optimization with CHIFIT
*
      CHITOL = 1.0D-1
      write(lsim,*) 'CHITOL =', CHITOL
      CHIOLD = 0.0D0
*
      DO 998 NOPT = 1, 1000
      write(lsim,*) 'NOPT =', NOPT
*
*************************************************************************
*     1st Call to D03PPF
*************************************************************************
*
      IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      IF (NPARAM .EQ. 3) THEN
            IF (NDEP .EQ. 1) THEN
            FDEP = A(1)
                  IF (NDIF .EQ. 2) THEN
                  DASC = A(2)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(2)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 2) THEN
            FDEP = A(2)
```

```
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                FLXG = A(3)
                ELSE
                FLXG = A(1)
                DASC = A(3)
                END IF
        ELSE IF (NDEP .EQ. 3) THEN
        FDEP = A(3)
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                FLXG = A(2)
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
        END IF
ELSE IF (NPARAM .EQ. 2) THEN
        IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                FLXG = A(2)
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
        ELSE IF (NDEP .EQ. 1) THEN
                FDEP = A(1)
                IF (NDIF .EQ. 2) THEN
                DASC = A(2)
                ELSE
                FLXG = A(2)
                END IF
        ELSE IF (NDEP .EQ. 2) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
                FDEP = A(2)
        END IF
ELSE IF (NPARAM .EQ. 1) THEN
        IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
        ELSE
                FDEP = A(1)
        END IF
END IF
*
```

```
      DO 201 IT1 = 1, 3000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF

      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)

*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 2011 KL = 1,NPDE
            DO 2012 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2012        CONTINUE
2011  CONTINUE
*
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/BZSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 2013 NN = 1, NEXDP
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
```

```
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
2013  CONTINUE
*
!     write(lsim,*) 'yfit(11), 1st pass =', yfit(11)
*
      TOLD = T
      IF (THRS .GT. 2400.0D0) THEN
      GO TO 2019
      ELSE
      END IF
      PPENOLD = PPEN
*
*
201   CONTINUE
************************************************************************
*     END 1st Call to D03PPF
************************************************************************
*
2019  CHISQ1 = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!     write(lsim,*) 'end 1st NAG time looping'
*
*     Evaluate the alpha and beta matrices
*
      DO 60 J = 1, NPARAM
      AJ = A(J)
      A(J) = AJ + DELTAA(J)
!     write(lsim,*) 'j =', j, 'change in a(j) =', a(j)/aj
*
************************************************************************
*     2nd Call to D03PPF
************************************************************************
*
      IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
            DO 2020 MM = 1,NEXDP
            YFIT(MM) = 0.0D0
2020  CONTINUE
*
```

```
IF (NPARAM .EQ. 3) THEN
        IF (NDEP .EQ. 1) THEN
        FDEP = A(1)
                IF (NDIF .EQ. 2) THEN
                DASC = A(2)
                FLXG = A(3)
                ELSE
                FLXG = A(2)
                DASC = A(3)
                END IF
        ELSE IF (NDEP .EQ. 2) THEN
        FDEP = A(2)
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                FLXG = A(3)
                ELSE
                FLXG = A(1)
                DASC = A(3)
                END IF
        ELSE IF (NDEP .EQ. 3) THEN
        FDEP = A(3)
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                FLXG = A(2)
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
        END IF
ELSE IF (NPARAM .EQ. 2) THEN
        IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                FLXG = A(2)
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
        ELSE IF (NDEP .EQ. 1) THEN
                FDEP = A(1)
                IF (NDIF .EQ. 2) THEN
                DASC = A(2)
                ELSE
                FLXG = A(2)
                END IF
        ELSE IF (NDEP .EQ. 2) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
                FDEP = A(2)
        END IF
```

```
      ELSE IF (NPARAM .EQ. 1) THEN
            IF (NDEP .EQ. 0) THEN
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  ELSE
                  FLXG = A(1)
                  END IF
            ELSE
                  FDEP = A(1)
            END IF
      END IF
*
!     write(lsim,*) ' DASC, 2nd pass =', DASC/DASCOLD
      DO 202 IT2 = 1, 3000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF

      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)
*
*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 2021 KL = 1,NPDE
            DO 2022 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2022        CONTINUE
2021  CONTINUE
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!!    FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/BZSCIN
      TMVH = 0.0D0
```

```
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 2023 NN = 1, NEXDP
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
2023  CONTINUE
*
!     write(lsim,*) 'yfit(11), 2nd pass =', yfit(11), 'it2 =', it2
*
      TOLD = T
      IF (THRS .GT. 2400.0D0) THEN
      GO TO 2029
      ELSE
      END IF
      PPENOLD = PPEN
*
*
202   CONTINUE
**********************************************************************
*     END 2nd Call to D03PPF
**********************************************************************
*
2029  CHISQ2 = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!     write(lsim,*) 'end 2nd NAG time looping'
!     write(lsim,*) 'chisq2, 2nd pass', chisq2, 'j =', j
      ALPHA(J,J) = CHISQ2 - 2.0D0*CHISQ1
      BETA(J) = -CHISQ2
      DO 50 K = 1, NPARAM
      IF (K .LT. J) THEN
!     write(lsim,*) 'k =', k, 'j =', j, 'k is less than j ... so no pass'
      ALPHA(K,J) = (ALPHA(K,J) - CHISQ2)/2.0D0
      ALPHA(J,K) = ALPHA(K,J)
      ELSE IF (K .GT. J) THEN
!     write (lsim,*) 'k =', k, 'j =', j, 'k is greater than j ... call d03ppf'
      ALPHA(J,K) = CHISQ1 - CHISQ2
      AK = A(K)
      A(K) = AK + DELTAA(K)
```

```
*
************************************************************************
*      3rd Call to D03PPF
************************************************************************
*
      IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
            DO 2030 MM = 1,NEXDP
            YFIT(MM) = 0.0D0
2030  CONTINUE
*
      IF (NPARAM .EQ. 3) THEN
            IF (NDEP .EQ. 1) THEN
            FDEP = A(1)
                  IF (NDIF .EQ. 2) THEN
                  DASC = A(2)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(2)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 2) THEN
            FDEP = A(2)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(1)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 3) THEN
            FDEP = A(3)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(2)
                  ELSE
                  FLXG = A(1)
                  DASC = A(2)
                  END IF
            END IF
      ELSE IF (NPARAM .EQ. 2) THEN
            IF (NDEP .EQ. 0) THEN
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(2)
                  ELSE
```

```
                FLXG = A(1)
                DASC = A(2)
                END IF
            ELSE IF (NDEP .EQ. 1) THEN
                FDEP = A(1)
                IF (NDIF .EQ. 2) THEN
                DASC = A(2)
                ELSE
                FLXG = A(2)
                END IF
            ELSE IF (NDEP .EQ. 2) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
                FDEP = A(2)
            END IF
        ELSE IF (NPARAM .EQ. 1) THEN
            IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
            ELSE
                FDEP = A(1)
            END IF
        END IF
!       write(lsim,*) ' FLXG, 3rd pass =', DASC/DASCOLD
*
        DO 203 IT3 = 1, 3000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*       Entering D03PPF

        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $              ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $              REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $              CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)

*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 2031 KL = 1,NPDE
            DO 2032 KM = 1,NPTS
                SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2032        CONTINUE
2031    CONTINUE
*
        VHTI = U(NEQN)/VHTH
```

