



RETURNING MATERIALS:
Place in book drop to
remove this checkout from
your record. FINES will
be charged if book is
returned after the date
stamped below.

~~JUL 24 '87~~

M 110

**NON-EQUILIBRIUM THERMODYNAMIC
MODELING AND PARAMETER ESTIMATION OF
PHENOMENOLOGICAL COEFFICIENTS DESCRIBING COUPLED
TRANSPORT ACROSS A MEMBRANE**

By

Steven Patrick Nowlen

A THESIS

**Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of**

MASTER OF SCIENCE

Department of Mechanical Engineering

1983

ABSTRACT

NON-EQUILIBRIUM THERMODYNAMIC MODELING AND PARAMETER ESTIMATION OF PHENOMENOLOGICAL COEFFICIENTS DESCRIBING COUPLED TRANSPORT ACROSS A MEMBRANE

By

Steven Patrick Nowlen

A diffusion chamber microscope stage has been developed which subjects a small sample of cells (10 μ l) to a psuedo-step-change in extracellular concentration of permeable and/or impermeable solutes. The response of an individual cell to the induced osmotic imbalance was documented via a series of photomicrographs. These images were processed using simple image analysis techniques to yield the volume-time history of the cell of interest. These volume-time data were used to estimate the values of the permeability parameters for the cell membrane through one of three modified ordinary least squares parameter estimation methods linked to an irreversible thermodynamic model of the membrane transport process. Data were successfully processed for hamster embryos for both the single parameter osmotic shrinkage, and the three parameter binary flow shrink-swell cases of the Kedem and Katchalsky permeability model. These results are consistent with results reported previously by other investigators.

ACKNOWLEDGMENTS

I would like to thank my adviser, John McGrath, for his help and guidance as well as for his friendship. I would also like to thank Mohsen Shabana for providing much of the data presented as a part of this work, and the other members of the BTP lab group for their help.

I would also like to express my appreciation to the Department of Mechanical Engineering and to the Department of Engineering Research whose financial support made my graduate work possible.

TABLE OF CONTENTS

List of Tables	v
List of Figures	vi
Nomenclature Listing	vii
Chapter 1: Introduction	1
1.1: Motivation for Present Work	1
1.2: Statement of Objectives	4
Chapter 2: Historical Development of Permeability Models	8
2.1: Historical Overview	8
2.2: Jacobs' Model	11
2.3: The K-K Permeability Formulation	15
2.4: The K-K Resistance Formulation	25
2.5: Comparison of the K-K Resistance and Permeability Formulations	33
2.6: A Power Series Solution to the K-K Permeability Equations	42
2.7: The Papanek Model	50
2.8: Nondimensionalization of the K-K Permeability Equations	56
Chapter 3: Parameter Estimation Methods	63
3.1: Parameter Estimation Overview	63
3.2: Ordinary Least Squares and Maximum Likelihood Estimators	66
3.3: The Gauss Minimization Method	69
3.4: Box-Kanemasu Interpolation Method	73
3.5: Marquardt's Method	78
Chapter 4: Data Gathering Techniques	81
4.1: The Diffusion Chamber	81
4.2: Characterization of the Dialysis Membrane Permeability	93
4.3: Laboratory Techniques	103
4.4: Image Processing Techniques	111
Chapter 5: Data and Results	118
5.1: Osmotic Shrinkage of Unfertilized Hamster Ova	118
5.2: Binary Flow in Unfertilized Hamster Ova	131
Chapter 6: Conclusions	137
Chapter 7: Recommendations for Future Work	141

Appendices:	146
Appendix A: Subprogram Unit MARBOX	146
Appendix B: Subprogram Unit MODEL	162
Appendix C: Subprogram Unit RK4	167
Appendix D: Subprogram Units DEQMOD and CEXT	171
Appendix E: Subprogram Units TAKEPT and CIRCLE	176
Appendix F: Subprogram Unit CUEGEN	182
Appendix G: Tabulated Osmotic Shrinkage Data	186
Appendix H: Tabulated Binary Flow Data	188
 Bibliography	 190

LIST OF TABLES

TABLE 5.1.1 - Results for Shabana's cell 1.

TABLE 5.1.2 - Results letting ω_d float as parameter.

TABLE 5.1.3 - Final permeability values as determined by parameter estimation routine.

LIST OF FIGURES

- FIGURE 2.3.1 The two-chamber system.
- FIGURE 2.4.1 Membrane cross-section.
- FIGURE 2.5.1 Membrane with capillary structure.
- FIGURE 2.6.1 Shape of V^* as a function of b .
- FIGURE 4.1.1 Cross-section of the diffusion chamber.
- FIGURE 4.1.2 Exploded view of the diffusion chamber.
- FIGURE 4.1.3 Permeability of Cuprophane dialysis membrane as a function of molecular weight.
- FIGURE 4.2.1 Two chamber system used to characterize permeability of the dialysis membrane.
- FIGURE 4.2.2 Typical results of dialysis membrane permeability experiment for glycerol.
- FIGURE 4.3.1 Typical ovum photographs from diffusion chamber.
- FIGURE 4.3.2 Typical liposome photographs from diffusion chamber.
- FIGURE 4.4.1 Point groups selection sequence and integration region selection sequence.
- FIGURE 5.1.1 Measured and predicted response of cell 1.
- FIGURE 5.1.2 Measured and predicted response of cell 2.
- FIGURE 5.1.3 Measured and predicted response of cell 3.
- FIGURE 5.1.4 Measured and predicted response of cell 4.
- FIGURE 5.2.1 Measured and predicted response of ova 5 and 6.
- FIGURE 5.2.2 Measured and predicted response of ova 7 and 8.

NOMENCLATURE LISTING

A	Area; membrane surface area.
b	Non-dimensional constant in Section 2.6.
$\vec{b}$	Vector of parameter estimates in Chapter 3.
C	Local concentration (osmol/kg).
c	Concentration (osmol/kg).
D^0	Diffusion coefficient in free solution.
k	Proportionality constant.
K	Distribution coefficient.
L	Phenomenological coefficient.
L_p	Hydraulic or solvent permeability.
N	Number of moles.
P	Modified permeability coefficient.
R	Resistance phenomenological coefficient.
S	Sum of the squares function.
t	Time
T	Temperature, absolute.
v	Velocity.
$\bar{v}$	Specific volume.
V	Volume.
x	Spatial coordinate.
X	Arbitrary driving potential.
Y	Measured data values.

Greek Symbols

α	Non-dimensional constant in Section 2.6.
$\vec{\beta}$	Vector of parameter values.
*	Model estimates to function values.
μ	Chemical potential.
π	Osmotic pressure.
ξ	Volume fraction.
σ	Reflection coefficient.
σ_i	Variance of data point (i) in Chapter 3.
Ω	Entropy dissipation function.
ω	Solute permeability.
ω_d	Dialysis membrane permeability.
τ	Non-dimensional time.

CHAPTER 1

Introduction

1.1 Motivation for Present Work

The work described in this thesis was performed at the Bio-Engineering Transport Processes (BTP) laboratory of Michigan State University. The problem addressed was to adapt and develop the experimental and analytical tools needed to perform experiments to determine the passive transport properties associated with the membrane of an individual cell. This work is part of an ongoing study of the effects of cryopreservation procedures on the survival of different cell types.

The transport properties of a cell will have a direct effect on the response of that cell to a freezing procedure. As water freezes the solutes in the water are excluded from the crystal structure of the ice. Thus as an ice front moves towards a cell the solutes are concentrated into the remaining liquid water. This rise in solute concentration will cause the cell to respond, in the form of an induced volume change, in an attempt to regain an equilibrium state with respect to the chemical potential inside and outside the cell. The dynamic response for such a situation will be governed by the permeability characteristics of the cell.

During the freezing process two major mechanisms are thought destructive, causing life-threatening damage to a cell, [22]. If the rate of freezing is "slow" then the cell is subjected to high concentration extracellular solutions for a relatively long period of time. The cell will generally respond by expelling water from inside the cell thereby decreasing the cell volume. In a sense the cell is attempting to increase the intracellular concentration in order to reestablish equilibrium. At slow freezing rates so much water is lost that the cell can suffer damage due to the high intracellular and/or extracellular concentration of solutes.

On the other hand, if the freezing rate is "fast" then the cell will contain a large fraction of the initial amount of internal water after extracellular ice has formed. This situation results in a high probability that internal ice will form, [23].

From this argument it is plain that the rate at which a cell transports water and other permeable solutes will, to a great extent, determine the freezing rate at which the cell is most likely to survive. It would be desirable to be able to predict the optimum freezing rates before attempting a freezing process. This requires that one know the transport properties of the cells of interest. It would also require that the properties be determined as a function of temperature as well.

The techniques described in this thesis make it possible to determine the passive transport properties of an individual cell.

This is a significant advance over previously published methods (such as the stop-flow methods) which only give average data based on bulk samples of the cells, [2,8,12]. Being able to observe individual cells as they undergo a non-equilibrium passive transport process makes it possible to determine information on the distribution of individual parameters within the population. It also makes it possible to work with cells for which a large bulk sample is not readily available.

The diffusion chamber microscope stage described in Chapter 4, Section 1 of the present work, which is used to gather the experimental data on the cells, can easily be modified to include a heat exchanger system. This will allow one to study the effects of temperature on the passive transport properties as well. This feature was not yet installed at the time this thesis was completed so that all of the data presented in the present work was gathered at room temperature.

1.2 Statement of Objectives

The overall objective of this work was to adapt existing and develop new experimental and analytical tools required for the determination of cell membrane permeability coefficients. It is intended that this thesis will serve as a starting point for future investigators working on the problem of passive membrane transport.

The first step towards realization of this goal was to conduct a literature survey to define the state of the art with respect to the most popular and widely used of the passive transport models. The presentation of these models is not intended to be all-encompassing or fully detailed. Rather they are meant to be used as an introduction to each of the models presented and to the field of passive membrane transport in general. Once the reader has gained a basic understanding of the passive transport process and the approach taken by various investigators, it should be possible to work with the original presentations of the models reviewed in this work and others not included here with the fundamentals in hand.

The most widely used of the passive transport models, developed by Kedem and Katchalsky in 1958, has been coded into FORTRAN for computer implementation. This subprogram is described in detail and listed in Appendix C of the present work. This subprogram is intended not only as a usable routine but also as an example of the interfacing required between the modeling subprogram and the other subprogram units described in this thesis. From this example the user should be

able to code and implement other models of interest for use with the other subprograms developed as a part of the present work.

The actual determination of the values of the permeability parameters contained in the model of interest is accomplished through a parameter estimation routine. Several of the available parameter estimation routines are presented in chapter 3 of this thesis. Here again the presentation is intended to serve only as an introduction to the science of parameter estimation. By reading this chapter the user should be able to gain sufficient understanding of the methods of parameter estimation to be able to use the subprograms developed as a part of the present work, and to understand some of the pitfalls involved. In parameter estimation the pitfalls are many and the corrective actions required are often learned only through experience. The programs presented in this work have been extensively documented and thoroughly tested and are to the best knowledge of the author in good working order.

The parameter estimation subprogram MARBOX is presented in Appendix A of this thesis. This subprogram is actually three parameter estimation routines in one. It has the capability to run as the Ordinary Least Squares (OLS) method (see chapter 3 sections 2 and 3), or the Box-Kanemasu method (see chapter 3 section 4), or as Marquardt's method (see chapter 3 section 5). It also enables the user to specify upper and lower bounds on the values of each of the floating parameters.

The user is encouraged to make his own copies of this routine and to modify it to suit his own needs. This will be particularly appropriate in the adaptation of new models and in tailoring the input and output formats. Each subroutine is written in a structured form in order to facilitate understanding. Emphasis has been placed on documenting each routine with liberal use of comment statements and variable definition blocks. The user is encouraged to follow suit.

The data reported in this thesis was collected using a simple diffusion chamber microscope stage developed by Ligon and documented in an unpublished work, [19]. This chamber is described in Chapter 4, Section 1 of the present work (see figures 4.1.1 through 4.1.3). This chamber makes it possible to subject an isolated sample of cells in suspension to a pseudo-step change in extracellular concentration. The cells remain stationary and are not sheared during this process. The response of an individual cell to this treatment can be observed directly, and videotaped or photographed. The resulting photo images can be processed to yield cell volume as a function of time. This processing can be done either by hand or through the use of computer image analysis techniques. The computer is able to define and enhance the boundary of the cell in the photomicrograph.

This volume-time history, along with the experimental conditions and initial estimates of the parameter values, form the input to the parameter estimation routine. The parameter estimation routine then calculates the statistically-determined values of the permeability parameters which result in the best fit between the experimental data

and the values predicted by the model.

The overall routine has been tested using several sets of data for two cases of the Kedem and Katchalsky permeability model. The first case was that of osmotic shrinkage in which no permeable solute is present. Osmotic shrinkage data reported by Shabana, [20], for unfertilized hamster ova was processed using the parameter estimation routine linked to the Kedem and Katchalsky permeability model for this case. The results are consistent with those generated by Shabana using a closed-form approximate solution to the Kedem and Katchalsky equations. Data has also been generated on the response of unfertilized hamster ova in a binary flow situation, that is, one in which there is a permeable solute present. Processing of these data using the parameter estimation routine also yielded consistent results.

The data on the response of the ova in the binary flow situation was generated from photomicrographs taken with 35mm black and white film. These photos were processed using simple computer image analysis techniques to yield the radius of the ovum as a function of time.

Taken together these techniques form a powerful and versatile tool for the determination of the passive transport properties associated with the membrane of an individual cell.

CHAPTER 2

Historical Development of Permeability Models

2.1 Historical Overview

Characterizing the flow of materials across the membrane of a cell has long been a problem of concern to investigators. Early models, such as Jacob's model, 1952, [1], attempted to describe the passive transport process for membranes with expressions similar to Fick's Law describing free diffusion. In this model the flowrate of a given species was assumed to be linearly dependent on the spatial gradient in concentration for that species. While this model adequately described the transport process for some cases it was found that it did not hold true in general.

The groundwork for the models generally used today was laid by Onsager in 1931, [4]. Onsager extended Lord Rayleigh's, [24], work to include thermodynamic flows and forces. Rayleigh originally expressed a linear dependence between all mechanical flows in a system and all the mechanical forces acting on the system. Thus Onsager proposed what are referred to as the phenomenological equations to govern thermodynamic systems which are not too far removed from equilibrium. He also derived certain restrictions on the resulting coefficients based on statistical considerations.

In 1958 Kedem and Katchalsky, [3], used an Onsager set of phenomenological equations and applied the principles of irreversible thermodynamics in order to derive a model for the passive transport process in a membrane. The resulting set of equations resolved the inadequacy of previous models and has become the classical model for membrane transport. This set of equations is generally referred to as the K-K permeability formulation.

The K-K model was expanded somewhat by Papanek in 1978, [12]. Papanek was able to derive a more general set of equations of the same form as the K-K equations but without assuming ideal, dilute solutions. The value of this aspect of Papanek's work remains to be proven as the increase in generality comes at the expense of considerable computational effort and little comparative data between the simpler K-K model and the Papanek model is available. The Papanek model will not be used in the present investigation as the consideration of non-ideal solutions requires the determination of certain empirical relationships for the solutes of interest and this is beyond the scope of the present work. Papanek's model is presented in anticipation of future work to be conducted in passive membrane transport.

Other investigators have derived direct closed-form functional volume-time solutions for certain special cases of the K-K equations, [13,14]. Johnson and Wilson, [8], derived an approximate closed-form power series solution for the binary K-K equations. Direct solutions of this type are generally easier to evaluate as they do not involve the evaluation of a set of differential equations over time as in the

full set of Kedem and Katchalsky equations. While these closed form solutions may involve simplifying assumptions and hence may not be as accurate as the full set of Kedem and Katchalsky equations it may be possible to use the simpler solutions to provide beginning estimates to the values of the permeability coefficients at a relatively small computational effort. This hypothesis has not been explored in the present work but again the methods are presented in anticipation of future work.

2.2 Jacobs' Model

One of the earliest models for the transport of materials across a membrane was proposed by Jacobs in 1952, [1]. He considered the problem of two regions in thermal and mechanical equilibrium separated by a semi-permeable membrane with non-electrolyte solutions of differing concentrations on each side of the membrane.

The flow of solute through the membrane was described by an equation analogous to Fick's Law:

$$dN_s^i/dt = k_s A (c_s^o - c_s^i) \quad (2.2.1)$$

where the superscript (o) represents one region and (i) the other; the subscript (s) implies solute; N represents the number of moles; A the membrane area; c_s the concentration of solute (in moles/liter); and k_s is the proportionality constant or solute permeability coefficient with units of (cm/sec).