```
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!!    FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/BZSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 2033 NN = 1, NEXDP
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
2033        CONTINUE
*
!     write(lsim,*) 'yfit(11), 3rd pass =', yfit(11), 'it3 =', it3
*
      TOLD = T
      IF (THRS .GT. 2400.0D0) THEN
      GO TO 2039
      ELSE
      END IF
      PPENOLD = PPEN
*
*
203   CONTINUE
**********************************************************************
```

```
*       END 3rd Call to D03PPF
*************************************************************************
*
2039  CHISQ3 = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!       write(lsim,*) 'end 3rd NAG time looping'
!       write(lsim,*) 'chisq3, 3rd pass', chisq3
        ALPHA(J,K) = ALPHA(J,K) + CHISQ3
        A(K) = AK
        ELSE
!       write (lsim,*) 'k =', k, 'j =', j, 'k is equal to j'
        END IF
50      CONTINUE
        A(J) = AJ - DELTAA(J)
*
*************************************************************************
*       4th Call to D03PPF
*************************************************************************
*
        IND  = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
              DO 2040 MM = 1,NEXDP
              YFIT(MM) = 0.0D0
2040  CONTINUE
*
        IF (NPARAM .EQ. 3) THEN
              IF (NDEP .EQ. 1) THEN
              FDEP = A(1)
                    IF (NDIF .EQ. 2) THEN
                    DASC = A(2)
                    FLXG = A(3)
                    ELSE
                    FLXG = A(2)
                    DASC = A(3)
                    END IF
              ELSE IF (NDEP .EQ. 2) THEN
              FDEP = A(2)
                    IF (NDIF .EQ. 1) THEN
                    DASC = A(1)
                    FLXG = A(3)
                    ELSE
                    FLXG = A(1)
                    DASC = A(3)
                    END IF
              ELSE IF (NDEP .EQ. 3) THEN
              FDEP = A(3)
                    IF (NDIF .EQ. 1) THEN
```

```
                DASC = A(1)
                FLXG = A(2)
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
            END IF
        ELSE IF (NPARAM .EQ. 2) THEN
            IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                FLXG = A(2)
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
            ELSE IF (NDEP .EQ. 1) THEN
                FDEP = A(1)
                IF (NDIF .EQ. 2) THEN
                DASC = A(2)
                ELSE
                FLXG = A(2)
                END IF
            ELSE IF (NDEP .EQ. 2) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
                FDEP = A(2)
            END IF
        ELSE IF (NPARAM .EQ. 1) THEN
            IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
            ELSE
                FDEP = A(1)
            END IF
        END IF
!       write(lsim,*) ' DASC, 4th pass =', DASC/DASCOLD
*
        DO 204 IT4 = 1, 3000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*       Entering D03PPF

        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                  ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
```

```
     $                REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)

*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 2041 KL = 1,NPDE
                DO 2042 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2042            CONTINUE
2041    CONTINUE
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!!      FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHBZ
*
        ELSE
*
        FEVP = FLXG*SOL(NPTS,1)/BZSCIN
        TMVH = 0.0D0
        END IF
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 2043 NN = 1, NEXDP
        IF (T .LT. XDATA(NN)) THEN
        ELSE IF (T .EQ. XDATA(NN)) THEN
        YFIT(NN) = PPEN
        ELSE
                IF (TOLD .LT. XDATA(NN)) THEN
                SLP = (PPEN - PPENOLD)/TDIF
                YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
                ELSE IF (TOLD .EQ. XDATA(NN)) THEN
                YFIT(NN) = PPENOLD
                END IF
        END IF
2043            CONTINUE
```

```
*
!       write(lsim,*) 'yfit(11), 4th pass =', yfit(11), 'it4 =', it4
*
        TOLD = T
        IF (THRS .GT. 2400.0D0) THEN
        GO TO 2049
        ELSE
        END IF
        PPENOLD = PPEN
*
*
204     CONTINUE
*************************************************************************
*       END 4th Call to D03PPF
*************************************************************************
*
2049    CHISQ3 = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!       write(lsim,*) '4th pass', yfit(9), ydata(9), yfit(11), ydata(11)
!       WRITE(lsim,*) 'end 4th NAG time looping'
!       write(lsim,*) 'chisq3, 4th pass', chisq3, 'j =', j
        A(J) = AJ
        ALPHA(J,J) = (ALPHA(J,J) + CHISQ3)/2.0D0
        BETA(J) = (BETA(J) + CHISQ3)/4.0D0
60      CONTINUE
*
*       Eliminate Negative Curvature
*
        DO 70 J = 1, NPARAM
        IF (ALPHA(J,J) .LT. 0.0D0) THEN
        ALPHA(J,J) = -ALPHA(J,J)
        ELSE IF (ALPHA(J,J) .EQ. 0.0D0) THEN
        ALPHA(J,J) = 0.01D0
        ELSE
        GOTO 70
        END IF
                DO 71 K = 1, NPARAM
                IF (K .NE. J) THEN
                ALPHA(J,K) = 0.0D0
                ALPHA(K,J) = 0.0D0
                ELSE
                END IF
71              CONTINUE
70      CONTINUE
*
*       Matrix Inversion and Parameter Increment Evaluations
*
        CALL MATINV (ALPHA, NPARAM, DET)
        DO 76 J = 1, NPARAM
        DA(J) = 0.0D0
                DO 75 K = 1, NPARAM
                DA(J) = DA(J) + BETA(K)*ALPHA(J,K)
75              CONTINUE
        DA(J) = 0.2D0*DA(J)*DELTAA(J)
```

```
76      CONTINUE
!       write(lsim,*) 'da2', da(2)
*
*       Making Sure CHI Square Decreases
*
        DO 82 J = 1, NPARAM
              A(J) = A(J) + DA(J)
82      CONTINUE
!       write(lsim,*) 'a2 =', a(2)
*
************************************************************************
*       5th Call to D03PPF
************************************************************************
*
83      IF (NPARAM .EQ. 3) THEN
              IF (NDEP .EQ. 1) THEN
              FDEP = A(1)
                    IF (NDIF .EQ. 2) THEN
                    DASC = A(2)
                    FLXG = A(3)
                    ELSE
                    FLXG = A(2)
                    DASC = A(3)
                    END IF
              ELSE IF (NDEP .EQ. 2) THEN
              FDEP = A(2)
                    IF (NDIF .EQ. 1) THEN
                    DASC = A(1)
                    FLXG = A(3)
                    ELSE
                    FLXG = A(1)
                    DASC = A(3)
                    END IF
              ELSE IF (NDEP .EQ. 3) THEN
              FDEP = A(3)
                    IF (NDIF .EQ. 1) THEN
                    DASC = A(1)
                    FLXG = A(2)
                    ELSE
                    FLXG = A(1)
                    DASC = A(2)
                    END IF
              END IF
        ELSE IF (NPARAM .EQ. 2) THEN
              IF (NDEP .EQ. 0) THEN
                    IF (NDIF .EQ. 1) THEN
                    DASC = A(1)
                    FLXG = A(2)
                    ELSE
                    FLXG = A(1)
                    DASC = A(2)
                    END IF
              ELSE IF (NDEP .EQ. 1) THEN
```

```
                FDEP = A(1)
                IF (NDIF .EQ. 2) THEN
                DASC = A(2)
                ELSE
                FLXG = A(2)
                END IF
            ELSE IF (NDEP .EQ. 2) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
                FDEP = A(2)
            END IF
        ELSE IF (NPARAM .EQ. 1) THEN
            IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
            ELSE
                FDEP = A(1)
            END IF
        END IF
!       write(lsim,*) ' DASC, 5th pass =', DASC/DASCOLD
*
        IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
            DO 2050 MM = 1,NEXDP
            YFIT(MM) = 0.0D0
2050    CONTINUE
*
        DO 205 IT5 = 1, 3000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*       Entering D03PPF
*
        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $              ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $              REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $              CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)