Similarly the volume flow, which Jacobs related to water flow, is assumed to be proportional to the transmembrane difference in chemical potential of the solution. Assuming no hydrostatic pressure difference exists across the membrane, the chemical potential difference is equivalent to the osmotic pressure difference, $(\pi^i - \pi^o)^1$. Further assuming ideal solution behaviour in both regions one can use the relation²:

1 Katchalsky and Curran, [9], eq(10-8), pg.118 .

2 Katchalsky and Curran, [9], eq(5-55), pg.56 .

$$\pi = RTc \quad (2.2.2)$$

where c is the osmotic concentration, which is the sum of the concentrations of all solutes. The osmotic concentration can be expressed as the sum of the concentration of the permeable solute and the total impermeable solute concentration:

$$c = c_s + c_m \quad (2.2.3)$$

The subscript (m) implies the sum of all impermeable solutes and the subscript (s) implies the permeable solute. Thus the equation describing volume flow can be written:

$$dV^i/dt = k_w A (c^i - c^o) \quad (2.2.4)$$

where V^i is the internal cell volume, and k_w is the permeability coefficient for water (the factor RT has been absorbed into k_w giving it typical units of $\text{cm}^4/\text{mole-sec}$).

For a system of only one permeable solute one can write:

$$dV^i/dt = k_w A \left[[(N_m^i + N_s^i)/V^i] - (c_s^o + c_m^o) \right] \quad (2.2.5)$$

This is the form of the equation generally referred to as Jacobs' equation.

Many investigators have shown this model to be inadequate to des-

cribe the general passive transport process. Among these are Zeuthen and Prescott, [2]. In their investigation Zeuthen and Prescott subjected frog eggs (otherwise in equilibrium with the solutes present) to high external concentrations of heavy water, D_2O . The heavy water was shown to act like any other solute as its penetration into the eggs followed the response predicted by (2.2.1) exactly. However, while the solute penetrated the cell as expected, it was observed that the volume of the cell remained constant so that $(dV/dt = 0)$. From (2.2.3) we see that:

$$dV^i/dt = 0 = k_w A(c^i - c^o) \quad (2.2.6)$$

which implies that $c^i = c^o$. Using equation (2.2.4) then:

$$c_s^i + c_m^i = c_s^o + c_m^o \quad (2.2.7)$$

As stated above the eggs were brought to equilibrium with all other solutes present prior to immersion in the heavy water solution by first immersing them in a similar solution with normal water replacing the heavy water so:

$$c_m^o = c_m^i \quad (2.2.8)$$

This leads to the conclusion that:

$$c_s^o = c_s^i \quad (2.2.9)$$

and this is clearly a contradiction of the known experimental conditions. This then proved that (2.2.4) and (2.2.5) are incomplete descriptions of the volume transport process.

Zeuthen and Prescott went on to show that for cells not in equilibrium with the non-penetrating solutes the penetration of heavy water was not adequately described by (2.2.1). Penetration was found to be more rapid in solutions where the extracellular concentration of non-permeating solutes was lower than the internal concentrations and slower in the reverse case. This demonstrates the need for some form of coupling between the volume and solute flows which is not accounted for in the Jacobs model.

2.3 The K-K Permeability Formulation

In 1958 Kedem and Katchalsky put forth what has come to be a classic model for membrane permeation based on the principles of irreversible thermodynamics, [3]. This set of equations is known as the K-K formulation and is still used extensively in various forms.

The K-K model resolved the inadequacy of the Jacobs model by relating the fluxes, J_i , of each species to all of the driving potentials, X_i , in the system through coupling coefficients, L_{ij} :

$$\begin{aligned} J_1 &= L_{11}X_1 + L_{12}X_2 + \dots + L_{1n}X_n \\ J_2 &= L_{21}X_1 + L_{22}X_2 + \dots + L_{2n}X_n \\ &\vdots \\ J_n &= L_{n1}X_1 + L_{n2}X_2 + \dots + L_{nn}X_n \end{aligned} \tag{2.3.1}$$

These equations also can be expressed in their conjugate form where the driving forces are expressed as functions of the species fluxes. This leads to the so-called resistance formulation and will be discussed in the following section. It should be noted that use of these equations implies the assumption that the system is not too far removed from equilibrium as the linear relationship between forces and fluxes can not necessarily be expected to hold as the forces increase.

The development begins by considering a system of two chambers separated by a membrane. Each chamber contains a solution of a single

permeating solute in the same solvent (presumably water). The regions are assumed to be in thermal equilibrium characterized by a single temperature, T (see Figure 2.3.1).

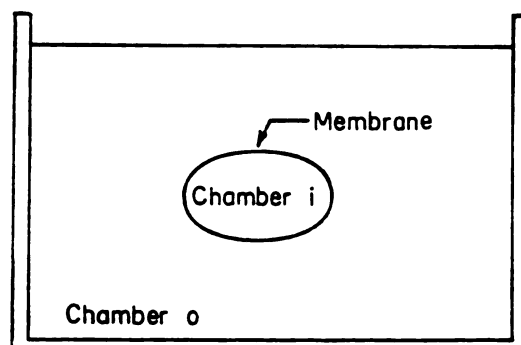


Figure 2.3.1, The two-chamber system.

One of the basic principles of irreversible thermodynamics is that if an adiabatic system undergoes a change of state via a reversible process then the entropy of the system will remain unchanged. If, however, the adiabatic process is irreversible then the system will experience a net increase in entropy. The rate at which entropy is produced in a system which is in thermal equilibrium can be expressed as the sum of the product of the flows in the system and their corresponding driving forces. The system of interest is defined as the chamber designated chamber (i), which will be referred to as the cell, and includes the membrane itself. The rate of entropy production for this system is given by the expression¹:

¹ Kedem and Katchalsky, [3], eq(12).

$$d_i S/dt = (1/T) \left[(\mu_W^o - \mu_W^i) dN_W^i/dt + (\mu_S^o - \mu_S^i) dN_S^i/dt \right] \quad (2.3.2)$$

where S is the entropy of the system, T is the absolute temperature, μ is the chemical potential, N_j^i is the number of moles of component (j) inside the cell, and t is time. The grouping $d_i()/dt$ implies internal to the system. The superscript (i) implies inside the cell while the superscript (o) implies outside the cell. A dissipation function per unit area of membrane, A , is defined for convenience as:

$$\Omega = (T/A) d_i S/dt = (1/A) \left[(\mu_W^o - \mu_W^i) dN_W^i/dt + (\mu_S^o - \mu_S^i) dN_S^i/dt \right] \quad (2.3.3)$$

letting each mole flux be represented by: $\dot{n}_j = (1/A) dN_j^i/dt$ (2.3.4)

$$\Omega = (\mu_W^o - \mu_W^i) \dot{n}_W + (\mu_S^o - \mu_S^i) \dot{n}_S \quad (2.3.5)$$

Thus for this lumped analysis the dissipation function, Ω , is the sum of the products of the fluxes $\dot{n}$ and their corresponding forces (the differences in chemical potentials).

The system of particular interest in this study is that of a membrane separating two solutions each made up of many solutes of which the membrane is only permeable to one. Further, we are interested in the case where there is no transmembrane hydrostatic pressure difference ($\Delta P=0$) as most biological membranes will not support such a difference, [3].

An alternate set of fluxes will be defined for use in the follow-

ing development. The total volume flux through the membrane is given by the expression:

$$J_v = \dot{n}_w \bar{v}_w + \dot{n}_s \bar{v}_s \quad (2.3.6)$$

The diffusion flux through the membrane is given as:

$$J_d = (\dot{n}_s / \bar{c}_s) - (\dot{n}_w / c_w) \quad (2.3.7)$$

where the quantity $\bar{c}_s$ is defined by the relation:

$$\Delta c_s / \bar{c}_s = \ln(c_s^0 / c_s^1) \quad (2.3.8)$$

If the transmembrane concentration difference is small then $\Delta c_s / \bar{c}_s \ll 1$ and $\bar{c}_s \approx (c_s^1 + c_s^0) / 2$. Kedem and Katchalsky describe the diffusion flow as the "relative velocity of solute versus solvent which is a measure for exchange flow", [3]. Note that $c_w = (1 - \xi_s - \xi_i) / \bar{v}_w$ where ξ represents the volume fraction and the subscript (s) implies the permeable solute and (i) the sum of all the impermeable solutes. Thus if the solution is assumed dilute so that the total volume fraction of the solutes is small compared to 1 then $c_w \approx 1 / \bar{v}_w$. This allows one to express J_d as:

$$J_d = (\dot{n}_s / \bar{c}_s) - \bar{v}_w \dot{n}_w \quad (2.3.9)$$

In order to derive an expression for the entropy dissipation function in terms of these new fluxes we must define a set of conju-

gate forces for this set of flows. Let the conjugate forces for J_v and J_d be represented by X_v and X_d respectively. Using these fluxes and their conjugate forces the dissipation function analogous to (2.3.5) then becomes:

$$\Omega = (\dot{n}_s \bar{v}_s + \dot{n}_w \bar{v}_w) X_v + \left[(\dot{n}_s / \bar{c}_s) - (\bar{v}_w \dot{n}_w) \right] X_d \quad (2.3.10)$$

Because the entropy dissipation function must remain unchanged under the transform one can equate the two expressions for Ω represented by (2.3.5) and (2.3.10) to form a single expression which excludes the term Ω . The fluxes $\dot{n}_s$ and $\dot{n}_w$ are independent so their coefficients on each side of this newly formed equation can be equated yielding two independent expressions relating the new forces, X_v and X_d , to the old set of forces, $\Delta\mu_s$ and $\Delta\mu_w$. These two expressions can be solved for X_v and X_d in terms of $\Delta\mu_s$ and $\Delta\mu_w$. The resulting expressions are given by:

$$X_v = c_w \Delta\mu_w + \bar{c}_s \Delta\mu_s \quad (2.3.11)$$

$$X_d = (1 - \xi_s) \bar{c}_s \Delta\mu_s - \xi_s c_w \Delta\mu_w \quad (2.3.12)$$

where $\xi_s = \bar{v}_s \bar{c}_s$. If the solutions are assumed to be ideal and the volume fractions of all the solutes are assumed to be small then²:

$$\Delta\mu_w = -(1/c_w) \left[RT\Delta c_s + RT\Delta c_1 \right] \quad (2.3.13)$$

² Kedem and Katchalsky, [3], eq. 35.

where the subscript (i) implies the sum of the impermeable solutes and the subscript (s) implies the permeable solute. Similarly for the permeable solute:

$$\Delta\mu_s = RT\Delta c_s / \bar{c}_s \quad (2.3.14)$$

Introducing (2.3.13) and (2.3.14) into (2.3.11) and (2.3.12):

$$X_v = -RT\Delta c_i \quad (2.3.15)$$

$$X_d = RT\Delta c_s + \xi_s RT\Delta c_i \quad (2.3.16)$$

Using the Onsager phenomenological equations each of the fluxes in the system is assumed to be a function of all of the driving potentials in the system. For the case of two permeating species (the permeable solute and the solvent) one will have two independent flows and hence two independent forces related by the expressions:

$$\begin{aligned} J_1 &= L_{11}X_1 + L_{12}X_2 \\ J_2 &= L_{21}X_1 + L_{22}X_2 \end{aligned} \quad (2.3.17)$$

The (L's) in these expressions are called the phenomenological coefficients and are governed by Onsager's Law which requires that the cross coefficients be equal, [4]:

$$L_{12} = L_{21} \quad (2.3.18)$$

For the system under consideration here (2.3.12) becomes:

$$\begin{aligned} J_v &= L_p X_v + L_{pd} X_d \\ J_d &= L_{pd} X_v + L_d X_d \end{aligned} \quad (2.3.19)$$

The values of the coefficients L_p , L_{pd} , and L_d are restricted by the requirement, under the principles of irreversible thermodynamics, that the entropy production and hence the dissipation function must be greater than or equal to zero. Substituting (2.3.19) into (2.3.10) using (2.3.6) and (2.3.7) the dissipation function can be expressed as:

$$\Omega = L_p X_v^2 + 2L_{pd} X_v X_d + L_d X_d^2 \geq 0 \quad (2.3.20)$$

Since either X_v or X_d can be made to go to zero independently this restricts both L_p and L_d to positive values only, and requires that the magnitude of L_{pd} be such that:

$$L_{pd}^2 \leq L_p L_d \quad (2.3.21)$$

Most investigators will use a transform changing from the phenomenological coefficients L_p , L_d , and L_{pd} to an alternate set of coefficients L_p , σ , and ω , [3,6,7,8,9,10]. The Staverman reflection coefficient, σ , is defined by the relation, [6]:

$$\sigma = \left[1 + \Delta\mu_w / \bar{v}_w \bar{c}_s \Delta\mu_s \right]_{J_v=0} \quad (2.3.22)$$

The condition of $J_v=0$ occurs when the solute and solvent are flowing in opposite directions with magnitudes such that the volume of the cell remains constant with time. Using (2.3.6), (2.3.7), and (2.3.19) one can show that:

$$\sigma = -L_{pd}/L_p \quad (2.3.23)$$

The solute permeability coefficient, ω , is defined by the relation, [12]:

$$\dot{n}_s = \left[\omega \bar{c}_s \Delta \mu_s \right]_{J_v=0} \quad (2.3.24)$$

so that using (2.3.6), (2.3.7), and (2.3.19) it can be shown that:

$$\omega = \bar{c}_s \left[(L_p L_d - L_{pd}^2) / L_p \right] = (L_d - L_p \sigma^2) \bar{c}_s \quad (2.3.25)$$

The restrictions on L_p and L_d and that represented by equation (2.3.17) imply a restriction on ω such that $\omega \geq 0$. The solvent permeability coefficient, L_p , remains unchanged with respect to the transformation.

By introducing (2.3.23) and (2.3.25) into (2.3.19) one can show that:

$$J_v = -L_p RT \sum \Delta c_i - \sigma L_p RT \left[\Delta c_s + \xi_s \sum \Delta c_i \right] \quad (2.3.26)$$

$$J_d = RT [\bar{v}_s \omega + \sigma L_p (1-\sigma)] \sum \Delta c_i + RT [(\omega/\bar{c}_s) + \sigma L_p] \Delta c_s \quad (2.3.27)$$

Since one must usually keep track of the internal solute content in order to calculate the internal solute concentration it is often convenient to work with the total volume flux, J_v , and the solute mole flux, $\dot{n}_s$, rather than J_v and J_d . From (2.3.6) and (2.3.7) one can show that:

$$\dot{n}_s = (J_v + J_d)\bar{c}_s \quad (2.3.28)$$

so that:

$$\dot{n}_s = \bar{c}_s(1-\sigma)J_v + \omega RT[\Delta c_s + \xi_s \sum \Delta c_i] \quad (2.3.29)$$

Ignoring the contribution of the term $\xi_s \sum \Delta c_i$, which represents the contribution of the impermeable solutes to the force X_d , in each expression (as it is normally very small compared to Δc_s)³ these expressions reduce to the more commonly used set:

$$J_v = -L_p RT \sum \Delta c_i - \sigma L_p RT \Delta c_s \quad (2.3.30)$$

$$\dot{n}_s = -RT\bar{c}_s \left[L_p(1-\sigma) \sum \Delta c_i + [\sigma L_p(1-\sigma) - (\omega/\bar{c}_s)] \Delta c_s \right] \quad (2.3.31)$$

It may be of interest to express the solvent flux alone. Using (2.3.6) and (2.3.7) one can show that:

³ Kedem and Katchalsky, [3], pg. 238.

$$\dot{n}_w = -(RT/\bar{v}_w) \left[L_p \sum \Delta c_i + (L_p \sigma + \bar{v}_s) \Delta c_s \right] \quad (2.3.32)$$

It is worth noting once again the restrictions under which this set of equations remain valid. The system considered was that of two compartments separated by a membrane and in thermal equilibrium. While there may be many solutes present on either side of the membrane the membrane is assumed to be permeable to only one of those solutes and the solvent. Further, the solutions are assumed to be ideal and dilute due to the approximations for the chemical potential used. Also it has been assumed that the driving forces in the system are sufficiently small such that a linear relation exists between all the driving forces and the resulting flow of each species. For the final version of the equations presented, (2.3.30) and (2.3.31), it is also assumed that no hydrostatic pressure difference exists across the membrane. This assumption is not inherent in the overall development so that if a pressure difference is believed to exist in the problem of interest this can be accounted for in the K-K model. For situations which comply reasonably well with these restrictions the K-K formulation has been found to provide quite reasonable correlation with experimental data, [8,10,11].

2.4 The K-K Resistance Formulation

In a work published in 1961 Kedem and Kachalsky presented an alternate formulation of their membrane permeation model, [7]. This formulation leads to a set of frictional coefficients governing the process rather than the phenomenological coefficients as derived in the previous section, [3]. These frictional coefficients allow for a more direct physical interpretation of the permeability parameters L_p , σ , and ω . While this formulation leads to more complicated expressions than the previous formulation it has the advantage of physical interpretability. It also represents an advantage in that the mechanical coefficients encountered are largely concentration independent whereas the phenomenological coefficients are generally concentration dependent, [7]. While this model may not be practical for calculation purposes, and will not be used for such in the present investigation, as it introduces additional unknown factors such as the distribution coefficient of solute in the membrane, it does provide some interesting insights into the passive transport process.

In this formulation relations reciprocal to those represented by (2.3.1) are used. In this form the driving forces, X_i , in the system are assumed to be linear functions of each of the fluxes, J_i , in the system:

$$\begin{aligned}
X_1 &= R_{11}J_1 + R_{12}J_2 + \dots + R_{1n}J_n \\
X_2 &= R_{21}J_1 + R_{22}J_2 + \dots + R_{2n}J_n \\
&\vdots \\
X_n &= R_{n1}J_1 + R_{n2}J_2 + \dots + R_{nn}J_n
\end{aligned}
\tag{2.4.1}$$

Here the coefficients R_{ij} are in essence frictional coefficients. The numerical values of these coefficients are restricted by the requirement that the entropy production and hence the dissipation function represented by (2.3.2) and (2.3.3) must be positive. This restricts the straight coefficients, R_{ii} , to positive values and the magnitude of the cross coefficients, R_{ij} , to the condition:

$$R_{ij}^2 \leq R_{ii}R_{jj} \tag{2.4.2}$$

These coefficients must also satisfy Onsager's Law so that, [4]:

$$R_{ij} = R_{ji} \tag{2.4.3}$$

As before, a system of one permeable solute and one solvent (usually water) will be considered. Again we will assume that the system is composed of two chambers separated by a membrane and in thermal equilibrium. The solutions in both chambers are assumed to be well-stirred so that no unstirred layers exist at the boundaries of the membrane. The membrane thickness will be represented by the quantity Δx so that the membrane covers the region $0 < x < \Delta x$ (see Figure 2.4.1).

The local molar flux of solute and solvent can be written as:

$$\begin{aligned} J_s &= C_s v_s = dn_s/dt \\ J_w &= C_w v_w = dn_w/dt \end{aligned} \quad (2.4.4)$$

where n_i is the number of moles of species (i) per unit area and v_i is the local velocity of species (i).

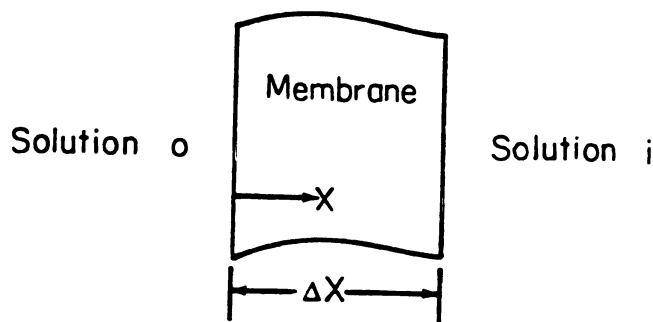


Figure 2.4.1, Membrane cross-section.

In order to develop a set of local equations which can be integrated across the membrane the system will be assumed to be in a steady state condition. This will allow us to express various quantities as total derivatives with respect to x rather than having to concern ourselves with a set of partial differential equations in x and t . This implies that the concentrations, C_i , are only functions of x , as are the velocities, v_i . Note also that the concentration, C_i , is defined as the number of moles of species (i) per unit volume of membrane not per unit volume of solution. This definition makes C_i a local concentration accounting for the membrane volume as well as

the volume of the solution. The fluxes, J_i , will be assumed to be independent of x and equal to the macroscopic or total transmembrane fluxes.

The problem is subject to the following boundary conditions:

$$\begin{aligned}\mu_s(x=0) &= \mu_s^0 \\ \mu_w(x=0) &= \mu_w^0 \\ \mu_s(x=\Delta x) &= \mu_s^i \\ \mu_w(x=\Delta x) &= \mu_w^i\end{aligned}\tag{2.4.5}$$

Since the objective is to derive a local set of equations to be integrated across the membrane, local driving potentials must be used. The local driving force is the local gradient in chemical potential:

$$X_j = -d\mu_j/dx\tag{2.4.6}$$

(which has units of dyne-cm/mole-sec). This is consistent with the development which led to equation (2.3.5) except that in section 2.3 the internal mechanics of the membrane were not considered and a set of transmembrane equations, as opposed to a set of local equations, was being developed. Using (2.4.6) for this system (2.4.1) becomes:

$$\begin{aligned}-d\mu_w/dx &= R_{ww}J_w + R_{ws}J_s \\ -d\mu_s/dx &= R_{sw}J_w + R_{ss}J_s\end{aligned}\tag{2.4.7}$$

where: $R_{sw} = R_{ws}$

It is now assumed that under steady flow conditions the thermodynamic force acting on a given component will be counterbalanced by a sum of mechanical friction forces so that there will be no net acceleration. These forces are assumed to be comprised of a set of forces due to a given species interacting with each of the other permeating species in the system and a force due to each species interacting with the membrane itself. For the two component system being considered here this translates to:

$$\begin{aligned} X_s &= -F_{sw} - F_{sm} \\ X_w &= -F_{ws} - F_{wm} \end{aligned} \quad (2.4.8)$$

where F represents the frictional force and the subscripts s , w , and m refer to solute, solvent, and membrane respectively. Thus the factor F_{sm} would represent the frictional force between the solute and membrane per unit mole of solute. Each of the frictional forces is in turn represented by an expression relating the force, F_{ij} , to the relative velocity of component (i) to component (j) through a mechanical friction coefficient, f_{ij} , per mole of (i) component:

$$F_{ij} = -f_{ij}(v_i - v_j) \quad (2.4.9)$$

Choosing the reference frame such that $v_m = 0$, (2.4.6) becomes:

$$\begin{aligned} -d\mu_s/dx &= f_{sw}(v_s - v_w) + f_{sm}v_s = v_s(f_{sw} + f_{sm}) - v_w f_{sw} \\ -d\mu_w/dx &= f_{ws}(v_w - v_s) + f_{wm}v_w = -v_s f_{ws} + v_w(f_{ws} + f_{wm}) \end{aligned} \quad (2.4.10)$$

It is also possible to rewrite this set of equations as:

$$\begin{aligned} -d\mu_s/dx &= [(f_{sw}+f_{sm})/C_s]J_s - (f_{sw}/C_w)J_w \\ -d\mu_w/dx &= -(f_{ws}/C_s)J_s + [(f_{ws}+f_{wm})/C_w]J_w \end{aligned} \quad (2.4.11)$$

Comparing (2.4.11) with (2.4.7) readily shows the frictional coefficients to be:

$$\begin{aligned} R_{ss} &= (f_{sw}+f_{sm})/C_s & R_{sw} &= -f_{sw}/C_w \\ R_{ws} &= -f_{ws}/C_s & R_{ww} &= (f_{ws}+f_{wm})/C_w \end{aligned} \quad (2.4.12)$$

The restriction represented by Onsager's Law requires that:

$$-f_{sw}/C_w = -f_{ws}/C_s \quad (2.4.13)$$

Thus only three independent frictional coefficients exist while the fourth is dependent. This is analogous to the need for three independent permeability coefficients in the model presented in section 2.3 .

Kedem and Katchalsky point out that " f_{sw} " is of the same nature as the friction coefficient of free diffusion f_{sw}^0 given by Einstein's equation:

$$f_{sw}^0 = RT/D^0 \quad (2.4.14)$$

and may therefore be assumed to be approximately independent of the

local concentration as D^0 is essentially concentration independent". D^0 is the diffusion coefficient of the solute in free solution. This implies that due to the relation expressed by (2.4.13) the coefficient f_{ws} must be strongly dependent on the solute concentration. This would imply that substituting f_{sw} for f_{ws} would be the most reasonable strategy in order to yield friction coefficients which are independent of the concentration. This leads to the expressions:

$$\begin{aligned} -d\mu_s/dx &= [(f_{sw}+f_{sm})/C_s]J_s - (f_{sw}/C_w)J_w \\ -d\mu_w/dx &= -(f_{sw}/C_w)J_s + \left[[(C_s/C_w)f_{sw}+f_{wm}]/C_w\right]J_w \end{aligned} \quad (2.4.15)$$

or:

$$\begin{aligned} -C_s d\mu_s/dx &= (f_{sw}+f_{sm})J_s - (C_s/C_w)f_{sw}J_w \\ -C_w d\mu_w/dx &= -f_{sw}J_s + [(C_s/C_w)f_{sw}+f_{wm}]J_w \end{aligned} \quad (2.4.16)$$

It must be noted that (2.4.16) represents a set of local equations only and that in order to be truly useful as experimental tools they must first be integrated across the membrane. The integration procedure is presented in detail in Kedem and Katchalsky's paper, [7], and for the sake of brevity only the results are presented here. After integrating from 0 to Δx (2.4.16) becomes:

$$\begin{aligned} K\Delta\pi_s &= -J_w \left[f_{sw} \bar{v}_w \bar{K}_c C_s / \xi_{wm} \right] \Delta x + J_s (f_{sw}+f_{sm}) \Delta x \\ \xi_{wm} (\Delta P - \Delta\pi_1 - \Delta\pi_s) &= J_w \left[f_{wm} + (f_{sw} \bar{v}_w \bar{K}_c C_s) / \xi_{wm} \right] \Delta x - J_s f_{sw} \Delta x \end{aligned} \quad (2.4.17)$$

where ξ_{wm} is the volume fraction of water in the membrane, and the quantity $\overline{K_c c_s}$ represents the mean value of solute concentration across the membrane and is given by:

$$\overline{K_c c_s} = (1/\Delta x) \int_0^{\Delta x} K_c^x c_s dx \quad (2.4.18)$$

where K_c^x is a local distribution coefficient of the solute in the membrane given by:

$$K_c^x = C_s^x / c_s^x \quad (2.4.19)$$

C_s represents the local concentration of solute based upon the total volume of the membrane itself. The quantity c_s represents the concentration of a free solution of equal solute chemical potential. If the solution is ideal in both chambers then K_c is constant and:

$$K_c = K_{ideal} \quad (2.4.20)$$

$$\overline{K_c c_s} = K_{ideal} (c_s^0 + c_s^i) / 2 \quad (2.4.21)$$

A further simplification also occurs if the solute and solvent penetrate the cell only by passing through solution-filled capillaries in the membrane rather than through reaction with the membrane itself. Kedem and Katchalsky observed that for a case such as this the solution in the capillaries approaches the composition of the free solution. This means that the solute distribution coefficient becomes equal to the volume fraction of water in the membrane.

2.5 Comparison of K-K Resistance and Permeability Formulations

By comparing the results of the two different formulations of Kedem and Katchalsky presented in the previous two sections interesting relationships appear. These relationships give added meaning to the physical significance of the permeability parameters which are commonly used to describe transport of materials across a membrane. The relationships derived here will not be used for calculation purposes in the present work but rather are presented for the insights into the passive transport process which they provide. An alternate method of comparison to that presented by Kedem and Katchalsky, [7], will be presented here. The overall strategy will be to pose the resistance equations represented by (2.4.17) into the same form as the permeability equations with flows expressed as functions of the driving forces.

Let us begin by recalling the K-K permeability equations in the form:

$$J_w = -(1/\bar{v}_w) [L_p \Delta\pi_1 + (L_p \sigma + \bar{v}_s \omega) \Delta\pi_s] \quad (2.5.1)$$

$$J_s = -\bar{c}_s [[L_p(1-\sigma) - \bar{v}_s \omega] \Delta\pi_1 + [L_p \sigma(1-\sigma) - (\omega/\bar{c}_s)] \Delta\pi_s] \quad (2.5.2)$$

where we have included the $\xi_s \sum \Delta c_i$ term which was neglected in (2.3.30) and (2.3.31). The comparable set of resistance equations, represented by (2.4.17), now must be expressed in this same form. One way to do this is to solve both of the equations for J_s and equate the results. The resulting expression can then be solved for J_w to give:

$$J_w = \frac{Kf_{sw}\Delta\pi_s - \xi_{wm}(f_{sw}+f_{sm})(\Delta\pi_i+\Delta\pi_s)}{\Delta x[f_{wm}(f_{sw}+f_{sm}) + \lambda f_{sw}f_{sm}]} \quad (2.5.3)$$

where: $\lambda = \bar{v}_w \bar{K}_c c_s / \xi_{wm}$. λ is used only for convenience of presentation. Note that it has been assumed that no transmembrane difference in hydrostatic pressure exists ($\Delta P=0$) in order to be compatible with the Kedem and Katchalsky set.

In a similar manner, by solving both of the equations represented by (2.4.17) for J_w , equating the results, and solving for J_s it can be shown that:

$$J_s = \frac{K(f_{wm}+\lambda f_{sw})\Delta\pi_s - \xi_{wm}\lambda f_{sw}(\Delta\pi_i+\Delta\pi_s)}{\Delta x[f_{wm}(f_{sw}+f_{sm}) + \lambda f_{sw}f_{sm}]} \quad (2.5.4)$$

It is now possible to equate the coefficients of the independent osmotic pressure differences to obtain the desired cross relationships. For instance by equating the coefficients of $\Delta\pi_i$ in (2.5.1) and (2.5.3) it can be shown that:

$$L_p = \frac{\bar{v}_w \xi_{wm} (f_{sw} + f_{sm})}{\Delta x[f_{wm}(f_{sw}+f_{sm}) + \lambda f_{sw}f_{sm}]} \quad (2.5.5)$$

A check on the units for this equation shows the right-hand side to have dimensions of (length³/force-time) which are the proper dimensions for L_p , the solvent permeability.

This equation also can be written:

$$L_p = (\bar{v}_w \zeta_{wm} / \Delta x) \left[f_{wm} + \lambda f_{sw} \left(\frac{1}{1 + (f_{sw} / f_{sm})} \right) \right]^{-1} \quad (2.5.6)$$

In this form it is apparent that L_p is inversely proportional to the sum of the solvent-membrane friction factor, f_{wm} , and an additional factor characterizing solute-solvent interactions (the bracketed term). The direct dependence of solvent permeability, L_p , on volume fraction of solvent in the membrane is also apparent. This seems consistent as a membrane with a larger solution content would be expected to be more permeable than an otherwise identical membrane with a smaller solution content as the larger solution content would imply that a larger volume fraction of the membrane is available for flow.

Kedem and Katchalsky developed an expression for L_p only for a single, very limited case. It would be expected that (2.5.6) should reduce to the K-K expression under the same restrictions. To show that this is the case we begin by rearranging (2.5.6) as:

$$L_p = (\bar{v}_w \zeta_{wm} / \Delta x) \left[f_{wm} + \lambda f_{sm} \left(\frac{1}{1 + (f_{sm} / f_{sw})} \right) \right]^{-1} \quad (2.5.7)$$

Consider now only the case of a coarse non-selective membrane. Kedem and Katchalsky give the non-selectivity condition as:

$$f_{sm} / \bar{v}_s = f_{wm} / \bar{v}_w \quad (2.5.8)$$

using this in (2.5.7) one can show that:

$$L_p = (\bar{v}_w \xi_{wm} / \Delta x) \left[f_{wm} \left[1 + \lambda (\bar{v}_s / \bar{v}_w) \left(\frac{1}{1 + (f_{sm} / f_{sw})} \right) \right] \right]^{-1} \quad (2.5.9)$$

Kedem and Katchalsky further restrict the consideration to a case with no transmembrane concentration difference. This reduces the value of the mean concentration of solute within the membrane to the concentration of the free solution multiplied by the (constant) distribution coefficient:

$$\bar{K}_c \bar{c}_s = K_c \bar{c}_s \quad (2.5.10)$$

While there would be no flow in this case one can still derive a simplified expression for the solvent permeability. Just because there is no flow this does not imply that the solvent permeability does not exist.

For membranes with a capillary structure (see Figure 2.5.1) the distribution coefficient is equal to the volume fraction of water in the membrane so that:

$$\lambda = \bar{v}_w \bar{c}_s \quad (2.5.11)$$

and (2.5.9) becomes:

$$L_p = (\bar{v}_w \bar{\epsilon}_{wm} / \Delta x) \left[f_{wm} \left[1 + \bar{v}_s \bar{c}_s \left(\frac{1}{1 + (f_{sm} / f_{sw})} \right) \right] \right]^{-1} \quad (2.5.12)$$

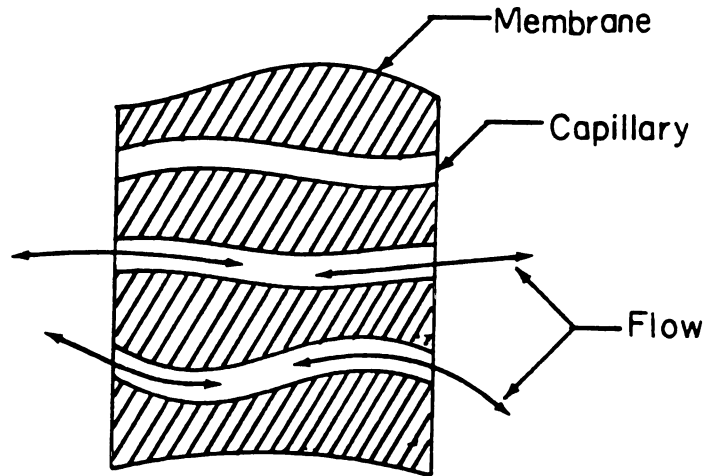


Figure 2.5.1, Membrane with capillary structure.