*       Leaving D03PPF
```

```
*
*       Separating the Temperature and Concentration Solutions
        DO 2051 KL = 1,NPDE
                DO 2052 KM = 1,NPTS
                        SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2052            CONTINUE
2051    CONTINUE
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
        FPEN = FCAL
!!      FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
        THRS = T/3600.0D0
        TDIF = T - TOLD ! Trapezoidal (b-a)
*
        IF (VHTI .GE. 0.0D0) THEN
        FEVP = FLXG
        TMVH = U(NEQN)*RHBZ
*
        ELSE
*
        FEVP = FLXG*SOL(NPTS,1)/BZSCIN
        TMVH = 0.0D0
        END IF
*
!       Amounts Evaporated and Penetrated
        CEVP = CEVP + (ABS(FEVP)*TDIF)
        CPEN = CPEN + (ABS(FPEN)*TDIF)
        PEVP = CEVP*100.0D0/DOSE
        PPEN = CPEN*100.0D0/DOSE
        PTTL = PEVP + PPEN
*
                DO 2053 NN = 1, NEXDP
        IF (T .LT. XDATA(NN)) THEN
        ELSE IF (T .EQ. XDATA(NN)) THEN
        YFIT(NN) = PPEN
        ELSE
                IF (TOLD .LT. XDATA(NN)) THEN
                SLP = (PPEN - PPENOLD)/TDIF
                YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
                ELSE IF (TOLD .EQ. XDATA(NN)) THEN
                YFIT(NN) = PPENOLD
                END IF
        END IF
2053            CONTINUE
*
!       write(lsim,*) 'yfit(11), 5th pass =', yfit(11), 'it5 =', it5
*
        TOLD = T
```

```
      IF (THRS .GT. 2400.0D0) THEN
      GO TO 2059
      ELSE
      END IF
      PPENOLD = PPEN
*
*
205   CONTINUE
************************************************************************
*     END 5th Call to D03PPF
************************************************************************
*
2059  CHISQ2 = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!     WRITE(lsim,*) 'end 5th NAG time looping'
!     write(lsim,*) 'chisq2, 5th pass', chisq2
      IF (CHISQ1 .LT. CHISQ2) THEN
      DO 89 J = 1,NPARAM
      DA(J) = DA(J)/2.0D0
            A(J) = A(J) - DA(J)
89    CONTINUE
      GO TO 83
      ELSE
      END IF
*
*     Increase Parameter until CHI Square starts to increase
*
91    DO 92 J = 1, NPARAM
      A(J) = A(J) + DA(J)
92    CONTINUE
*
************************************************************************
*     6th Call to D03PPF
************************************************************************
*
      IF (NPARAM .EQ. 3) THEN
            IF (NDEP .EQ. 1) THEN
            FDEP = A(1)
                  IF (NDIF .EQ. 2) THEN
                  DASC = A(2)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(2)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 2) THEN
            FDEP = A(2)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(1)
                  DASC = A(3)
                  END IF
```

```
            ELSE IF (NDEP .EQ. 3) THEN
            FDEP = A(3)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(2)
                  ELSE
                  FLXG = A(1)
                  DASC = A(2)
                  END IF
            END IF
      ELSE IF (NPARAM .EQ. 2) THEN
            IF (NDEP .EQ. 0) THEN
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(2)
                  ELSE
                  FLXG = A(1)
                  DASC = A(2)
                  END IF
            ELSE IF (NDEP .EQ. 1) THEN
                  FDEP = A(1)
                  IF (NDIF .EQ. 2) THEN
                  DASC = A(2)
                  ELSE
                  FLXG = A(2)
                  END IF
            ELSE IF (NDEP .EQ. 2) THEN
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  ELSE
                  FLXG = A(1)
                  END IF
                  FDEP = A(2)
            END IF
      ELSE IF (NPARAM .EQ. 1) THEN
            IF (NDEP .EQ. 0) THEN
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  ELSE
                  FLXG = A(1)
                  END IF
            ELSE
                  FDEP = A(1)
            END IF
      END IF
!     write(lsim,*) ' FLXG, 6th pass =', FLXG/FLXGOLD
!     write(lsim,*) ' DASC, 6th pass =', DASC/DASCOLD
*
      IND  = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
```

```
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
            DO 2060 MM = 1,NEXDP
            YFIT(MM) = 0.0D0
2060  CONTINUE
*
      DO 206 IT6 = 1, 3000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF

      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $            ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $            REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $            CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)

*     Leaving D03PPF
*
*     Separating the Temperature and Concentration Solutions
      DO 2061 KL = 1,NPDE
            DO 2062 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2062        CONTINUE
2061  CONTINUE
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!!    FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/BZSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
```

```
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 2063 NN = 1, NEXDP
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
2063        CONTINUE
*
!     write(lsim,*) 'time =', thrs, 'yfit(11), 6th pass =', yfit(11), 'it6 =', it6
*
      TOLD = T
      IF (THRS .GT. 2400.0D0) THEN
      GO TO 2069
      ELSE
      END IF
      PPENOLD = PPEN
*
*
206   CONTINUE
************************************************************************
*     END 6th Call to D03PPF
************************************************************************
*
2069  CHISQ3 = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!     WRITE(lsim,*) 'end 6th NAG time looping'
!     write(lsim,*) 'chisq3, 6th pass', chisq3
      IF (CHISQ3 .LT. CHISQ2) THEN
      CHISQ1 = CHISQ2
      CHISQ2 = CHISQ3
      GO TO 91
      ELSE
      END IF
*
*     Find Minimum of Parabola Defined by Last Three Points
*
      DELTA = 1.0D0/(1.0D0 + (CHISQ1-CHISQ2)/(CHISQ3-CHISQ2)) + 0.5D0
      DO 104 J = 1, NPARAM
      A(J) = A(J) - DELTA*DA(J)
      SIGMAA(J) = DELTAA(J) + SQRT(DBLE(NFREE)*ALPHA(J,J))
104   CONTINUE
*
************************************************************************
*     7th Call to D03PPF
```

```
************************************************************************
*
      IF (NPARAM .EQ. 3) THEN
            IF (NDEP .EQ. 1) THEN
            FDEP = A(1)
                  IF (NDIF .EQ. 2) THEN
                  DASC = A(2)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(2)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 2) THEN
            FDEP = A(2)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(1)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 3) THEN
            FDEP = A(3)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(2)
                  ELSE
                  FLXG = A(1)
                  DASC = A(2)
                  END IF
            END IF
      ELSE IF (NPARAM .EQ. 2) THEN
            IF (NDEP .EQ. 0) THEN
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(2)
                  ELSE
                  FLXG = A(1)
                  DASC = A(2)
                  END IF
            ELSE IF (NDEP .EQ. 1) THEN
                  FDEP = A(1)
                  IF (NDIF .EQ. 2) THEN
                  DASC = A(2)
                  ELSE
                  FLXG = A(2)
                  END IF
            ELSE IF (NDEP .EQ. 2) THEN
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  ELSE
                  FLXG = A(1)
                  END IF
```

```
                FDEP = A(2)
            END IF
        ELSE IF (NPARAM .EQ. 1) THEN
            IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
            ELSE
                FDEP = A(1)
            END IF
        END IF
!       write(lsim,*) ' DASC, 7th pass =', DASC/DASCOLD
*
        IND = 0
        T = 0.0D0
        TOUT = 0.0D0
        TOLD = 0.0D0
        CFLX = 0.0D0
        CEVP = 0.0D0
        CPEN = 0.0D0
        PPENOLD = 0.0D0
*
            DO 2070 MM = 1,NEXDP
            YFIT(MM) = 0.0D0
2070    CONTINUE
*
        DO 207 IT7 = 1, 3000
        TOUT = T + DELT
*
        IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*       Entering D03PPF

        CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $              ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $              REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $              CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)

*       Leaving D03PPF
*
*       Separating the Temperature and Concentration Solutions
        DO 2071 KL = 1,NPDE
            DO 2072 KM = 1,NPTS
                SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2072        CONTINUE
2071    CONTINUE
*
        VHTI = U(NEQN)/VHTH
        DP = SOL(NPTS,2) - SOL(NPTS-1,2)
        DX = VTTH/DBLE(NPTS-1)
        FCAL = RHVT*DAVT*(DP/DX)
```

```
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/BZSCIN
      TMVH = 0.0D0
      END IF
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 2073 NN = 1, NEXDP
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
2073  CONTINUE
*
!     write(lsim,*) 'yfit(11), 7th pass =', yfit(11), 'it7 =', it7
*
      TOLD = T
      IF (THRS .GT. 2400.0D0) THEN
      GO TO 2079
      ELSE
      END IF
      PPENOLD = PPEN
*
*
207         CONTINUE
*************************************************************************
*     END 7th Call to D03PPF
*************************************************************************
*
```

```
2079  CHISQR = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
!     WRITE(lsim,*) 'end 7th NAG time looping'
      write(lsim,*) 'chisqr, 7th pass', chisqr
      write(lsim,*) 'if negative then end optimization', chisqr-chisq2
      IF (CHISQ2 .LT. CHISQR) THEN
            DO 113 J = 1, NPARAM
            A(J) = A(J) + (DELTA-1.0D0)*DA(J)
113         CONTINUE
      ELSE
!     GO TO 999 ! Exit Optimization Process
      END IF
*
      EPSCHI = ABS(100.0D0*(CHISQR-CHIOLD)/CHISQR)
      write(lsim,*) 'EPSCHI =', EPSCHI
      IF (EPSCHI .LE. CHITOL) THEN
      GO TO 999
      ELSE
      END IF
      CHIOLD = CHISQR
998   CONTINUE
*
************************************************************************
*     Final Call to D03PPF
************************************************************************
*
999   IF (NPARAM .EQ. 3) THEN
            IF (NDEP .EQ. 1) THEN
            FDEP = A(1)
                  IF (NDIF .EQ. 2) THEN
                  DASC = A(2)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(2)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 2) THEN
            FDEP = A(2)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(3)
                  ELSE
                  FLXG = A(1)
                  DASC = A(3)
                  END IF
            ELSE IF (NDEP .EQ. 3) THEN
            FDEP = A(3)
                  IF (NDIF .EQ. 1) THEN
                  DASC = A(1)
                  FLXG = A(2)
                  ELSE
                  FLXG = A(1)
                  DASC = A(2)
                  END IF
```

```
            END IF
        ELSE IF (NPARAM .EQ. 2) THEN
            IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                FLXG = A(2)
                ELSE
                FLXG = A(1)
                DASC = A(2)
                END IF
            ELSE IF (NDEP .EQ. 1) THEN
                FDEP = A(1)
                IF (NDIF .EQ. 2) THEN
                DASC = A(2)
                ELSE
                FLXG = A(2)
                END IF
            ELSE IF (NDEP .EQ. 2) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
                FDEP = A(2)
            END IF
        ELSE IF (NPARAM .EQ. 1) THEN
            IF (NDEP .EQ. 0) THEN
                IF (NDIF .EQ. 1) THEN
                DASC = A(1)
                ELSE
                FLXG = A(1)
                END IF
            ELSE
                FDEP = A(1)
            END IF
        END IF
*
        WRITE(lsim,*) '*****************************************************'
        write (lsim,*) 'change in FLXG =', FLXG*FCOR/FLXGOLD
        write (lsim,*) 'change in dasc =', DASC*DCOR/DASCOLD
        write (lsim,*) 'change in fdep =', FDEP/FDEPOLD
        WRITE(lsim,*) '*****************************************************'
        WRITE (lsim,*) 'FINAL FLXG =', FLXG*1.0D6
        WRITE (lsim,*) 'FINAL DASC =', DASC
        WRITE (lsim,*) 'FINAL FDEP =', FDEP
*
        DO 101 I = 1,NPARAM
        IF (A(I) .LE. 0.0D0) THEN
        GO TO 2089
        ELSE
        END IF
101     CONTINUE
*
```

```
*
      IND = 0
      T = 0.0D0
      TOUT = 0.0D0
      TOLD = 0.0D0
      CFLX = 0.0D0
      CEVP = 0.0D0
      CPEN = 0.0D0
      PPENOLD = 0.0D0
*
      DO 2080 IT = 1, 3000
      TOUT = T + DELT
*
      IFAIL= 0 !use IFAIL = -1 if complicated errors crop up, default = 0 !
*
*     Entering D03PPF

      CALL D03PPF(NPDE,M,T,TOUT,PDEDEF,BNDARY,UVINIT,U,NPTS,X,NCODE,
     $                ODEDEF,NXI,XI,NEQN,RTOL,ATOL,ITOL,NORM,LAOPT,ALGOPT,
     $                REMESH,NXFIX,XFIX,NRMESH,DXMESH,TRMESH,IPMINF,XRATIO,
     $                CONST,D03PCL,W,NW,IW,NIW,ITASK,ITRACE,IND,IFAIL)

*     Leaving D03PPF
*
      DO 2081 KL = 1,NPDE
            DO 2082 KM = 1,NPTS
                  SOL(KM,KL) = U(NPDE*(KM-1)+KL)
2082        CONTINUE
2081  CONTINUE
*
      VHTI = U(NEQN)/VHTH
      DP = SOL(NPTS,2) - SOL(NPTS-1,2)
      DX = VTTH/DBLE(NPTS-1)
      FCAL = RHVT*DAVT*(DP/DX)
      FPEN = FCAL
!     FPEN = FBTM ! IF FPEN is obtained from BNDARY
*
*
      THRS = T/3600.0D0
      TDIF = T - TOLD ! Trapezoidal (b-a)
*
      IF (VHTI .GE. 0.0D0) THEN
      FEVP = FLXG
      TMVH = U(NEQN)*RHBZ
      WRITE(llvt,1004) THRS, VHTI
*
      ELSE
*
      FEVP = FLXG*SOL(NPTS,1)/BZSCIN
      TMVH = 0.0D0
      END IF
      CALL MASSBL (NPTS, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
      CFLX = CFLX + FTTL*TDIF ! Cumulative Flux
```

```
      CHECK = (TMAS + CFLX)/DOSE ! Should be ~ 1
      WRITE(lmbc,1004) THRS, CHECK
*
!     Amounts Evaporated and Penetrated
      CEVP = CEVP + (ABS(FEVP)*TDIF)
      CPEN = CPEN + (ABS(FPEN)*TDIF)
      PEVP = CEVP*100.0D0/DOSE
      PPEN = CPEN*100.0D0/DOSE
      PTTL = PEVP + PPEN
*
            DO 2083 NN = 1, NEXDP
      IF (T .LT. XDATA(NN)) THEN
      ELSE IF (T .EQ. XDATA(NN)) THEN
      YFIT(NN) = PPEN
      ELSE
            IF (TOLD .LT. XDATA(NN)) THEN
            SLP = (PPEN - PPENOLD)/TDIF
            YFIT(NN) = SLP*(XDATA(NN)-TOLD) + PPENOLD
            ELSE IF (TOLD .EQ. XDATA(NN)) THEN
            YFIT(NN) = PPENOLD
            END IF
      END IF
2083  CONTINUE
*
      TOLD = T
      IF (THRS .GT. 240.0D0) THEN
      CHISQR = FCHISQ (YDATA, SIGMAY, NEXDP, NFREE, MODE, YFIT)
      WRITE(lsim,*) '*****************************************************'
      write (lsim,*) 'NFREE =', NFREE
      WRITE (lsim,*) 'FINAL CHISQ =', CHISQR*1.0D5
      WRITE (lsim,*) 'FINAL SSR =', CHISQR*DBLE(NFREE)
      GO TO 2089
      ELSE
      END IF
      PPENOLD = PPEN
*
*
*     Writing the Solutions!
      WRITE(lAsc,1001) THRS
      WRITE(lAsc,1002) (SOL(J,1), J=1, NPTS)
      WRITE(lAve,1001) THRS
      WRITE(lAve,1002) (SOL(J,2), J=1, NPTS)
      UFLX = 1.0D6*3600.0D0
      WRITE(lflx,1004) THRS, ABS(FEVP)*UFLX, ABS(FBTM)*UFLX, ABS(FCAL)*UFLX
      WRITE(lcmf,1004) THRS, PEVP, PPEN, PTTL
*
2080  CONTINUE
*************************************************************************
*     END Final Call to D03PPF
*************************************************************************

*
*
```

```
1001  FORMAT('Time (s) ', F24.12)
1002  FORMAT(5(F18.12,1X))
1003  FORMAT(F36.12, 2X, F18.12, 2X, F18.12)
1004  FORMAT(F36.12, 2X, 6(F36.12))
*
2089  CLOSE(lAsc, STATUS='KEEP')
            CLOSE(lAve, STATUS='KEEP')
            CLOSE(llvt, STATUS='KEEP')
            CLOSE(lsim, STATUS='KEEP')
            CLOSE(lflx, STATUS='KEEP')
            CLOSE(lmbc, STATUS='KEEP')
            CLOSE(lcmf, STATUS='KEEP')
!           CLOSE(lext, STATUS='KEEP')
*
            STOP
            END
*
*
```

```
************************************************************************
*      SUBROUTINE PDEDEF – This Subroutine defines the system PDEs
*********************************************
************************************************************************
*
      SUBROUTINE PDEDEF (NPDE,T,X,U,UX,NCODE,V,VDOT,P,Q,R,IRES)
*
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      IMPLICIT INTEGER (I-N)
      DOUBLE PRECISION U(NPDE), UX(NPDE),P(NPDE,NPDE),
     $                 Q(NPDE), R(NPDE), V(NCODE), VDOT(NCODE)
*
      COMMON /VHPAR/ RHBZ, WMBZ
      COMMON /SCPAR/ RHSC, SCTH, DASC, FDEP, CORR
      COMMON /VEPAR/ RHVT, VTTH, DAVT
*
            DO 201 I = 1, NPDE
                  DO 202 J = 1, NPDE
                  P(I,J) = 0.0D0
202               CONTINUE
201         CONTINUE
*
C           Mass Transport Equation for Component A (Stratum Corneum)
      BZMF = U(1)
      CALL DENSTY (BZMF, RHBZ, RHSC, RHAV)
*
      SCCONV = 1.0D0-U(1)
      VECONV = 1.0D0-U(2)
*
      FASC   =  -RHAV*DASC*UX(1)/(SCTH*SCCONV)
      P(1,1) =  -RHAV*SCTH
      Q(1)   =  0.0D0
      R(1)   =  FASC
*
C           Mass Transport Equation for Component A (Viable Epidermis)
      FAVE   =  RHVT*DAVT*UX(2)/(VTTH*VECONV)
      P(2,2) =  RHVT*VTTH
      Q(2)   =  0.0D0
      R(2)   =  FAVE
*
      RETURN
      END
*
*
```

```
************************************************************************
*       SUBROUTINE UVINIT – This subroutine defines the initial conditions
************************************************
************************************************************************
*
      SUBROUTINE UVINIT(NPDE,NPTS,NXI,X,XI,U,NCODE,V)
      IMPLICIT DOUBLE PRECISION (A-H , O-Z)
      IMPLICIT INTEGER (I-N)
      DOUBLE PRECISION X(NPTS), U(NPDE,NPTS), V(NCODE)
*
      COMMON /EXPMT/ TEMP, DOSE, VHTH, BZSCIN
      COMMON /SCPAR/ RHSC, SCTH, DASC, FDEP, CORR
*
      DO 10 I = 1,NPTS
      U(1,I) = 0.0D0
      U(2,I) = 0.0D0
10    CONTINUE
*
      LDEP = FDEP*100.0D0
      NSAT = ((NPTS-1)*LDEP/100) + 1
      DO 11 I = NPTS, NPTS-NSAT, -1
      U(1,I) = BZSCIN
11    CONTINUE
*
      V(1) = VHTH
*
      RETURN
      END
*
*
```

```
************************************************************************
*       SUBROUTINE ODEDEF – This subroutine defines the ODEs
************************************************
************************************************************************
*
      SUBROUTINE ODEDEF (NPDE,T,NCODE,V,VDOT,NXI,XI,UCP,UCPX,RCP,
     $            UCPT,UCPTX,F,IRES)
*
      IMPLICIT DOUBLE PRECISION (A-H , O-Z)
      IMPLICIT INTEGER (I-N)
      DOUBLE PRECISION XI(NXI),UCP(NPDE,NXI),UCPX(NPDE,NXI),RCP(NPDE,NXI),
     $                 UCPT(NPDE,NXI),UCPTX(NPDE,NXI),F(NCODE),V(NCODE),
     $                 VDOT(NCODE)
*
      COMMON /EXPMT/ TEMP, DOSE, VHTH, BZSCIN
      COMMON /VHPAR/ RHBZ, WMBZ
      COMMON /INTER/ FLXG, FLSCVH, FLVESC, FBTM, SCVHPC, VTSCPC, PTOR
*
      IF(IRES.EQ.-1) THEN
      F(1) = - VDOT(1)
      ELSE
      F(1) = - VDOT(1) + (FLSCVH - FLXG)/RHBZ
      END IF
*
      RETURN
      END
*
*
```

```
************************************************************************
*       SUBROUTINE BNDARY – This subroutine defines the boundary conditions
*********************************************
************************************************************************
*
      SUBROUTINE BNDARY(NPDE,T,U,UX,NCODE,V,VDOT,IBND,BETA,GAMMA,IRES)
*
      IMPLICIT DOUBLE PRECISION (A-H , O-Z)
      IMPLICIT INTEGER (I-N)
      DOUBLE PRECISION BETA(NPDE),GAMMA(NPDE),U(NPDE),UX(NPDE),V(NCODE)
*
      COMMON /EXPMT/ TEMP, DOSE, VHTH, BZSCIN
      COMMON /VHPAR/ RHBZ, WMBZ
      COMMON /SCPAR/ RHSC, SCTH, DASC, FDEP, CORR
      COMMON /VEPAR/ RHVT, VTTH, DAVT
      COMMON /INTER/ FLXG, FLSCVH, FLVESC, FBTM, SCVHPC, VTSCPC, PTOR
*
      SCCONV = 1.0D0-U(1)
      VECONV = 1.0D0-U(2)
*
      IF (IBND .EQ. 0) THEN
*
      BZMF = U(1)
      CALL DENSTY (BZMF, RHBZ, RHSC, RHAV)
      FAVE = RHVT*DAVT*UX(2)/(VTTH*VECONV)
*
C     Mass Transport/ Component A in SC/ SC-VT Interface
      BETA(1)  = 1.0D0
      GAMMA(1)  = FAVE
*
C     Mass Transport/ Component A in VT/ SC-VT Interface
      BETA(2)  = 0.0D0
      GAMMA(2)  = U(2)*RHVT - U(1)*RHAV*VTSCPC
*
      FLVESC = FAVE
*
*
      ELSE
*
*
C     Mass Transport/ Component A in SC/ VH-SC Interface
*
      BZMF = U(1)
      CALL DENSTY (BZMF, RHBZ, RHSC, RHAV)
*
      IF (V(1) .GE. 0.0D0) THEN
      BETA(1)  = 0.0D0
!     GAMMA(1)  = U(1)*RHSC - SCVHPC*RHBZ
      GAMMA(1)  = U(1) - BZSCIN
      ELSE
      BETA(1)  = 1.0D0
!     GAMMA(1)  = FLXG*U(1)
      GAMMA(1)  = FLXG*BZMF/BZSCIN
```