Unless the membrane is somehow actively inducing the flow of either the solvent or the solute the frictional coefficients will be positive. That is to say that the presence of the membrane will act to inhibit the flow. For relatively dilute solutions with positive friction coefficients:

$$\bar{v}_s \bar{c}_s \ll 1 \quad (2.5.13)$$

$$1 + (f_{sm} / f_{sw}) > 1 \quad (2.5.14)$$

so that we can further reduce (2.5.12) to:

$$L_p = (\bar{v}_w \bar{\epsilon}_{wm}) / (\Delta x f_{wm}) \quad (2.5.15)$$

This is indeed the same expression derived by Kedem and Katchalsky for L_p under these conditions¹. This lends greater confidence to the validity of the more general expression for L_p represented by (2.5.6) derived for the first time in this work.

Further confidence in the validity of (2.5.6) in particular and to the present strategy as a whole is gained by consideration of the Staverman reflection coefficient. By equating the coefficients of $\Delta\pi_s$ in (2.5.1) and (2.5.3) one can show that:

$$\sigma = -(\bar{v}_s \omega / L_p) - (\bar{v}_w / L_p \Delta x) \left[\frac{K f_{sw} - \xi_{wm}(f_{sw} + f_{sm})}{f_{wm}(f_{sw} + f_{sm}) + \lambda f_{sw} f_{sm}} \right] \quad (2.5.16)$$

Substituting for L_p using (2.5.5) in the last term only and rearranging gives:

$$\sigma = 1 - (\bar{v}_s \omega / L_p) - \left[(K f_{sw}) / [\xi_{wm}(f_{sw} + f_{sm})] \right] \quad (2.5.17)$$

This is again identical to the relationship derived by Kedem and Katchalsky. This same result can be achieved by equating the coefficients of $\Delta\pi_1$ in (2.5.2) and (2.5.4).

Kedem and Katchalsky point out that if the solute and solvent penetrate the cell by different paths so that there is no solute-solvent interaction then $f_{sw}=0$. Using this in (2.5.17) gives

¹ Kedem and Katchalsky, [7], eq. 4-21.

the expression:

$$\sigma = 1 - (\bar{v}_s \omega / L_p) \quad (2.5.18)$$

This relationship then can be used as a test for the condition of non-interaction between solute and solvent flows. It is interesting to note that while this appears to imply that the solute and solvent flows are uncoupled they are in fact coupled in the sense that the magnitude of $\Delta\mu_w$ effects the solute flow as well as the solvent flow and that the magnitude of $\Delta\mu_s$ effects both the solvent flow and the solute flow. This coupling is implied in the phenomenological equations.

In order to carry the development of σ farther more information about the nature of the solute permeability ω is required. One way to solve for ω as a function of the friction coefficients would be to equate the remaining coefficients (those of $\Delta\pi_s$ in (2.5.2) and (2.5.4)), substitute for L_p and σ , and solve for ω . This however requires solving an extremely complex quadratic equation. For this reason only the form developed by Kedem and Katchalsky will be considered here.

Kedem and Katchalsky developed the following expression for the solute permeability:

$$\omega = K[1 - \xi_s(1-\sigma)] / [\Delta x(f_{sw} + f_{sm})] \quad (2.5.19)$$

In most cases the solute volume fraction, ξ_s , will be very small compared to 1.0. This would reduce the above expression to the more simple form:

$$\omega = K / [\Delta x(f_{sw} + f_{sm})] \quad (2.5.20)$$

If the membrane is assumed to be of the capillary structure then $K = \xi_{wm}$ and this expression reduces further to:

$$\omega = \xi_{wm} / [\Delta x(f_{sw} + f_{sm})] \quad (2.5.21)$$

These relationships show that the solute permeability is inversely proportional to the sum of the solute-water and solute-membrane friction factors.

It is possible to use the above relationships in (2.5.17) to develop an expression for the reflection coefficient, or to use (2.5.17) in (2.5.19) to develop an expression for solute permeability in which only the friction coefficients appear. These procedures lead to complicated expressions which are not particularly useful and hence will not be presented.

In summary this comparison has shown that L_p can be expressed as a function of all the friction factors with f_{wm} playing the dominant role (see equations 2.5.12 and 2.5.15). The reflection coefficient, σ , was shown to be a function of the ratio of the solute permeability to the solvent permeability, ω/L_p , and an additional factor character-

izing solute-solvent interactions. An expression for ω was not derived using this method but Kedem and Katchalsky showed that ω is dependent on the factors f_{sw} and f_{sm} which characterize the solute-solvent and solute-membrane interactions respectively.

2.6 A Power Series Solution to the K-K Permeability Equations

In 1967 Johnson and Wilson, [8], developed an approximate solution to the Kedem and Katchalsky permeability equations represented by equations (2.3.30) and (2.3.31). The solution they presented was based on a perturbation analysis and a power series expansion of the K-K equations.

While the values obtained from this solution may not be as accurate as one might wish they can be used as starting values for other more accurate parameter estimation routines. The model presented here gives volume as a function of time directly from a closed-form analytical expression and hence is readily evaluated. Other methods which deal with the full set of K-K equations in differential form will require the repeated numerical integration of the governing equations and hence will require much greater computational effort. In these latter types of routines good starting estimates of the parameters can significantly reduce the number of iterations required to reach the final values. In many cases this may mean the difference between a routine converging to a solution or not. Thus by using the model of Johnson and Wilson as the generator of starting values for other more accurate routines one should be able to realize a significant reduction in total computational effort required to estimate permeability parameters. While this method has not been utilized in the present work it is presented in anticipation of future work to be conducted.

The development begins by considering the K-K equations in the

form:

$$dV/dt = L_p RTA \left[[(c_o V_o + \sigma N_s)/V] - \sigma c_s - c_o \right] \quad (2.6.1)$$

$$dN_s/dt = \omega RTA \left[c_s - (N_s/V) \right] + \bar{c}_s (1-\sigma) dV/dt \quad (2.6.2)$$

where N_s is the number of moles of the permeating solute inside the cell, V is the volume of the cell, V_o is the initial cell volume, A is the surface area of the cell, c_o is the initial concentration of all the impermeable solutes, c_s is the external concentration of the permeable solute, and $\bar{c}_s$ is the mean transmembrane concentration of the permeable solute defined by equation (2.3.8). L_p , ω , and σ are the solvent permeability, solute permeability, and reflection coefficient respectively. These equations carry the implicit assumptions that the cell is in equilibrium prior to time zero so that the impermeable solute concentration is the same inside and out, and that none of the permeating solute is present inside or outside the cell. At time equal to zero the extracellular concentration of the permeating solute undergoes a step change from zero to c_s and remains constant thereafter. It will also be assumed that the surface area of the cell remains constant.

Johnson and Wilson define an alternate set of permeability coefficients by combining the factor RT with L_p and ω such that:

$$P_w = L_p RT \quad (2.6.3)$$

$$P = \omega RT \quad (2.6.4)$$

This gives P_w typical units of $(\text{cm}^4/\text{mole-sec})$ and P typical units of (cm/sec) . Using (2.6.3) and (2.6.4) in (2.6.1) and (2.6.2) one gets:

$$dV/dt = AP_w[(c_0V_0 + \sigma N_s)V^{-1} - \sigma c_s - c_0] \quad (2.6.5)$$

$$dN_s/dt = PA[c_s - (N_s/V)] + \bar{c}_s(1-\sigma)dV/dt \quad (2.6.6)$$

This set of equations is then nondimensionalized using the following groups:

$$\begin{aligned} V^* &= V/V_0 & N^* &= N_s/c_sV_0 \\ \tau &= P_wAc_0t/V_0 & b &= P/P_wc_0 \\ \alpha &= \sigma c_s/c_0 \end{aligned} \quad (2.6.9)$$

V^* is the volume nondimensionalized with respect to the initial volume, N^* is a nondimensional permeable solute content, and τ is nondimensional time. The factors b and α are used for convenience of notation. Using these groups in (2.6.5) and (2.6.6) yields the expressions:

$$V^*dV^*/d\tau = 1 - V^* + \alpha(N^*-V^*) \quad (2.6.8)$$

$$V^*dN^*/d\tau = b(V^*-N^*) + (1-\sigma)(\bar{c}_s/c_s)V^*dV^*/d\tau \quad (2.6.9)$$

This set of equations is subject to the initial conditions:

$$V^*(0) = 1 \quad N^*(0) = 0$$

In a typical case the solvent will penetrate the membrane faster than the solute so that the cell will initially shrink in size as the solvent will be leaving the cell faster than the solute is entering. Eventually the internal solution will reach a concentration high enough to cause the solvent to begin reentering the cell and the cell volume will increase. This is the typical "shrink-swell" behaviour observed for many cases. At the point where the cell reaches its minimum volume $dV^*/d\tau=0$ so that by equation (2.6.8) we see that:

$$V_m^* = (1 + \alpha N_m^*) / (1 + \alpha) \quad (2.6.10)$$

where the subscript m implies minimum. Since $N_m^* > 0$ one can show that:

$$V_m^* > (1 - \alpha) \quad (2.6.11)$$

Thus for small values of α (which from (2.6.9) implies small changes in the permeable solute concentration and/or little rejection of the permeable solute) the perturbations in volume will be small. One can then express $V^*(\tau)$ and $N^*(\tau)$ as power series in α such that:

$$V^*(\tau) = V_0^*(\tau) + \alpha V_1^*(\tau) + \alpha^2 V_2^*(\tau) + \dots \quad (2.6.12)$$

$$N^*(\tau) = N_0^*(\tau) + \alpha N_1^*(\tau) + \alpha^2 N_2^*(\tau) + \dots \quad (2.6.13)$$

where $V_i^*(\tau)$ and $N_i^*(\tau)$ are independent solutions which when weighted as indicated by powers of α and summed up will yield the full solutions V^* and N^* . Note that for values of α less than one, $\alpha^n \rightarrow 0$ as $n \rightarrow \infty$. Thus for small values of α a small number of the independent solutions

will have a significant contribution to the total solution.

The initial conditions on $V^*(\tau)$ and $N^*(\tau)$ can be satisfied by the power series expressions by requiring that $V_0^*(0)=1$; $V_i^*(0)=0$ for $(i>0)$; and that $N_i^*(0)=0$ for all (i) .

These power series expressions can be substituted into (2.6.8) and (2.6.9). One can then collect like powers of α in the resulting expressions and equate the coefficients of a given power of α on either side of a given equation. Equating the coefficients of the zeroth power of α in the equation resulting from (2.6.8) gives the expression:

$$V_0^* dV_0^*/d\tau = 1 - V_0^* \quad (2.6.14)$$

This equation and the initial condition on V_0^* are satisfied by $V_0^*=1$.

Equating the coefficients of the zeroth power of α in the equation resulting from (2.6.9) gives the expression:

$$V_0^* dN_0^*/d\tau = b(V_0^* - N_0^*) + (1-\sigma)(\bar{c}_g/c_g)V_0^*(dV_0^*/d\tau) \quad (2.6.15)$$

Using $V_0^* = 1$ in this expression reduces it to:

$$dN_0^*/d\tau = b(1 - N_0^*) \quad (2.6.16)$$

This expression and the initial condition on N_0^* are satisfied by:

$$N_0^* = 1 - \exp(-b\tau) \quad (2.6.17)$$

Equating the coefficients of the first power of α from (2.6.8) gives:

$$V_1^*(dV_0^*/d\tau) + V_0^*(dV_1^*/d\tau) = -V_1^* + N_0^* - V_0^*$$

or:
$$dV_1^*/d\tau = V_1^* - \exp(-b\tau) \quad (2.6.18)$$

This equation and the initial condition $V_1^*=0$ are satisfied by:

$$V_1^* = -(1-b)^{-1} [\exp(-b\tau) - \exp(-\tau)] \quad (2.6.19)$$

Ignoring the higher order terms one can now write:

$$V^* \approx V_0^* + \alpha V_1^* = 1 - [\alpha/(1-b)] [\exp(-b\tau) - \exp(-\tau)] \quad (2.6.20)$$

This expression will have an error of the order α^2 which is approximately of order $(1-V_m^*)^2$.

Johnson and Wilson also give a similar expression for the case of cells initially in equilibrium with a penetrating solute present and subjected to a step change in extracellular solute concentration from the initial value to zero at time equal to zero. This expression is:

$$V^* \approx 1 + [\alpha'/(1-b)] [\exp(-b\tau) - \exp(-\tau)] \quad (2.6.21)$$

where $\alpha' = \sigma N_s(0)/c_0 V_0$ and $N_s(0)$ is the total number of moles of perme-

able solute in the cell at time equal to zero. This case corresponds to the "washing" of a solute from a cell.

In order to obtain the desired permeability parameters from the above model we need three pieces of information relating the experimental data to the model. The first of these is the matching of the shape of the curve for the experimental data to the shape of the curve for the model. Johnson and Wilson suggest using the following expression to characterize the shape of the modeling curve:

$$(V^* - V_m^*) / (1 - V_m^*) = 1 - \left[\frac{[\exp(-b\tau) - \exp(-\tau)]}{[\exp(-b\tau_m) - \exp(-\tau_m)]} \right] \quad (2.6.22)$$

Note that this expression does not involve the variable α . Figure (2.6.1) shows the shape of the curve described by (2.6.22) for various values of b . This figure shows that reciprocal values of b will give the same curve. Johnson and Wilson observe that this implies that two different sets of values for the permeability parameters will satisfy the model equally well. They found however that one of these two sets would include unrealistic values such as $\sigma > 1$ and could therefore be eliminated. They also observe that the model will be insensitive to b for values of $b \approx 1$ so that special care must be taken in a computer program implementing this solution.

Once the shape of the experimental curve has been matched to that of the model and a value for b has been obtained, one can then match the time and volume scales of the model to that of the experimental

data. The time scale is matched by matching the time at which the minimum volume is reached. From (2.6.20) one can show that:

$$\tau_m^* = (\ln b)/(b-1) \quad (2.6.23)$$

so that a value of τ_m^* can be calculated. Similarly one can match the volume scale by matching the values of the minimum volume for the model to that of the experimental data:

$$V_m^* = 1 - ab^{[b/(1-b)]} \quad (2.6.24)$$

and a value of a can be obtained. Using the numerical values of a , b , and τ and the definitions of a , b , and τ given earlier one can determine the values of the permeability parameters L_p , ω , and σ as one will have three equations and three unknowns.

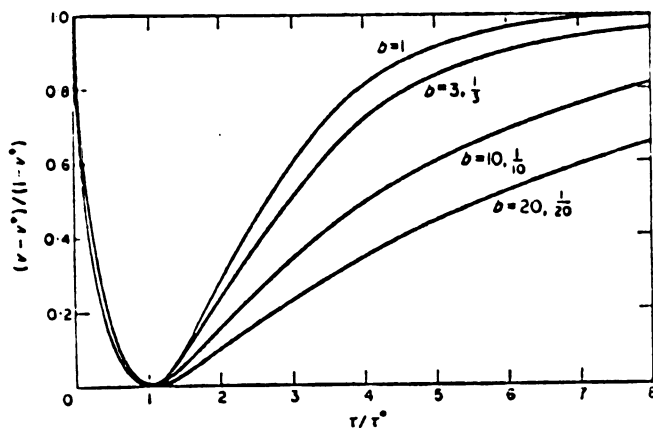


Figure 2.6.1, Shape of V^* as a function of b .

2.7 The Papanek Model

In 1978 Papanek, [12], presented a permeability model which closely paralleled the Kedem and Katchalsky permeability model, [3]. The Papanek model differs from the Kedem and Katchalsky model in that the assumption of dilute and ideal solutions is not made. Thus the Papanek model attains broader applicability at the expense of computational simplicity. This model will not be used for calculation purposes in the present work as it requires the development of certain empirical relationships for each of the solutes of interest in order to handle the non-ideality. This is beyond the scope of the present work. Since the development parallels that of Kedem and Katchalsky many references will be made to the equations of section 3 of chapter 2 in the present work.

Papanek begins by assuming that the phenomenological equations, (2.3.1), hold true. The same expressions as those used by Kedem and Katchalsky are also used for the local rate of entropy production, (2.3.2), and the entropy dissipation function, (2.3.3). Papanek's model differs from the Kedem and Katchalsky model in that Papanek does not substitute for the difference in chemical potential, $\Delta\mu$, but rather retains it as the driving force. Kedem and Katchalsky on the other hand assumed that the solutions are dilute and ideal and transformed from the $\Delta\mu$ driving force yielding the hydrostatic pressure difference and the difference in solute concentrations as the new driving forces. By retaining the chemical potential as the driving potential Papanek removes these restrictions from the model.