```
      END IF
*
C     Mass Transport/ Component A in VT/ VT-Bottom Interface
      BETA(2)  = 0.0D0
      GAMMA(2)  = U(2)
*
      FLSCVH = -RHAV*DASC*UX(1)/(SCTH*SCCONV)
      FBTM = RHVT*DAVT*UX(2)/(VTTH*VECONV)
*
      END IF
*
      RETURN
      END
*
*
```

```
*************************************************************************
*      SUBROUTINE MASSBL – This subroutine checks for mass balance errors
*********************************************
*************************************************************************
*
      SUBROUTINE MASSBL (N, SOL, TMVH, FEVP, FPEN, TMAS, FTTL)
      IMPLICIT DOUBLE PRECISION (A-H , O-Z)
      IMPLICIT INTEGER (I-N)
      DOUBLE PRECISION SOL(N,N)
*
      COMMON /VHPAR/ RHBZ, WMBZ
      COMMON /SCPAR/ RHSC, SCTH, DASC, FDEP, CORR
      COMMON /VEPAR/ RHVT, VTTH, DAVT
*
*     Calculating Total Amount in Individual Layers
*
      BZMF = SOL(1,1)
      CALL DENSTY (BZMF, RHBZ, RHSC, RHAV)
      CASC = SOL(1,1)*RHAV
      DO 522 K = 2, N-1
      BZMF = SOL(K,1)
      CALL DENSTY (BZMF, RHBZ, RHSC, RHAV)
      CASC = CASC + 2.0D0*RHAV*SOL(K,1) ! 2{F(2)+F(3)+...+F(N-1)}
522   CONTINUE
      BZMF = SOL(N,1)
      CALL DENSTY (BZMF, RHBZ, RHSC, RHAV)
      CASC = CASC + SOL(N,1)*RHAV ! F(N)
      TMSC = CASC*SCTH/(2.0D0*DBLE(N-1)) ! h = L/100; L is DST
*
      CAVE = SOL(1,2)*RHVT
      DO 523 K = 2, N-1
      CAVE = CAVE + 2.0D0*SOL(K,2)*RHVT! 2{F(2)+F(3)+...+F(N-1)}
523   CONTINUE
      CAVE = CAVE + (SOL(N,2))*RHVT ! F(N)
      TMVE = CAVE*VTTH/(2.0D0*DBLE(N-1)) ! h = L/100; L is DST
*
      TMAS = TMVH + TMSC + TMVE
*
      FTTL = ABS(FEVP) + ABS(FPEN)
*
      RETURN
      END
*
*
```

```
************************************************************************
*       SUBROUTINE DENSTY – This subroutine calculates a variable density using the
composition of a mixture using the specific volumes of the components
************************************************
************************************************************************
*
      SUBROUTINE DENSTY (PNMF, RHPN, RHLY, RHAV)
      IMPLICIT DOUBLE PRECISION (A-H , O-Z)
      IMPLICIT INTEGER (I-N)
*
      RHAV = 1.0D0/(PNMF/RHPN + (1.0D0-PNMF)/RHLY)
*
      RETURN
      END
*
*
```

```
************************************************************************
*       SUBROUTINE CONSTS – Physicochemical and transport properties for Benzene
************************************************
************************************************************************
*
        SUBROUTINE CONSTS
        IMPLICIT DOUBLE PRECISION (A-H , O-Z)
        IMPLICIT INTEGER (I-N)
*
        COMMON /EXPMT/ TEMP, DOSE, VHTH, BZSCIN
        COMMON /VHPAR/ RHBZ, WMBZ
        COMMON /SCPAR/ RHSC, SCTH, DASC, FDEP, CORR
        COMMON /VEPAR/ RHVT, VTTH, DAVT
        COMMON /INTER/ FLXG, FLSCVH, FLVESC, FBTM, SCVHPC, VTSCPC, PTOR
        COMMON /CRRCT/ FCOR, DCOR
        COMMON /VALUE/ DAVE, VESCPC, DFREE, FUBZ, SCWRPC, VTWRPC
*
*       Evaporating & Absorbing Component - Benzene
        BZMOLV = 105.0D0 ! Molar Volume of Penetrant at its Normal BP
        WMBZ  = 78.114D0 ! Molecular Weight of The Penetrant
        PKABZ = 43.0D0
!       http://chemweb.unp.ac.za/chemistry/Physical_Data/pKa_values.htm
*
*-----------------------------------------------------------------
C       Vapor-Pressure
        PTOR = 129.1D0
*-----------------------------------------------------------------
*
*       Experimental Conditions
        TEMP = 305.0D0
*
!       VOL = 40.0D-3
!       AREA = 0.79D0
!       VHTH = VOL/AREA
        RHBZ = 0.8679D0 ! Chemical Properties Handbook
!       DOSE = VOL*RHBZ/AREA ! mass/area
*
!       Calculating the Gas-phase Flux
        UAIR = 0.7368D0 ! From experimental data
        VAIR = 42.0D0
        CALL PERESS (TEMP, WMBZ, UAIR, VAIR, PTOR, CMTA, FLXG)
        FCOR = 1.0D0
        FLXG = FLXG*FCOR
*
*-----------------------------------------------------------------
C       SC Parameters and Constants pertaining to the SC-VH Interface
*
        SCTH = 13.4D-4
        FDEP = 0.360D0
        HDEP = FDEP*SCTH
        BZLOGK = 2.13D0
*
        VH2O = 0.43D0
```