Papanek defines a simple set of phenomenological equations for the situation involving a binary flow of water, w, and a single permeating solute, s, using the molar mass flux, defined by (2.3.4), and the difference in chemical potential ($\Delta\mu = \mu^0 - \mu^1$) as the flows and forces respectively:

$$\begin{aligned}\dot{n}_w &= J_w = L_{11}^* \Delta\mu_w + L_{12}^* \Delta\mu_s \\ \dot{n}_s &= J_s = L_{21}^* \Delta\mu_w + L_{22}^* \Delta\mu_s\end{aligned}\quad (2.7.1)$$

where Onsager's law, [4], requires that $L_{12}^* = L_{21}^*$. Substituting these relationships into the dissipation function (2.3.5) one can show that:

$$\Omega = J_w \Delta\mu_w + J_s \Delta\mu_s \quad (2.7.2)$$

$$\Omega = L_{11}^* \Delta\mu_w^2 + 2L_{12}^* \Delta\mu_w \Delta\mu_s + L_{22}^* \Delta\mu_s^2 \quad (2.7.3)$$

The fact that the rate of entropy production and hence the dissipation function must be positive definite implies the following restrictions on the phenomenological coefficients:

$$\begin{aligned}L_{11}^* &\geq 0 \\ L_{22}^* &\geq 0 \\ (L_{12}^*)^2 &\leq L_{11}^* L_{22}^*\end{aligned}\quad (2.7.4)$$

Next the alternate set of flows derived by Kedem and Katchalsky are used:

$$J_v = J_w \bar{v}_w + J_s \bar{v}_s \quad (2.7.5)$$

$$J_d = J_s / \bar{c}_s - J_w \bar{v}_w \quad (2.7.6)$$

where J_v is the total volume flux and J_d is the velocity of solute relative to solvent. The phenomenological equations for this set of flows are:

$$J_v = L'_{11} X_v + L'_{12} X_d \quad (2.7.7)$$

$$J_d = L'_{21} X_v + L'_{22} X_d \quad (2.7.8)$$

where Onsager's law requires that $L'_{12} = L'_{21}$. The corresponding dissipation function for this set of flows and forces is:

$$\Omega = J_v X_v + J_d X_d \quad (2.7.9)$$

It is now required that the local rate of entropy production and hence the dissipation function must remain unchanged by the transform. This implies that one can equate the two expressions for Ω ((2.7.2) and (2.7.9)). Recognizing that J_w and J_s are independent one can then equate their coefficients using (2.7.2), (2.7.5), and (2.7.9) to show that:

$$X_v = \Delta\mu_w / (\bar{v}_w(1+\Theta)) + \bar{c}_s \Delta\mu_s / (1+\Theta) \quad (2.7.10)$$

$$X_d = [\Delta\mu_s - (\bar{v}_s / \bar{v}_w) \Delta\mu_w] \bar{c}_s / (1+\Theta) \quad (2.7.11)$$

where $\Theta = \bar{v}_s \bar{c}_s$. Using these expressions in (2.7.7), (2.7.8), and

(2.7.6) one can show that:

$$J_w = [\bar{v}_w(1+\theta)^2]^{-1} \left[L'_{11}[(\Delta\mu_w/\bar{v}_w) + \bar{c}_s\Delta\mu_s] + L'_{12}[\bar{c}_s\Delta\mu_s(1-\theta) - 2\theta\Delta\mu_w/\bar{v}_w] \right. \\ \left. - L'_{22}[\theta\bar{c}_s\Delta\mu_s - \theta^2\Delta\mu_w/\bar{v}_w] \right] \quad (2.7.12)$$

$$J_s = [\bar{c}_s/(1+\theta)^2] \left[L'_{11}[(\Delta\mu_w/\bar{v}_w) + \bar{c}_s\Delta\mu_s] + L'_{12}[(1-\theta)\Delta\mu_w/\bar{v}_w + 2\bar{c}_s\Delta\mu_s] \right. \\ \left. + L'_{22}[\bar{c}_s\Delta\mu_s - (\theta\Delta\mu_w/\bar{v}_w)] \right] \quad (2.7.13)$$

A conversion can now be made from the phenomenological coefficients to the more widely used set of permeability coefficients L_p , ω , and σ . The first of these, L_p , is equal to the first of the phenomenological coefficients, L'_{11} . This parameter is normally called the solvent (or water) permeability and has typical units of $\text{cm}^3/\text{dyne-sec}$. The solute permeability at zero volume flow, ω , is defined by the relationship:

$$J_s = \left[\omega \bar{c}_s \Delta\mu_s \right]_{J_v=0} \quad (2.7.14)$$

The situation of $J_v=0$ occurs when the solute and solvent are flowing in opposite directions and with magnitudes such that the volume of the cell remains constant with time so that:

$$-\bar{v}_s J_s = \bar{v}_w J_w \quad (2.7.15)$$

By eliminating $\Delta\mu_s$ from (2.7.13) using (2.7.10) and (2.7.5) one can show that:

$$\omega = [\bar{c}_s/(1+\Theta)][L'_{11}L'_{22}-(L'_{12})^2] / [L'_{11}-\Theta L'_{12}] \quad (2.7.16)$$

Note that ω has typical units of mol/dyne-sec.

The final parameter, σ , is the solute reflection coefficient which is defined by the expression:

$$\sigma = \left[1 + \Delta\mu_w / (\bar{v}_w \bar{c}_s \Delta\mu_s) \right]_{J_v=0} \quad (2.7.17)$$

so that using (2.7.13), (2.7.10), and (2.7.5) one can show that:

$$\sigma = -L'_{12}(1+\Theta) / (L'_{11}-\Theta L'_{12}) \quad (2.7.15)$$

Substituting these new parameters into (2.7.12) and (2.7.13) gives:

$$J_w = [\bar{v}_w s]^{-1} \left[[L_p/\varepsilon + \Theta \bar{v}_s \omega] \Delta\mu_w / \bar{v}_w + [L_p(1-\sigma)/\varepsilon - \bar{v}_s \omega] \bar{c}_s \Delta\mu_s \right] \quad (2.7.19)$$

$$J_s = s^{-1} \left[[L_p(1-\sigma)/\varepsilon - \bar{v}_s \omega] \bar{c}_s \Delta\mu_w / \bar{v}_w + [\bar{c}_s L_p(1-\sigma)^2/\varepsilon + \omega] \bar{c}_s \Delta\mu_s \right] \quad (2.7.20)$$

where $s=1+\Theta-\sigma\Theta$. Papanek points out that the factor s is typically very close to one and hence can usually be considered to have very

little effect on the evaluation of the above expressions. Papanek proposes as a worst case an ethylene glycol solution with $\bar{c}_s=4M$, $\bar{v}_s=42.3\text{cm}^3/\text{mol}$, $\sigma=0.8$ so that for this highly concentrated solution $\varepsilon=1.03\approx 1.0$ so that one can, in general, assume $\varepsilon\approx 1.0$.

While this set of permeability equations is less restrictive than the Kedem and Katchalsky set one must deal with the problem of calculating the chemical potential of both the permeating solute and solvent inside and outside the cell. This adds to the computational complexity of the model. Since very little comparative data between the simpler K-K model and the Papanek model is available it is still unclear whether or not a comparable increase in accuracy is also achieved.

2.8 Nondimensionalization of the K-K Permeability Equations

For practical calculations it is desirable to work with a nondimensional (or partly nondimensional) set of equations. In this section the Kedem and Katchalsky equations for coupled binary flow represented by equations (2.3.30) and (2.3.31) will be nondimensionalized. The equations used for calculation purposes in the parameter estimation routines developed as a part of the present work are the partially nondimensional set represented by equations (2.8.14) and (2.8.15). The fully nondimensional set of equations are not used as one must absorb either the solute or solvent permeability into the nondimensional time factor to achieve full nondimensionalization. Since the permeabilities are unknowns in the parameter routine this makes the independent variable time an unknown factor as well. This unnecessarily complicates the input of data to the routine. Thus the set of equations used for practical calculations retain the dimensional time factor as the independent variable. Only the cell volume and cell solute content are nondimensional.

Begin by writing (2.3.30) and (2.3.31) in the form:

$$J_v = -L_p RT \left[(c_1^o - c_1^i) + \sigma (c_s^o - c_s^i) \right] \quad (2.8.1)$$

$$\dot{n}_s = \omega RT (c_s^o - c_s^i) + \bar{c}_s (1 - \sigma) J_v \quad (2.8.2)$$

where the superscript (o) implies outside the cell and (i) inside the cell. Recall that these equations apply to cases where there is no hydrostatic pressure difference, $\Delta P = 0$, and where the solute volume

fraction is small. The subscript (i) implies the sum of all impermeable solutes and (s) the permeable solute.

Recall that J_v is the volume flux and can be expressed as:

$$J_v = (1/A) dV/dt \quad (2.8.3)$$

and that $\dot{n}_s$ is the permeable solute flux given by:

$$\dot{n}_s = (1/A) dN/dt \quad (2.8.4)$$

where A is the surface area of the membrane, V is the cell volume, N is the number of moles of permeable solute inside the cell, and t is time. An alternate set of permeability coefficients will be used and are given by:

$$P_w = L_p RT \quad (2.8.5)$$

$$P_s = \omega RT$$

so that P_w will have typical units of (cm⁴/sec-mole) and P_s will have typical units of (cm/sec).

Using (2.8.3), (2.8.4), and (2.8.5) in (2.8.1) and (2.8.2) gives:

$$dV/dt = -P_w A [(c_i^0 - c_i^1) + \sigma(c_s^0 - c_s^1)] \quad (2.8.6)$$

$$dN/dt = P_s A (c_s^0 - c_s^1) + \bar{c}_s (1 - \sigma) dV/dt \quad (2.8.7)$$

If it is assumed that there is a relatively small difference in the concentration of the permeable solute across the membrane then the mean permeable solute concentration, $\bar{c}_s$, which is defined by equation (2.3.8), can be approximated by the arithmetic mean:

$$\bar{c}_s \approx (c_s^0 + c_s^i)/2 \quad (2.8.8)$$

Also notice that the internal concentrations can be expressed as functions of the cell volume. The internal permeable solute concentration is equal to the number of moles of permeable solute inside the cell, N , divided by the cell volume, V :

$$c_s^i = N/V \quad (2.8.9)$$

Similarly the impermeable solute concentration inside the cell can be expressed as:

$$c_i^i = c_{i0} V_0/V \quad (2.8.10)$$

where c_{i0} is the initial impermeable solute concentration and V_0 is the initial cell volume.

Incorporating this information into (2.8.6) and (2.8.7) gives:

$$dV/dt = -P_w A \left[(c_i^0 - c_{i0} V_0/V) + \sigma (c_s^0 - N/V) \right] \quad (2.8.11)$$

$$dN/dt = P_s A (c_s^0 - N/V) + (1/2) (c_s^0 + N/V) (1-\sigma) dV/dt \quad (2.8.12)$$

Defining the nondimensional volume and solute content as:

$$V^* = V/V_0 \quad N^* = N/c_s^0 V_0 \quad (2.8.13)$$

and using them in (2.8.11) and (2.8.12) gives:

$$dV^*/dt = -(P_w A/V_0) \left[(c_1^0 - c_{10}/V^*) + \sigma c_s^0 (1 - N^*/V^*) \right] \quad (2.8.14)$$

$$dN^*/dt = (P_s A/V_0) (1 - N^*/V^*) + (1/2) (1 + N^*/V^*) (1 - \sigma) dV^*/dt \quad (2.8.15)$$

While equations (2.8.14) and (2.8.15) are not fully nondimensionalized, as they still contain the time factor, they are a good set of equations to use for practical calculation. This is because the data available is typically the nondimensional volume as a function of real time a priori. In order to fully nondimensionalize the equation set one must absorb either P_s or P_w into the time factor. Since P_s and P_w are both typically unknowns this makes it impossible to calculate the nondimensional time as a function of the real time. This will make the procedure for estimating the permeability coefficients unnecessarily complicated.

If one wishes to work with a fully nondimensional set of equations then the groupings used by Johnson and Wilson, [8]:

$$\begin{aligned} \tau &= P_w A c_{10} t / V_0 \\ b &= P_s / P_w c_{10} \\ a &= \sigma c_s^0 / c_{10} \end{aligned} \quad (2.8.16)$$

can be used so that (2.8.14) and (2.8.15) become:

$$dV^*/d\tau = [(1/V^*) - (c_1^0/c_{10})] + \alpha[(N^*/V^*) - 1] \quad (2.8.17)$$

$$dN^*/d\tau = b[1 - (N^*/V^*)] + (1/2)[1 + (N^*/V^*)](1-\sigma) dV^*/d\tau \quad (2.8.18)$$

For the case of no impermeable solute present at the initial state ($c_{10}=0$) the groupings in (2.8.16) will not work. For this case one can use the alternate set of nondimensional groups:

$$\begin{aligned} \tau' &= P_w A c_s^0 t / V_0 \\ b' &= P_s / P_w c_s^0 \\ \alpha' &= c_1^0 / c_s^0 \end{aligned} \quad (2.8.19)$$

in which case the analogous nondimensional equations become:

$$dV^*/d\tau' = -\left[\alpha' - (c_{10}/c_s^0 V^*) + \sigma[1 - (N^*/V^*)]\right] \quad (2.8.20)$$

$$dN^*/d\tau' = b'[1 - (N^*/V^*)] + (1/2)[1 + (N^*/V^*)](1-\sigma) dV^*/d\tau' \quad (2.8.21)$$

No further complication occurs if no impermeable solute is introduced at time equal to zero. One can merely set $c_1^0=0$ in (2.8.20) and (2.8.21).

Many authors will simplify the evaluation of the above equations by assuming that the volume changes induced in a cell will be sufficiently small such that the surface area of the membrane, A , remains constant [10,12]. The surface area is then typically calculated using

the initial cell diameter. Stusnick, [10,11], explored the effects of this assumption on the resulting values calculated for the permeability coefficients using data reported by Stewart and Jacobs, [18]. Stusnick first calculated the permeability coefficients assuming that the surface area remained constant and then by allowing the surface area to be calculated as a function of cell volume using the expression:

$$A = (36\pi)^{1/3} V^{2/3} \quad (2.8.22)$$

While for the most part the change in the results was not significant the parameter values in at least one case varied by as much as 20%.

Papanek points out that for a typical case of a cell subjected to a rise in the extracellular concentration of the permeable solute while in the presence of impermeable solutes a shrink-swell behavior is observed: "The rate of rapid initial shrinkage will be determined almost wholly by the water permeability, L_p , as $L_p \gg \omega$ and there is initially very little solute transport. As solute begins to enter, and water continues to exit the cell, the interaction term σ becomes important. The shape of the volume minimum seen will be fixed primarily by σ . As solute slowly enters, the rate of swelling is almost entirely a function of solute permeability ω . During this phase, intracellular concentration is increasing and water re-enters the cell, but the change in cell volume is rate-limited by permeable solute entry", [12].

One would expect these trends to express themselves in reverse in a parameter estimation routine. For instance the value of L_p will be determined almost wholly by the rate of initial shrinkage. In the case of the shrink-swell behavior the assumption of surface area remaining constant at its initial value will consistently over-predict the actual surface area. This would suggest that permeability would be correspondingly under-predicted in order to keep the volume flux the same for an over-predicted area, particularly during the initial period of rapid shrinkage. This effect does appear when comparing the values of L_p in Stusnick's work. The effect does not, however, express itself in the values of ω whose value was changed in alternate directions under the new evaluation depending on the specific case. It is also interesting to note that the value of σ was consistently over-predicted when using the assumption of constant area as compared to the values when the area was calculated as a function of volume. One would suspect that this might be due to compensation for the under-predicted values of L_p .

With the availability of fast, efficient computers the extra evaluations required for calculating the surface area of the membrane as a function of volume will not significantly increase the execution time and hence should, in general, be included in order to prevent biasing of the calculated parameter values.

CHAPTER 3

Parameter Estimation Methods

3.1 Parameter Estimation Overview

When one thinks about solving an equation they inherently think in terms of solving for the state of a system given certain parameters and initial and/or boundary conditions. The parameters are given no special consideration but are merely looked up in a hand book. in many cases, however, the parameters are unknown or perhaps unknown functions of the state variables. It is this problem which is addressed by the methods of parameter estimation.

Parameter estimation is the science of determining from measured data the values of the physical properties of materials and systems or arbitrarily chosen parameters which play the role of constants in an equation or mathematical model. The determination of physical properties is typically performed through measurements of the state variables (temperature, pressure, etc.). These measurements are matched to a mathematical model of the process being monitored in which the parameters of interest appear either explicitly or implicitly. In the case of arbitrary constants one may be trying to fit a physical property of a system to an empirical or analytical model. This model may be a function of one or more of the state variables.

Typically one would be working from measured values of the property of interest at specific states of the system. Many other applications for parameter estimation methods have been proposed including the separate yet related science called function estimation by Beck, [15].

It is not the purpose here to enumerate the applications of parameter estimation but rather to present for the student new to estimation theory a simple explanation of some of the more common methods. This presentation is particularly slanted towards the problem of parameter estimation in membrane transport. The methods applicable to this problem are not the most powerful techniques available. Because of the nature of the data being processed very little information of a statistical nature is available. For instance, since the data available using the techniques described in this work are typically the volume-time history of an individual cell in a single experiment no knowledge of the variance-covariance properties of the data is available. Given this type of information many more powerful techniques could be employed. One of the simpler of the methods which utilize statistical information, maximum likelihood (ML) estimation, will be presented in order to illustrate one of the ways in which the investigators knowledge can be incorporated into these methods.

The first methods to be presented are the ordinary least squares (OLS) method and the Gauss method of minimization for nonlinear estimation. These are perhaps the oldest and most basic of the parameter estimation methods. Together they form the basis of a multitude of other techniques. Some of these modified least squares methods will

also be presented including the Box-Kanemasu method and Marquardt's method.

A FORTRAN computer program which is based on the OLS estimator and the Gauss minimization method has been coded as a part of the present work. This program also incorporates the modifications of Marquardt and Box-Kanemasu. This program is call MARBOX and is described in detail in Appendix A of the present work. This program has been particularly tailored to the passive membrane transport problem which has at most three parameters.

3.2 Ordinary Least Squares and Maximum Likelihood Estimators

The Ordinary Least Squares (OLS) method is perhaps the oldest and simplest of the parameter estimation routines. This method, like all other estimation methods, attempts to minimize the error between the function values predicted by a mathematical model and the known or measured functional values with respect to the floating parameters. The various routines will differ in the way in which they measure the fit between the data and the model and how they choose the direction and size of the step changes taken to determine the "true" parameter values from initial estimates. It is this method which has been incorporated into the parameter estimation routines developed as a part of the present work.

In the OLS method the fit between the data and the model is measured by the sum of the square of the difference between the measured values and the corresponding values predicted by the model. This is referred to as the sum of the squares function and for OLS is given by:

$$S = \sum_{i=1}^n (Y_i - \eta_i(\bar{\beta}))^2 \quad (3.2.1)$$

where n is the number of data points, the Y_i 's are the measured data values, and the η_i 's are the values predicted by the model. $\bar{\beta}$ represents the vector of parameter values so that the notation $\eta_i(\bar{\beta})$ expresses the dependency of the predicted values on the parameter values.

The problem then becomes that of minimizing the functional S with respect to the parameter values. Mathematically this is expressed through the derivatives of the functional S with respect to the vector of parameters by meeting the condition:

$$\partial S / \partial \bar{\beta}^* = 0 \quad (3.2.2)$$

where $\bar{\beta}^*$ is the vector of "best fit" parameter values. When the values of $\bar{\beta}$ are found that satisfy equation (3.2.2) then the sum of the squares function has been minimized with respect to the parameters. Since one can not generally satisfy (3.2.2) exactly one typically will continue iteration on the parameter values until the change in parameter values becomes insignificant. It is possible in some cases to have more than one set of values satisfy this condition due to the presence of local minimums. One would wish to find the values corresponding to the global minimum. This generally is not a problem as most well-posed models will possess only one minimum value.

Under Maximum Likelihood (ML) estimation a slightly modified sum of the squares function is utilized. Each data point is weighted by its variance so that S becomes:

$$S_{ML} = \sum_{i=1}^n [(Y_i - \eta_i(\bar{\beta})) / \sigma_i]^2 \quad (3.2.3)$$

where σ_i here represents the variance of data point (i). Thus those data points with a lower variance will be weighted more than those with higher variances. This type of method is not utilized in the

present work as the variance of the data points used in the present work are unknown. Many other methods which utilize the investigator's knowledge are available.

3.3 The Gauss Minimization Method

The Gauss or Gauss-Newton method for minimizing the sum of the squares function is one of the simplest minimization methods. While it was proposed many decades ago it is only with the advent of the computer that it has become practical to apply due to the large number of calculations required. This method has proven to be effective for well-defined problems which have a distinct minimum.

Recall that we are attempting to find values of the vector $\bar{\beta}$ such that an equation of the form (3.2.2) is satisfied. By differentiating (3.2.1) one can show that for the OLS method:

$$\partial S / \partial \bar{\beta} = \partial / \partial \bar{\beta} \left[\sum_{i=1}^n (Y_i - \eta_i(\bar{\beta}))^2 \right] \quad (3.3.1)$$

$$\partial S / \partial \bar{\beta} = \sum_{i=1}^n -2(Y_i - \eta_i(\bar{\beta})) \partial \eta_i / \partial \bar{\beta} \quad (3.3.2)$$

Because $\bar{\beta}$ is a vector, satisfying (3.2.2) implies simultaneously satisfying the set of p equations:

$$\begin{aligned} \partial S / \partial \beta_1^* &= 0 \\ \partial S / \partial \beta_2^* &= 0 \\ &\cdot \\ &\cdot \\ \partial S / \partial \beta_p^* &= 0 \end{aligned} \quad (3.3.3)$$

where p is the number of floating parameters.

A sensitivity matrix $\bar{X}$ is defined with the components:

$$\bar{X}_{ij} = \partial \eta_i / \partial \beta_j \quad (3.3.4)$$

so that $\bar{X}$ has dimensions of $n \times p$. Further using the matrix notation on $\bar{Y}$ and $\bar{\eta}$:

$$\bar{Y} = [Y_1, Y_2, \dots, Y_n]^T \quad (3.3.5)$$

$$\bar{\eta} = [\eta_1, \eta_2, \dots, \eta_n]^T \quad (3.3.6)$$

equation (3.3.2) can be rewritten:

$$\partial S / \partial \bar{\beta} = -2 \bar{X}^T (\bar{Y} - \bar{\eta}(\bar{\beta})) \quad (3.3.8)$$

Thus the desired solution, $\bar{\beta}^*$, will satisfy the condition:

$$\bar{X}^T (\bar{Y} - \bar{\eta}(\bar{\beta}^*)) = 0 \quad (3.3.9)$$

The problem is now to solve for the value of $\bar{\beta}^*$ in this expression. This is not easy, however, as both $\bar{X}$ and $\bar{\eta}$ are dependent on the value of $\bar{\beta}$.

In the Gauss method the first two terms in a Taylor series expansion of $\bar{\eta}$ about the estimated values of $\bar{\beta}^*$, designated $\bar{b}$, are used to generate a better estimate of $\bar{\beta}^*$. This method requires that $\bar{\eta}$ have continuous first derivatives in $\bar{\beta}$ and bounded higher order derivatives. The Taylor series is given by:

$$\bar{\eta}(\bar{\beta}^*) = \bar{\eta}(\bar{b}) + (\partial \bar{\eta}(\bar{b}) / \partial \bar{b}) (\bar{\beta}^* - \bar{b}) + \text{H.O.T.} \quad (3.3.10)$$

or using the sensitivity matrix and ignoring the higher order terms (H.O.T.):

$$\bar{\eta}(\bar{\beta}^*) \approx \bar{\eta}(\bar{b}) + \bar{X}(\bar{b}) (\bar{\beta}^* - \bar{b}) \quad (3.3.11)$$

Substituting this expression into (3.3.9):

$$\bar{X}^T [\bar{Y} - \bar{\eta}(\bar{b}) - \bar{X}(\bar{b}) (\bar{\beta}^* - \bar{b})] \approx \bar{0} \quad (3.3.12)$$

This expression can now be solved for the approximate value of $\bar{\beta}^*$ giving:

$$\bar{\beta}^* \approx \bar{b} + [\bar{X}^T \bar{X}]^{-1} \bar{X}^T (\bar{Y} - \bar{\eta}(\bar{b})) \quad (3.3.13)$$

For models which are linear in the parameters, that is for models in which the parameters appear only to the first power, equations (3.3.11) through (3.3.13) will be true equalities and only a single iteration will be required to reach the final values. However for models nonlinear in the parameters, several iterations may be required depending on the nature of the model and the accuracy of the initial parameter estimates.

The iteration procedure can be summarized as follows. Defining the matrix $\bar{P}$ as:

$$P = [\bar{X}^T \bar{X}]^{-1} \quad (3.3.14)$$

and using the superscript (k) to indicate the current values and (k+1) to indicate the future values then one iterates on the following two equations:

$$P^{(k)} = [\bar{X}^{T(k)} \bar{X}^{(k)}]^{-1} \quad (3.3.15)$$

$$\bar{b}^{(k+1)} = \bar{b}^{(k)} + P^{(k)} [\bar{X}^{T(k)} (\bar{Y} - \bar{\eta}^{(k)})] \quad (3.3.16)$$

where $\bar{\eta}^{(k)}$ is a function of $\bar{b}^{(k)}$. The iteration continues until there is a negligible change in the parameter values. One way of testing for convergence is to set a tolerance factor, δ , such that execution ends when:

$$\frac{ABS[b_i^{(k+1)} - b_i^{(k)}]}{ABS[b_i^{(k)}] + \delta_1} < \delta \quad (i=1,2,\dots,p) \quad (3.3.17)$$

where δ_1 is a very small number ($\delta_1 \ll \delta$) which will prevent a divide by zero condition in the event that one or more of the parameter values goes to zero during an iteration. Typically the value of δ_1 will be about 10^{-16} but this will depend on the computer being used.

3.4 Box-Kanemasu Interpolation Method

The Box-Kanemasu method is a minimization method based upon a modification of the Gauss method. In some cases this method can prevent oscillations of increasing amplitude in the parameter values caused by the linearization of a nonlinear model using the Gauss method. Included in this presentation is a modification to the Box-Kanemasu method used by Bard, [17], which insures a decrease in the sum of the squares function for each iteration.

The basic strategy in this method is to modify the step size calculated by the Gauss method in order to insure a better fit with the new parameter values. The direction of the step is not changed under this method.

Begin by decomposing equation (3.3.16) into two equations:

$$\Delta \bar{b}^{(k)} = \bar{P}^{(k)} \left[\bar{X}^{T(k)} (\bar{Y} - \bar{\eta}^{(k)}) \right] \quad (3.4.1)$$

$$\bar{b}^{(k+1)} = \bar{b}^{(k)} + h^{(k+1)} \Delta \bar{b}^{(k)} \quad (3.4.2)$$

where h is a scalar quantity whose magnitude can vary with each iteration. If $h=1$ for all values of (k) then one has the Gauss method. Equation (3.4.2) is typical of many of the modified least squares methods. These methods will differ in the way in which the value of h is calculated. Some methods will perform a search on h to determine the value which gives the minimum value for the sum of the squares function.

In the Box-Kanemasu method the value of h which will minimize the sum of the squares function is approximated through a second order quadratic approximation for S in h :

$$S \simeq a_0 + a_1 h + a_2 h^2 \quad (3.4.3)$$

where a_0 , a_1 , and a_2 are arbitrary constants to be calculated for each iteration. A corresponding linear approximation of the parameter vector $\bar{\beta}$ will also be used:

$$\bar{\beta} \simeq \bar{b} + h \Delta \bar{b} \quad (3.4.4)$$

To calculate the values of the three arbitrary constants in (3.4.3) three pieces of information are required. The first is to calculate S at $h=0$ so that:

$$S(h=0) = a_0 = S_0^{(k)} \quad (3.4.5)$$

$$\bar{\beta}(h=0) = \bar{b}^{(k)} \quad (3.4.6)$$

$$a_0 = S_0^{(k)} = S(\bar{b}^{(k)}) \quad (3.4.7)$$

A second value is determined at $h=\alpha$ where α is initially set to 1.0. This value of S is denoted $S_\alpha^{(k)}$. The presence of α in this development represents the modification used by Bard. If $\alpha=1$ then the following development would match that of Box-Kanemasu. In the modified version the value of α is manipulated to insure a reduction in the sum of the squares function.

The final piece of information is gained from the derivative of S with respect to h in the $\Delta \bar{b}^{(k)}$ direction at h=0:

$$\left[dS/dh \right]_{h=0} = a_1 = \sum_{i=1}^p \left[\partial S / \partial \beta_i \right]_{h=0} \left[\partial \beta_i / \partial h \right]_{h=0} \quad (3.4.8)$$

or:
$$a_1 = \left[(\partial S / \partial \bar{\beta})^T (d\bar{\beta} / dh) \right]_{h=0} \quad (3.4.9)$$

From equation (3.3.8) for the ordinary least squares sum of the squares function one can show that:

$$\partial S / \partial \bar{\beta} = -2 \bar{X}^T (\bar{Y} - \bar{\eta}) \quad (3.4.10)$$

and from (3.4.4) one can show that:

$$d\bar{\beta} / dh = \Delta \bar{b}^{(k)} \quad (3.4.11)$$

so that:
$$a_1 = \left[-2 \bar{X}^T (\bar{Y} - \bar{\eta}) \right]^T \Delta \bar{b}^{(k)} \quad (3.4.12)$$

The scalar quantity $G^{(k)}$ is defined as:

$$G^{(k)} = \left[\bar{X}^T (\bar{Y} - \bar{\eta}) \right]^T \Delta \bar{b}^{(k)} \quad (3.4.13)$$

so that:
$$a_1 = -2G^{(k)} \quad (3.4.14)$$

Using the definition of the matrix $\bar{P}$ given by equation (3.3.14) it is possible to show that:

$$G^{(k)} = [\Delta \bar{b}^{(k)}]^T \bar{P}^{-1} \Delta \bar{b}^{(k)} \quad (3.4.15)$$

It can also be shown that:

$$a_2 = [S_a^{(k)} - S_0^{(k)} + 2G^{(k)}a]/a^2 \quad (3.4.16)$$

The minimum of S with respect to h can be found by setting the derivative of S with respect to h equal to zero and solving the resulting expression for h . This will yield the expression:

$$h^{(k+1)} = G^{(k)} a^2 / [S_a^{(k)} - S_0^{(k)} + 2G^{(k)}a] \quad (3.4.17)$$

Bard suggests using the variable α to insure that:

$$S_a^{(k)} < S_0^{(k)} \quad (3.4.18)$$

If this condition is not met by the initial value of $\alpha=1$ then α is divided by two and S_a is recalculated. This process continues until a value of α is found such that the condition of (3.4.18) is met. If the value of α must be reduced to about 0.01 and (3.4.18) is still not satisfied then it is suggested that execution be discontinued. Beck, [16], suggests that this condition may indicate improper programming, especially in the calculation of the sensitivity coefficients, or near

linear dependence of the sensitivity coefficients.

The use of equation (3.4.17) in the calculation of h is restricted by the value of a_2 . Beck, [16], explores the implications of all three possibilities on the value of a_2 , that is, $a_2 < 0$, $a_2 = 0$, and $a_2 > 0$. The results show that equation (3.4.17) should be used only if $a_2 > 0$. This can be tested by requiring that:

$$S_a^{(k)} \geq S_0^{(k)} - (2-A^{-1})aG^{(k)} \quad (3.4.19)$$

where A is a value equal to or slightly greater than one. Beck suggests using $A=1.1$. If the inequality represented by (3.4.19) is not satisfied while (3.4.18) is satisfied then Beck suggests that:

$$h^{(k+1)} = Aa \quad (3.4.20)$$

be used. If (3.4.19) is satisfied then $h^{(k+1)}$ should be set to the value calculated by (3.4.17) or by (3.4.20) whichever is the smaller.

3.5 Marquardt's Method

Marquardt's method of minimization utilizes a compromise between two different methods. One method is called the method of steepest descent. In this method the step direction is taken in the direction opposite the gradient. Thus the direction chosen for the parameter step is that of the steepest downward slope in the sum of the squares function with respect to the parameter values. Mathematically this is expressed by:

$$\Delta b_j \propto -\partial S / \partial b_j = 2 \sum_{i=1}^n (Y_i - \eta_i) \partial \eta_i / \partial b_j \quad (3.5.1)$$

This method typically works well when the parameter estimates are far from the minimum but is slow to converge near the minimum as the value of $\partial S / \partial b_j$ approaches zero and the step size decreases.

The second method is the Gauss method of minimization which utilizes a linearization of the fitting function to calculate the step size. This method is outlined in section 3.2 of this work. The Gauss method works well for parameter estimates close to the minimum but poorly when the estimates are far removed and the model is non-linear.

In Marquardt's method a compromise is made between the two methods. Initially the method resembles that of the steepest descent and as the parameter estimates improve it shifts toward the Gauss method. This strategy is implemented by weighting of the $\mathbf{X}^T \mathbf{X}$ matrix, which is the matrix of sensitivity coefficients.

The matrix $\bar{A}$ is defined as:

$$\bar{A} = \bar{X}^T \bar{X} \quad (3.5.2)$$

so that $\bar{A}$ is a prp symmetric matrix. Also note that from equation (3.3.14):

$$\bar{P} = \bar{A}^{-1} \quad (3.5.3)$$

The matrix $\bar{A}$ is modified using the expressions:

$$A_{ij} = A_{ij}(1+\lambda) \quad (i=j) \quad (3.5.4)$$

$$A_{ij} = A_{ij} \quad (i \neq j)$$

so that if $\lambda=0$ the matrix is unchanged. For large values of λ this procedure weights the $\bar{A}$ matrix towards its main diagonal driving the routine towards the steepest descent method. As $\lambda \rightarrow 0$ the method resembles the Gauss method.

In application the initial value of λ is made relatively large (say about 1000). The worse the initial parameter estimates and the more ill-behaved the model the larger the initial value of λ . A new set of parameter values and the sum of the squares function for these new values are calculated based on the initial parameter estimates. If the sum of the squares function has been reduced by the new parameter values then λ is reduced by a certain factor (anywhere from 2 to

about 10) and iteration proceeds. If the sum of the squares function is not reduced then λ is increased by the same factor and a new set of parameter values is calculated using this new value of λ . This continues until a reduction in the sum of the squares is realized. The routine then proceeds to the next iteration. New parameter values are calculated based on the updated estimates and iteration continues until a standard convergence criterion is met.