```
      OLIP = 0.10D0
      DLIP = 0.90D0
      OPRO = 0.90D0
      DPRO = 1.37D0
      RHSC = (1.0D0 + VH2O)/(OLIP/DLIP + OPRO/DPRO + VH2O)
*
      CALL GBKJMN (BZLOGK, WMBZ, BZMOLV, SCWRPC, DASC)
      DCOR = 1.0D0
      DCOR = 11.7029D0
      DASC = DASC*DCOR
*
      BZWSLB = 1.787D-3
      BZCSAT = SCWRPC*BZWSLB
*
      CORR = HDEP*BZCSAT/RHBZ
      SCTH = SCTH + CORR
!     VHTH = VHTH - CORR
      A = 1.0D0/RHBZ - 1.0D0/RHSC
      RHSK = RHSC*(1.0D0 - A*BZCSAT)
      BZSCIN = BZCSAT/RHSK
      SCVHPC = SCWRPC*BZWSLB/RHBZ
*
*-------------------------------------------------------------------------------------------
C     VT Parameters and Constants pertaining to the VT-SC Interface
*
      VTTH = 285.0D-4
      VTTH = 500.0D-4
*
      AQVT = 0.7D0
      RHVT = 1.0D0/(AQVT + (0.649*(1.0D0-AQVT)*0.5D0) + (1.227*(1.0D0-
AQVT)*0.5D0)) !Andersen et al.
*
!     DIND = -3.191D0 - 1.355D0*LOG10(WMBZ)
      DIND = -5.025D0 - 0.0262D0*(WMBZ**(2.0D0/3.0D0))
      DAVE = 1.0D1**DIND
      DIND = -4.15D0 - 0.655D0*LOG10(WMBZ)
      DFREE = 1.0D1**DIND
      CALL FRACTN (PKABZ, BZLOGK, FUBZ)
      FNON = 1.0D0
      BZKOCT = 1.0D1**BZLOGK
      DAVT = DFREE/(0.68D0 + 0.32D0/FUBZ + 0.001D0*FNON*BZKOCT)
*
      PCFREE = 0.60D0
      VTWRPC = PCFREE*(((0.68D0 + 0.32D0/FUBZ)/FNON) + 0.001D0*BZKOCT)
      VTSCPC = VTWRPC/SCWRPC
      VESCPC = 1.0D0/SCWRPC
*
*
      RETURN
      END
*
*
```

```
**********************************************************************
*       SUBROUTINE FRACTN – This subroutine calculates fraction unbound plasma
according to the algorithm proposed by Yamazaki and Kanaoka
**********************************************
**********************************************************************
*
      SUBROUTINE FRACTN (PKA, VLGKOCTW, FSBU)
      IMPLICIT DOUBLE PRECISION (A-H , O-Z)
      IMPLICIT INTEGER (I-N)
*
!     IF ACIDIC, THEN I = -1
!     IF NEUTRAL, THEN I = 0
!     IF BASIC, THEN I = 1
*
      IF ((PKA .GT. 10.0D0) .OR. (PKA .LT. 4.0D0)) THEN
      I = 0
!     GO TO 1
      ELSE
      END IF

      VLGDPH0 = VLGKOCTW - LOG10(1.0D0 + 10.0D0**(0.0D0-PKA))
      VLGDPH7 = VLGKOCTW - LOG10(1.0D0 + 10.0D0**(7.0D0-PKA))
      VLGDPH14 = VLGKOCTW - LOG10(1.0D0 + 10.0D0**(14.0D0-PKA))
      VLGDPH74 = VLGKOCTW - LOG10(1.0D0 + 10.0D0**(7.4D0-PKA))
*
      IF (VLGDPH0 .GT. VLGDPH14) THEN
      I = -1
      ELSE IF (VLGDPH0 .LT. VLGDPH14) THEN
      I = 1
      ELSE
      I = 0
      END IF
*
      IF ((VLGDPH0 .LT. VLGDPH7) .AND. (VLGDPH7 .GT. VLGDPH14)) THEN
      I = 2 ! Zwitterion
      ELSE
      END IF
*
      IF  ((I .EQ. 0) .OR. (I .EQ. 1)) THEN
!     write (*,*) 'flag'
      PBR = (0.5578D0*EXP(VLGDPH74) + 0.0188D0)/(0.5578D0*EXP(VLGDPH74) +
1.0188D0) ! Formula 9
      ELSE
      PBR = (0.7936D0*EXP(VLGKOCTW) + 0.2239D0)/(0.7936D0*EXP(VLGKOCTW) +
1.2239D0) ! Formula 11
      END IF
      FSBU = 1.0D0 - PBR
*
*
      RETURN
      END
*
*
```

```
**************************************************************************
*       SUBROUTINE PERESS – This subroutine calculates the evaporative mass-transfer
coefficient according to the Peress correlation
**********************************************
**************************************************************************
*
      SUBROUTINE PERESS (TEMP, WMCP, UMPS, VAIR, PTOR, CMTR, FLXG)
      IMPLICIT DOUBLE PRECISION (A-H , O-Z)
      IMPLICIT INTEGER (I-N)
*
      CMTG = 6320.0D0*(UMPS**0.78D0)/(WMCP**0.33D0) ! in cm/h
*
!     Making the airflow correction (optional)
      CMTR = CMTG*VAIR/42.0D0 ! airflow correction
*
!     Calculating the temperature independent constant function
      RGAS = 0.0821D0 ! in Latmmol-1K-1
      FNMT = CMTR*WMCP/(0.76D0*RGAS)
*
!     Calculating the actual Gas-phase Flux
      FLXG = (FNMT*PTOR/TEMP)*(1.0D-6/3600.0D0) ! converting to CGS
*
*
      RETURN
      END
*
*
```

```
************************************************************************
*       SUBROUTINE GBKJMN – This subroutine calculates SC diffusivity for a permeant
*********************************************
************************************************************************
*
      SUBROUTINE GBKJMN (VLGKOCTW, WMCP, BPMOLV, SCWRPC, DASC)
      IMPLICIT DOUBLE PRECISION (A-H , O-Z)
      IMPLICIT INTEGER (I-N)
      DOUBLE PRECISION C(5,5)
*
      MODEL = 2
*
      OCTWPC = 1.0D1**VLGKOCTW
*
*     Lipid Phase Properties
      RLPWPC = 0.43D0*(OCTWPC**0.81D0)
      HLAT = 3.0D0
      HTRN = 3.0D0
      DLIP = (1.24D-7*((1.0D2/WMCP)**2.43D0) + 2.34D-9)/HLAT
      IF (MODEL .EQ. 1) THEN
      RLGKTRNS = -0.570D0 - 0.840*(WMCP**0.333D0) - LOG10(HTRN)
      ELSE IF (MODEL .EQ. 2) THEN
      RLGKTRNS = -0.725D0 - 0.792*(WMCP**0.333D0) - LOG10(HTRN) ! MODEL 2
      END IF
      SMLKTRNS = 1.0D1**RLGKTRNS
*
*     Corneocyte Phase Properties
      IF (BPMOLV .LE. 445.2D0) THEN
      DAQ = 1.92D-4/(BPMOLV**0.6D0)
      AS = 0.145*(BPMOLV**0.6D0)
      ELSE
      DAQ = 3.78D-5/(BPMOLV**0.333D0)
      AS = 0.735*(BPMOLV**0.333D0)
      END IF
      RLMBDA = AS/35.0D0
      PHIFPR = 0.6044D0*((1.0D0+RLMBDA)**2.0D0)
      CORWPCFR = 1.0D0 - PHIFPR
      FCTR = 1.0001D0 - 2.4497D0*RLMBDA + 1.1410*(RLMBDA**2.0D0)
     $     + 0.5432D0*(RLMBDA**3.0D0)
      DCORFR = DAQ*CORWPCFR*FCTR
*
*     Dimensionless Groups
      R = SMLKTRNS/(0.141916D0*DLIP)
      SMSIGM = DLIP*RLPWPC/(DCORFR*CORWPCFR)
      IF (MODEL .EQ. 1) THEN
      PERMSCWRNDCR = 1.0D0/(0.8979D0*SMSIGM + 5.5360D5/R) + 2.9350D-4
*
      ELSE IF (MODEL .EQ. 2) THEN
      IF (R .GT. 100.0D0) THEN
      PERMSCWRNDCR = 1.0D0/(0.8979D0*SMSIGM + 5.5360D5/R)
      ELSE
*
      C(1,1) = -4.95725D0
```