CHAPTER 4

Data Gathering Techniques

4.1 The Diffusion Chamber

The diffusion chamber is a microscope stage capable of subjecting an isolated sample of cells to a psuedo-step change in extra-cellular concentration of solutes. These solutes can be either permeable or impermeable to the cell membrane. Figure 4.1.1 shows a cross section of the chamber and Figure 4.1.2 shows an exploded view of the chamber parts.

The sample of cells to be tested is held in a region bounded by glass on top, a dialysis membrane on the bottom, and a rigid support ring around the side. The thickness of this region is manipulated by a spacing ring which is thick enough so that the cells are not compressed between the membrane and the glass and yet not so large as to allow the cells to drift in and out of focus. Thus the size of the cells of interest determines the thickness of the spacing ring and hence the sample region.

Below the sample region on the other side of the dialysis membrane is a second region containing a bulk flow of the extracellular solution. This region has a volume much greater than that of the sam-

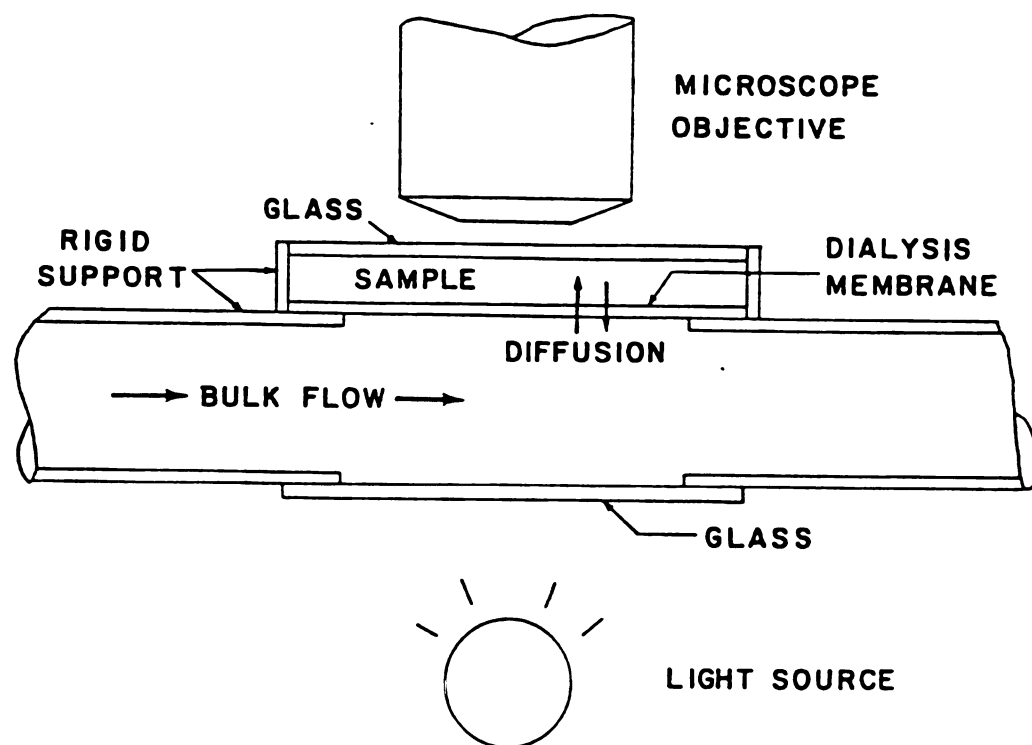


Figure 4.1.1- Cross-section of the diffusion chamber.

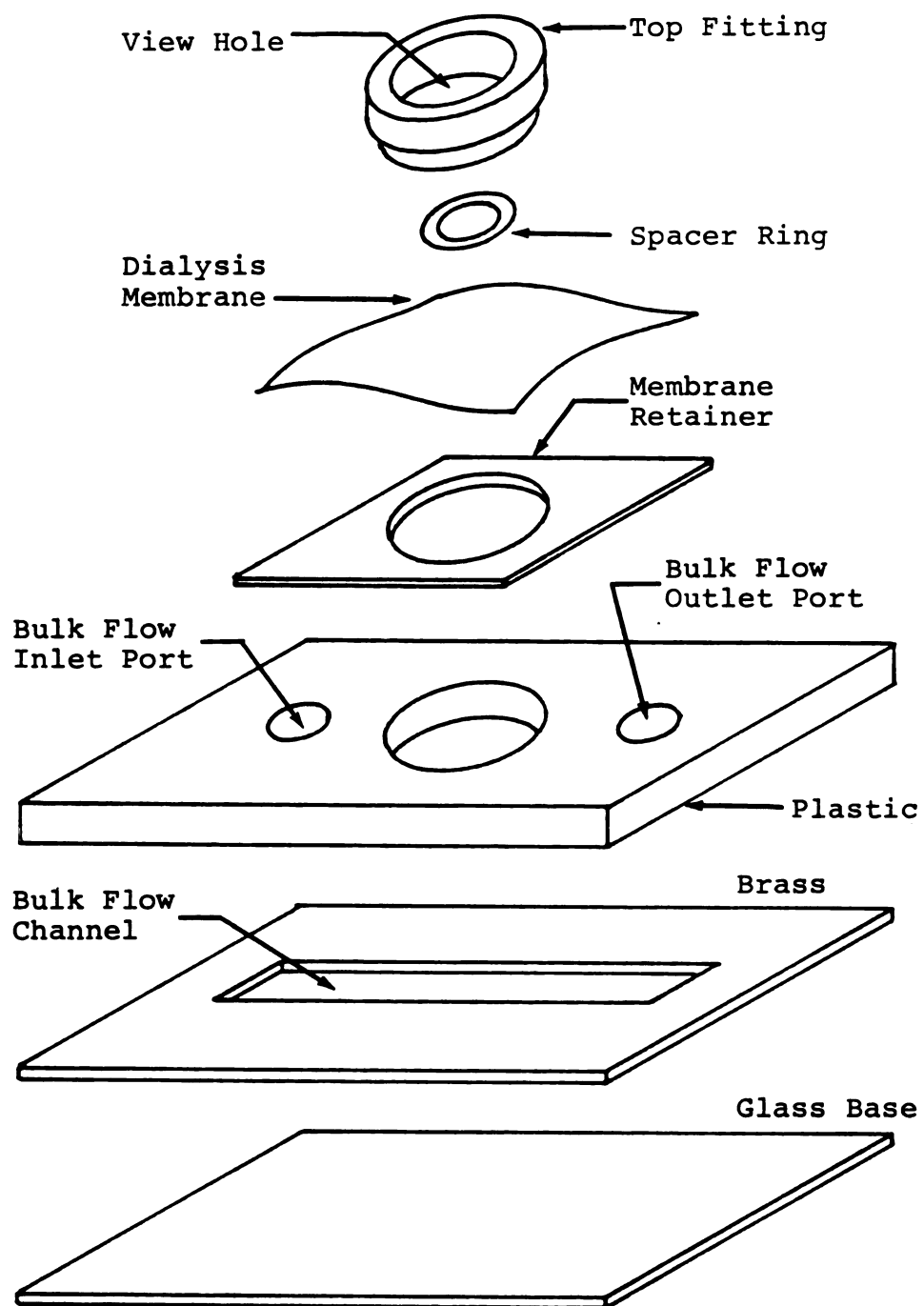


Figure 4.1.2- Exploded view of the diffusion chamber.

ple region. Initially the bulk flow region will contain the same solution as the sample region. The chamber and the cells are allowed to come to equilibrium with this solution. At time equal to zero the bulk flow region is flushed with a new solution whose solute concentration differs from that of the initial solution. If the new concentration is higher than that of the initial solution then solutes will diffuse upward across the dialysis membrane from the bulk flow region to the sample region. If the new solution has a lower solute concentration then the solute flow will be reversed.

The bulk flow can be regulated through a pair of pressurized bottles, one containing the initial solution and the other containing the new solution. The bottles are connected with plastic tubing to a pair of small pumps with a single outlet. This outlet is connected to the bulk flow inlet port on the diffusion chamber. A plastic tube is connected to the bulk flow outlet port and inserted into a discharge vessel. One switch is provided to switch the pump intake from one bottle to the other and a second switch turns the pumps on and off. This mechanism was developed by Ligon and is detailed in an unpublished work. This technique works well for very large cells which do not have a tendency to move about.

For smaller cells the pressurized bottles created large scale disturbance due to the sudden introduction of flow in the bulk region which could cause the cells to drift out of view. For cases where this is a problem an alternate injection method has been developed. Rather than using the pressurized bottles a large (35cc) syringe is

used to introduce the bulk flow. The chamber is set up initially with the equilibrium solution in the bulk flow channel. The syringe is filled with the new solution and a plastic tube attached to the end. Air is then removed from both the syringe body and the tube. The other end of the tube is then connected to the bulk flow inlet port of the diffusion chamber. At time equal to zero the syringe is manually depressed introducing a bulk flow of the new solution into the bulk flow region. This system allows the user to manipulate the velocity of the bulk flow in order to prevent large scale disturbances in the system. When using a syringe with smooth plunger action this technique is easily implemented.

Once the fluid in the bulk flow region has been replaced by the new solution the change in solute concentration will be gradually introduced into the sample region. The cells in the sample region will begin to respond to the non-equilibrium osmotic conditions created. The response of the cells will depend explicitly on the extracellular concentration of the sample region as expressed in equations (2.3.30) and (2.3.31). Thus in order to estimate the value of the permeability parameters L_p , σ , and ω one must have explicit knowledge of the $c_s^0(t)$ and $c_i^0(t)$, the extracellular concentrations of permeable and impermeable (relative to the cells membrane) solutes. These extracellular concentrations correspond to the concentrations in the sample region so that the extracellular concentration will depend on the rate at which the new solution in the bulk flow region manifests itself in the sample region. This will in turn depend on the permeability of the dialysis membrane to each of the solutes present.

In Ligon's work it was shown that for most solutes the sample region will have very small concentration gradients and hence can be characterized by a single concentration. In the following a model is developed which allows for the prediction of $c_s(t)$ and $c_i(t)$ in the sample region based on the solute permeability characteristics of the dialysis membrane used in the diffusion chamber.

The diffusion chamber itself can be modeled using the Kedem and Katchalsky definition of solute permeability. The defining relationship for solute permeability, ω , is given by equation (2.3.24):

$$J_s = [\omega \bar{c}_s \Delta \mu_s]_{J_v=0} \quad (4.1.1)$$

where J_s is the mole flux, ω is the solute permeability, and $\bar{c}_s$ is the mean transmembrane solute concentration defined by equation (2.3.8). $\Delta \mu_s$ is the transmembrane difference in chemical potential in this case given by $(\mu_{sb} - \mu_{se})$ where (b) implies the bulk solution and (e) the extracellular or sample solution. It will be assumed that the volume of the sample region remains constant and $J_v=0$. In experiments run at high magnification using the chamber the dialysis membrane does not appear to deform to accommodate volume changes as the cells will generally remain in focus throughout the experiment. This would tend to support the validity of the constant volume assumption. This means that for each solute present one can write that:

$$J_i = \omega_i \bar{c}_i \Delta \mu_i \quad (4.1.2)$$

Using the expression, [9]: $\Delta\mu_1 = \bar{v}_1\Delta P + RT\Delta c_1/\bar{c}_1$ (4.1.3)

and assuming no transmembrane pressure difference ($\Delta P=0$) then:

$$J_1 = \omega_1 RT\Delta c_1 \quad (4.1.4)$$

Again since the dialysis membrane does not appear to deform during an experiment, and since the fluids are incompressible the assumption of no hydrostatic pressure difference appears justified as no other mechanism exists by which a pressure difference could be maintained. Defining N as the number of moles of a given solute in the sample region and dropping the subscript (i) one can write:

$$dN/dt = JA = \omega RTA\Delta c \quad (4.1.5)$$

where A is the effective transfer area determined by the spacing ring. One can then use $\Delta c = c_b - c_s$ and $c_s = N/V$ where V is the constant volume of the sample region. Note that if the cells make up a significant portion of the sample region then V can be replaced by $(V - V_c)$ where V_c is the total volume of cells. If the overall change in cell volume is assumed to be small then V can be assumed to be constant and no further change in the following development is needed. Equation (4.1.5) then becomes:

$$dN/dt = \omega RTA [c_b - (N/V)] \quad (4.1.6)$$

or:
$$dN/dt + (\omega RTA/V)N = \omega RTA c_b \quad (4.1.7)$$

Assuming that c_b is not a function of time (i.e. that it undergoes a

step change at time equal to zero) then this is a linear first order ordinary differential equation in N and t . It is subject to the initial condition:

$$N(0) = N_0 = c_0 V \quad (4.1.8)$$

This differential equation can be solved by two linearly independent solutions which together satisfy the initial condition and the differential equation. One solution is generated by considering the homogeneous equation:

$$dN_h/dt + (\omega RTA/V)N_h = 0 \quad (4.1.9)$$

This equation is satisfied by the expression:

$$N_h = \beta \exp[-(\omega RTA/V)t] \quad (4.1.10)$$

where β is an arbitrary constant. The second solution comes from the non-homogeneous equation (4.1.7) and is given by:

$$N_n = c_b V \quad (4.1.11)$$

Reconstructing the full solution:

$$N = N_h + N_n = \beta \exp[-(\omega RTA/V)t] + Vc_b \quad (4.1.12)$$

Applying the initial condition:

$$N_0 = \beta + Vc_b \quad (4.1.13)$$

so that:
$$\beta = N_0 - Vc_b \quad (4.1.14)$$

This makes the final solution:

$$N(t) = Vc_b + (N_0 - Vc_b) \exp[-(\omega RTA/V)t] \quad (4.1.15)$$

Dividing this expression through by V and letting $c(t)=N(t)/V$ then:

$$c(t) = c_b + (c_0 - c_b) \exp[-(\omega RTA/V)t] \quad (4.1.16)$$

Thus the concentration of each solute in the sample region is expressed as an exponential function of the concentration of the bulk solution, the initial concentration of the sample region, the permeability of the dialysis membrane to that solute, the volume of the sample region, the effective area of transfer, and time.

The most difficult of these factors to obtain will be the permeability of the dialysis membrane. Data has been obtained for the permeability of Cuprophane dialysis membrane produced by Enka Glanzstoff AG. These data imply a linear relationship between the log of the permeability of the dialysis membrane for a given species and the log of the molecular weight of that species. This dependency is shown in Figure (4.1.3) for three types of the Cuprophane membrane.

It is interesting to note that the time constant for equation (4.1.16) is given by the expression:

$$t_c = V/A\omega RT \quad (4.1.17)$$

Consider as one example the typical configuration used to generate the data reported in the present work. In that configuration the quantity V/A was given as 2.286×10^{-2} cm. The Cuprophane 80pm membrane was used so that for sodium chloride $\omega RT = 1.166 \times 10^{-3}$ cm/sec. This yields a time constant of 19.6 sec. Thus it will take 58.8 seconds (three time constants) to reach 95% response in the sample region when using sodium chloride. This is a considerable length of time for experiments which typically last 4-6 minutes.

Consider as a second example the same configuration of the diffusion chamber and the same dialysis membrane but with sucrose as the solute of interest. The molecular weight of sucrose is 342.3 grams so that the permeability of the 80pm dialysis membrane to sucrose is approximately 3.3×10^{-4} and the time constant becomes $t_c = 69.3$ sec. This means that it will take nearly 4 minutes to achieve 95% response.

From these simple calculations it is clear that the presence of the dialysis membrane in the diffusion chamber will introduce a significant delay in the time in which the sample region comes to equilibrium with the new solution in the bulk flow region. One would expect this delay to have a significant effect on the values of the permeability parameters calculated in a parameter estimation routine

based on data gathered using the diffusion chamber system.

4.2 Characterization of the Dialysis Membrane Permeability

A technique developed by Ligon, [19], and modified as a part of the present work has been used to characterize the permeability of the dialysis membrane to various solutes. In the present study the Cuprophane 100pm flat membrane was tested using the solute glycerol. Other studies performed in the BTP lab by Dupuis have utilized other solutes including sodium chloride and sucrose, [25]. The values for the permeability of the dialysis membrane generated as a part of the present work were used in the processing of data on the binary flow response of hamster embryos. The results of this process are documented in Chapter 5, Section 2 of the present work. The method used to characterize the dialysis membrane permeability is presented here in order to document the procedure used for future reference. The characterization of the dialysis membrane permeability to various solutes will be an ongoing research project in the BTP lab. In processing data obtained using the diffusion chamber system the dialysis membrane permeability value used in the transport model will have a great effect on the resulting cell membrane permeabilities calculated as shown in Chapter 5 of the present work. Thus it is important that the dialysis membrane permeability to each of the solutes of interest be known accurately in order to obtain reliable results using the diffusion chamber system.