```
      C(1,2) =  8.38263D-1
      C(1,3) = -2.07133D-1
      C(1,4) =  8.37613D-2
      C(1,5) = -8.61439D-3
*
      C(2,1) =  1.87015D-3
      C(2,2) = -1.05288D-4
      C(2,3) = -8.49198D-3
      C(2,4) =  6.66713D-3
      C(2,5) = -1.19412D-3
*
      C(3,1) =  7.45813D-4
      C(3,2) = -1.86240D-4
      C(3,3) = -3.60234D-3
      C(3,4) =  3.43394D-3
      C(3,5) = -7.68073D-4
*
      C(4,1) =  1.82904D-4
      C(4,2) = -8.28876D-5
      C(4,3) = -5.26007D-4
      C(4,4) =  7.13520D-4
      C(4,5) = -2.13017D-4
*
      C(5,1) =  2.83838D-5
      C(5,2) = -1.98892D-5
      C(5,3) = -6.05354D-6
      C(5,4) =  4.84676D-5
      C(5,5) = -2.15660D-5
*
      RLGPRM = 0.0D0
      DO 11 I = 1,5
            DO 12 J =1,5
            RLGPRM = RLGPRM + C(I,J)*((LOG10(SMSIGM))**(I-1))*((LOG10(R)**(J-
1)))
12          CONTINUE
11    CONTINUE
*
      PERMSCWRNDCR = 1.0D1**RLGPRM
      END IF
      END IF
*
*     Effective SC Parameters
*
      SCWRPC = 0.040D0*(OCTWPC**0.81D0) + 4.057*(OCTWPC**0.27D0) + 0.359D0
*
      DASC = PERMSCWRNDCR*DLIP*RLPWPC/SCWRPC
*
      RETURN
      END
*
*
```

```
**************************************************************************
*       SUBROUTINE VARDIF – This subroutine calculates variable diffusivity according to a
user-defined sigmoidal function
**********************************************
**************************************************************************
*
      SUBROUTINE VARDIF (CINS, DASC)
      IMPLICIT DOUBLE PRECISION (A-H , O-Z)
      IMPLICIT INTEGER (I-N)
*
      COMMON /CRRCT/ FCOR, DCORR1, DCORR2, SLPD, CTRN
      COMMON /SCPAR/ RHSC, SCTH, DWNK, FDEP, CORR
*
      DASC = DWNK*DCORR1*(1.0D0 + (DCORR2 - 1.0D0)/(1.0D0 + EXP(SLPD*(1.0D0 -
     $    CINS/CTRN))))
*
      RETURN
      END
*
*
```

```
*************************************************************************
*       FUNCTION FCHISQ – This subprogram calculates the error value chi square
************************************************
*************************************************************************
*
      FUNCTION FCHISQ (Y, SIGMAY, NEXDP, NFREE, MODE, YFIT)
      IMPLICIT DOUBLE PRECISION (A-H , O-Z)
      IMPLICIT INTEGER (I-N)
      DOUBLE PRECISION Y(NEXDP), SIGMAY(NEXDP), YFIT(NEXDP)
*
      CHISQ = 0.0D0
      IF (NFREE .LE. 0) THEN
      FCHISQ = 0.0D0
      RETURN
      ELSE ! Accumulate CHI Squared
      DO 11 I = 1, NEXDP
            IF (MODE .LT. 0) THEN
                  IF (Y(I) .LT. 0.0D0) THEN
                  WEIGHT = 1.0D0/(-Y(I))
                  ELSE IF (Y(I) .EQ. 0.0D0) THEN
                  WEIGHT = 1.0D0
                  ELSE
                  WEIGHT = 1.0D0/Y(I)
                  END IF
            ELSE IF (MODE .EQ. 0) THEN
            WEIGHT = 1.0D0
            ELSE
            WEIGHT = 1.0D0/(SIGMAY(I)**2.0D0)
            END IF
      CHISQ = CHISQ + WEIGHT*((Y(I)-YFIT(I))**2.0D0)
11    CONTINUE
      END IF
      FCHISQ = CHISQ/DBLE(NFREE)
*
      RETURN
      END
*
*
```

```
************************************************************************
*       SUBROUTINE MATINV – This subroutine inverts an invertible matrix
*********************************************
************************************************************************
*
      SUBROUTINE MATINV (ARRAY, NORDER, DET)
      IMPLICIT DOUBLE PRECISION (A-H , O-Z)
      IMPLICIT INTEGER (I-N)
      DOUBLE PRECISION ARRAY(NORDER,NORDER), IK(NORDER), JK(NORDER)
*
      DET = 1.0D0
      DO 100 K = 1, NORDER
*
*     Find Largest Element Array(I,J) in Rest of Matrix
*
      AMAX = 0.0D0

21    DO 30 I = K, NORDER
      DO 30 J = K, NORDER
            IF ((ABS(AMAX)) .LE. (ABS(ARRAY(I,J)))) THEN
            AMAX = ARRAY(I,J)
            IK(K) = I
            JK(K) = J
            ELSE
            END IF
30    CONTINUE
*
*     Interchange Rows and Columns to Put AMAX in Array(K,K)
*
      IF (AMAX .EQ. 0.0D0) THEN
      DET = 0.0D0
      RETURN
      ELSE
      I = IK(K)
*
            IF (I .LT. K) THEN
            GO TO 21
            ELSE IF (I .GT. K) THEN
            DO 50 J = 1, NORDER
            SAVE = ARRAY(K,J)
            ARRAY(K,J) = ARRAY(I,J)
50          ARRAY(I,J) = -SAVE
            ELSE
            END IF
      J = JK(K)
*
            IF (J .LT. K) THEN
            GO TO 21
            ELSE IF (J .GT. K) THEN
            DO 60 I = 1, NORDER
            SAVE = ARRAY(I,K)
            ARRAY(I,K) = ARRAY(I,J)
60          ARRAY(I,J) = -SAVE
```

```
            ELSE
            END IF
      END IF
*
*     Accumulate Elements of Inverse Matrix
*
      DO 70 I = 1, NORDER
      IF (I .NE. K) THEN
      ARRAY(I,K) = -ARRAY(I,K)/AMAX
      ELSE
      END IF
70    CONTINUE
*
      DO 80 I = 1, NORDER
      DO 80 J = 1, NORDER
                  IF (I .NE. K) THEN
                        IF (J .NE. K) THEN
                        ARRAY(I,J) = ARRAY(I,J) + ARRAY(I,K)*ARRAY(K,J)
                        ELSE
                        END IF
                  ELSE
                  END IF
80    CONTINUE
*
      DO 90 J = 1, NORDER
      IF (J .NE. K) THEN
      ARRAY(K,J) = ARRAY(K,J)/AMAX
      ELSE
      END IF
90    CONTINUE
      ARRAY(K,K) = 1.0D0/AMAX
      ITER = ITER + 1
100   DET = DET*AMAX
*
*     Restore Ordering of the Matrix
*
      DO 130 L = 1, NORDER
      K = NORDER - L + 1
      J = IK(K)
      IF (J .GT. K) THEN
      DO 110 I = 1, NORDER
      SAVE = ARRAY(I,K)
      ARRAY(I,K) = -ARRAY(I,J)
110   ARRAY(I,J) = SAVE
      ELSE
      END IF
      I = JK(K)
      IF (I .GT. K) THEN
      DO 120 J = 1, NORDER
      SAVE = ARRAY(K,J)
      ARRAY(K,J) = -ARRAY(I,J)
120   ARRAY(I,J) = SAVE
      ELSE
```

```
      END IF
130   CONTINUE
*
      RETURN
      END
*
*
```