This technique utilizes two well-stirred chambers separated by a piece of the dialysis membrane (see Figure 4.2.1). One chamber contains 400ml of a relatively low concentration solution (typically

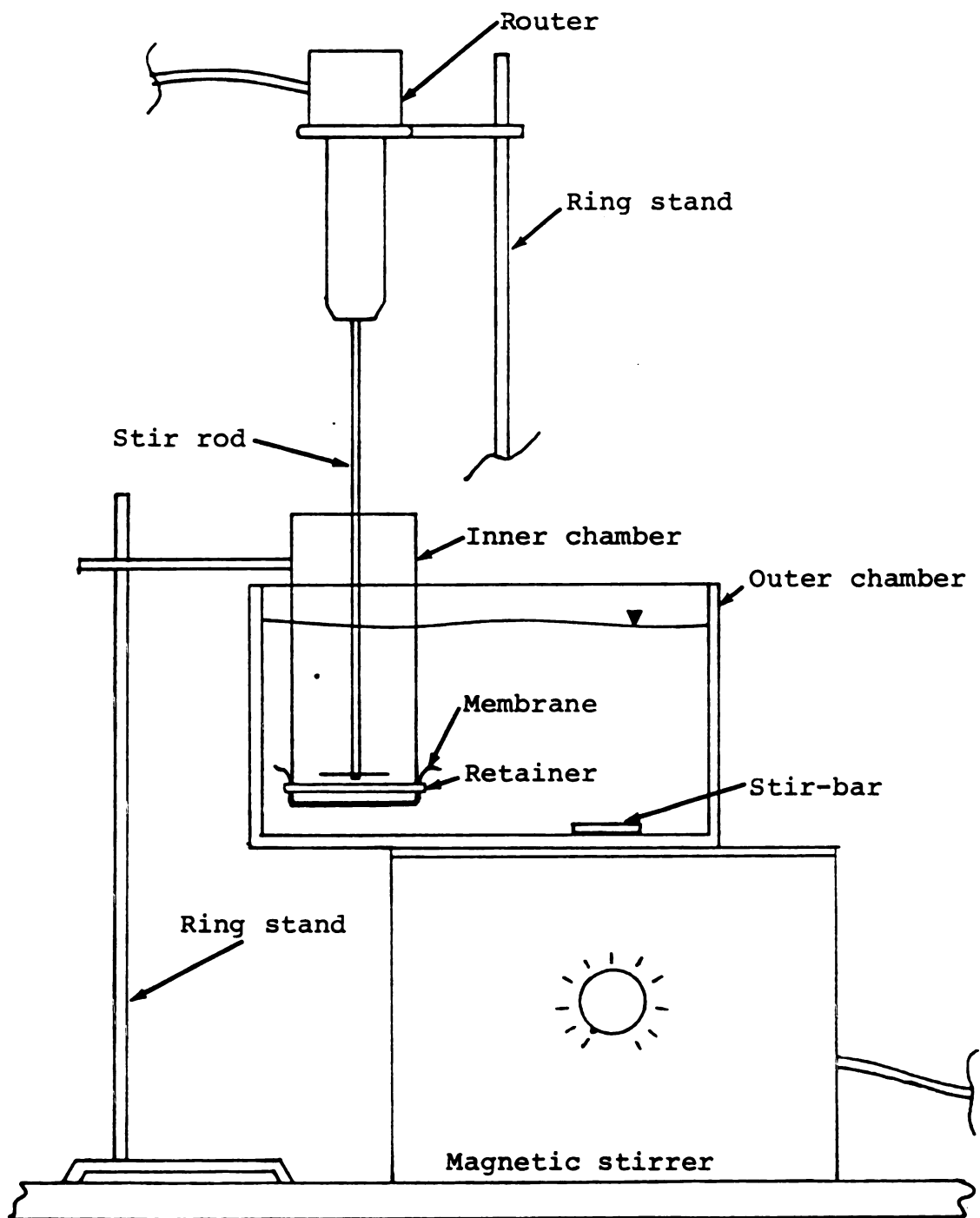


Figure 4.2.1- Experimental set-up for determination of dialysis membrane permeability.

0.1-0.2 moles/liter) of glycerol. The other chamber holds 1600ml of distilled water.

The larger outer chamber used in the present work was a rectangular (9x4x6 inches) chamber made from clear plexiglass and sealed with silicon caulk. This chamber was set off-center on a magnetic stirrer platform so that the stir-bar was positioned towards one end of the chamber. The smaller inner chamber, which holds the higher concentration solution, was made from a section of 3 inch inside diameter PVC drainage pipe approximately 8 inches long. A clear plastic window was cut into the side of this chamber in order to facilitate observation of the inner solution during an experiment. The lower end of the inner chamber was milled to accommodate a membrane retaining ring also made from plastic PVC pipe.

A piece of the dialysis membrane approximately 4 inches square was soaked in distilled water for 20-30 minutes prior to use in an experiment. At the beginning of an experiment the membrane was stretched over the lower end of the inner chamber and the membrane retaining ring pressed over the membrane, holding it securely in place. The inner chamber was then suspended within the outer chamber approximately 1 inch above the bottom of the outer chamber using a ringstand and clamp. The actual level at which the inner chamber was suspended was predetermined such that when the inner chamber held 400ml and the outer chamber held 1600ml the fluid levels in each chamber would be the same. This insured that no hydrostatic pressure difference would be created due to unequal levels in the chambers.

The presence of the window in the inner chamber made it possible to check the levels visually as well.

The inner chamber was positioned at the end of the outer chamber opposite the end with the magnetic stir-bar. In the original set-up a large circular beaker was used for the outer chamber so that the stir-bar was positioned directly below the center of the inner chamber. It was found that using a set-up such as this would induce a hydrostatic pressure difference across the membrane due to the vortexing action created by the spinning of the stir-bar. This resulted in a pressure-driven flow leaving the inner chamber. This could be seen in that the membrane would bow outward and the level in the inner chamber would drop significantly during an experiment. By moving the stir-bar out from under the inner chamber this problem was avoided.

With both chambers in place distilled water was added to the outer chamber until the level just reached the level of the membrane. At this time a stop-watch was started and the remainder of the 1600ml of distilled water and the 400ml of the glycerol solution were quickly but carefully added to their respective chambers. The two solutions were added simultaneously at rates such that the fluid levels in the two chambers remained equal. Stirring in the outer chamber was then initiated. A Craftsman router with a plastic stir rod powered through a Powerstat variable autotransformer was suspended over the inner chamber using a second ringstand. The height of the router was set so that the cross bar at the lower end of the stir rod was about 1/2 inch above the membrane. Enough power was then applied to the router to

provide slow but thorough mixing of the inner solution.

A small sample (about 3ml) of the original glycerol solution was retained at the beginning of each experiment. Additional samples of the inner solution were taken at ten minute intervals during the course of an experiment. A typical experiment would have a total duration of one hour. Each of the samples was tested for concentration using an automatic osmometer. The osmometer is normally switched on one hour prior to use and calibrated using 100 and 500 milli-osmole standard solutions as described in the osmometer manual.

These samples provided a time history of the concentration of the inner solution. Ligon, [19], used the following expression to characterize the concentration of the inner solution as a function of time:

$$c_i(t) = [V_i/(V_i+V_o)]c_i(0) + [V_o/(V_i+V_o)]c_i(0)\exp[-tA(D/h)(V_i+V_o)/(V_iV_o)] \quad (4.2.1)$$

where $c_i(t)$ is the concentration of the inner solution at time t , V represents volume, A the surface area of the dialysis membrane, D is the diffusivity of the solute (glycerol) in the membrane, and h is the thickness of the membrane. The subscript (i) implies the inner chamber and the subscript (o) the outer chamber.

This expression can be rearranged and the natural log of each side taken to yield the expression:

$$\ln\left[\frac{(V_i+V_o)}{V_o}\left(\frac{c_i(t)}{c_i(0)}\right) - \frac{V_i}{V_o}\right] = st \quad (4.2.2)$$

where:

$$s = -A(D/h)(V_i+V_o)/(V_iV_o) \quad (4.2.3)$$

For the experimental conditions used in the present work $V_i=400\text{ml}$ and $V_o=1600\text{ml}$ so that:

$$\ln\left[5c_i(t)/4c_i(0) - 1/4\right] = st \quad (4.2.4)$$

and:

$$s = 3.125 \times 10^{-3} A(D/h) \quad (4.2.5)$$

Equation (4.2.4) expresses a linear relationship between the natural log of a simple expression involving the ratio of the concentration at a given time, t , to the initial concentration and time. Figure 4.2.2 shows the results obtained in a typical experiment conducted as a part of the present work. These data clearly show that this type of linear relationship does indeed exist. A linear least squares minimization was utilized to find the best fit line through this data set. The resulting slope is also shown on Figure 4.2.2. The line was not forced to pass through the origin as would be the case using equation (4.2.4) exactly, since the start time of each experiment is somewhat hard to define exactly. This is due to the fact that start-up of an experiment is rather clumsy and errors of up to a minute in the start

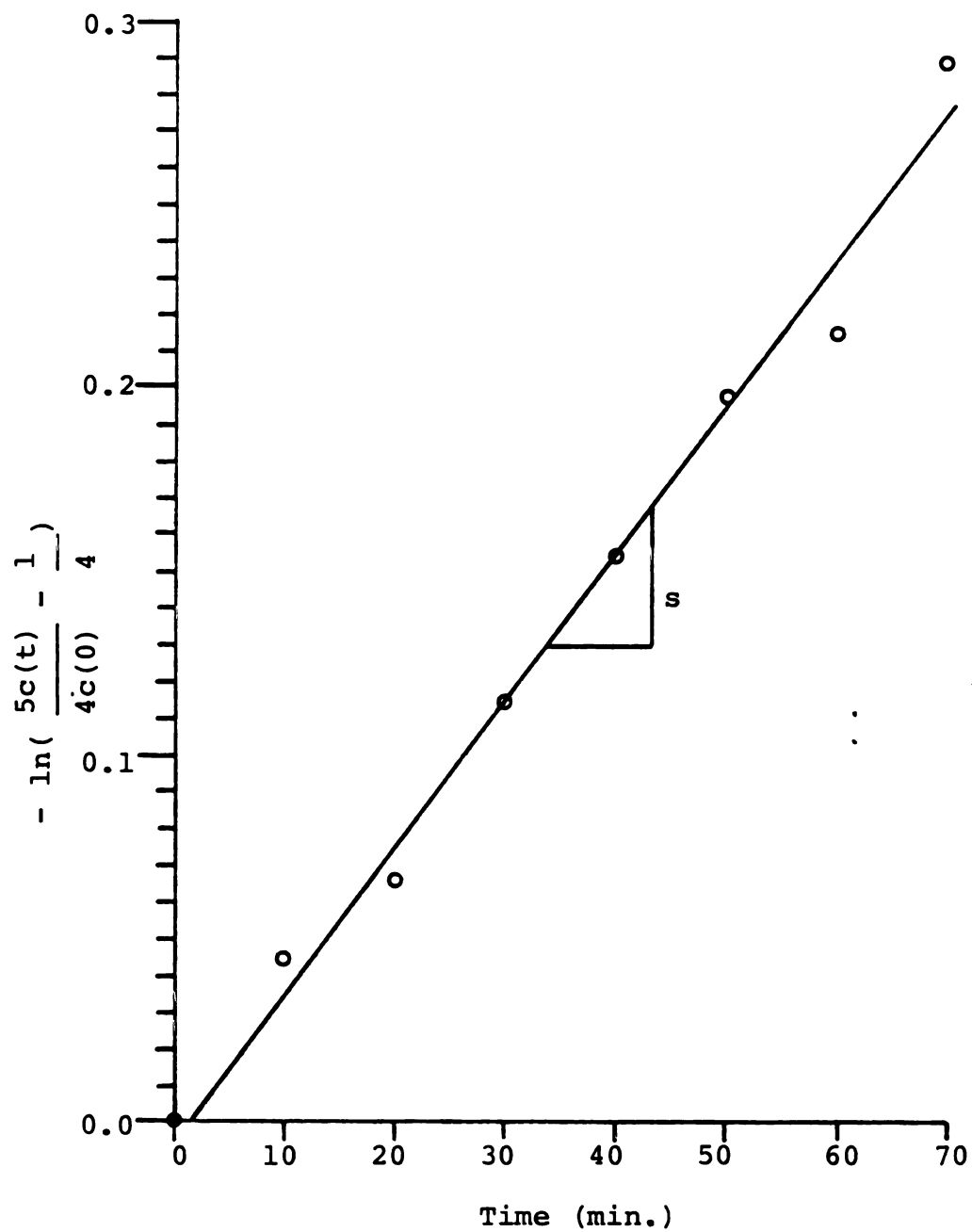


Figure 4.2.2- Typical results of dialysis membrane permeability experiment for Cuprophane 100pm membrane and solute glycerol.

time are possible.

The quantity (D/h) is taken to be the permeability of the membrane. Solving (4.2.5) for (D/h) yields:

$$\omega_d = D/h = s/(3.125 \times 10^{-3} A) \quad (4.2.6)$$

where ω_d will be used to represent the solute permeability of the dialysis membrane. Using the inside diameter of the inner chamber to calculate A and inserting this value into (4.2.6) gives:

$$\omega_d = 7.107 \text{ s} \quad (4.2.7)$$

where if s is in units of $(1/\text{min})$ ω_d will have units of (cm/min) . Using the value of s from the experiment shown in Figure 4.2.2 gives the result:

$$\omega_d = 2.835 \times 10^{-2} \text{ cm/min} \quad (4.2.8)$$

This is compared to the manufacturer's suggested value (see Figure 4.1.3) of $\omega_d = 5.2 \times 10^{-2} \text{ cm/min}$. This relatively large difference may be accounted for by the difference in the temperature at which the two values were obtained. The manufacturer's value was reported for 37°C while that generated in the present work was for room temperature (approximately 23°C). Modifications to the system which would allow one to run experiments at various temperatures (both above and below room temperature) are being considered for future investigations. These

modifications will make it possible to more closely evaluate the correlation between the values obtained using this method and those reported by the manufacturer.

4.3 Laboratory Techniques

This section describes the techniques used while working in the Bio-engineering Transport Processes (BTP) Laboratory of Michigan State University. The tasks described are the formation of unilamellar liposomes ("artificial" cells), and the generation of a series of photomicrographs using the diffusion chamber.

The liposomes used in the present study were formed from L- α -lecithin produced by Leon Laboratories of St. Louis, Mo. (lot number 102112). When not in use the lecithin was kept frozen in a plastic jar. About 10 minutes prior to its use the lecithin was removed from the freezer and placed in a vacuum chamber with dessicant beads in the bottom and allowed to come to room temperature in this dry environment. No vacuum was drawn on the chamber at this time. A small sample of the lecithin (about a 1/8 inch diameter ball) was removed from the bottle and transferred to a glass coverslide. The coverslide and lecithin were placed in a 150ml beaker and about 10ml of a 1:2 (V:V) chloroform-to-methanol solution was added. This mixture was then agitated until all of the lecithin dissolved. The glass cover slide was then removed from the beaker with tweezers and the beaker was placed in the vacuum chamber. A vacuum pump was connected to the chamber and activated. The pump was left on for 30-45 minutes causing the chloroform and methanol to evaporate. This process leaves a thin coating of lecithin on the bottom of the beaker.

The beaker was then removed from the chamber and about 20ml of

the solution in which the liposomes are to be formed was carefully added. Most of the liposomes used in the present study were formed in a 0.2 mole/liter solution of sucrose. Almost immediately after introduction of the sucrose solution a cloud will begin to form in the solution. Care was taken not to disturb the solution once this cloud begins to form. The beaker was covered with Parafilm and placed in a constant temperature bath preheated to 60°C. The bath was turned off just prior to placing the beaker into the bath in order to minimize the disturbance to the solution in the beaker which could be caused by the turbulent mixing of the bath. The cover was placed on the bath and the solution allowed to sit for 24 hours.

After the waiting period the beaker was removed from the bath. At this time one will typically observe that a cloudy ring has formed in the solution. Experience has shown that the best, most useable liposomes will be found in and around this cloudy ring. A pipet was used to draw small samples of the solution (about 5-7ml each) from the beaker. These samples were placed in centrifuge tubes and centrifuged at 15,600 G, 15,000 RPM for 15 minutes. After centrifuging the cloudy material will tend to collect near the top of the tube. Experimental samples were generally taken from the edges of this cloudy material. One faces a trade-off as the best liposomes are generally found within the cloud and yet when sampling from the cloud a significant amount of "junk" is also obtained. No technique has yet been developed for separating the good liposomes from the "junk".

Once the liposomes (or cells) have been prepared one is ready to

generate a series of photomicrographs using the diffusion chamber. The first step in this process is to prepare the chamber itself. One must begin by soaking a small (about 2 inches square) piece of the dialysis membrane in a solution the same as that in which the liposomes (or cells) are in at the onset of an experiment. The dialysis membrane is soaked for 20-30 minutes prior to use in order to insure full saturation. A small sample of the liposome suspension is removed from one of the centrifuge tubes with a micro-pipet and placed in the sample region of the diffusion chamber with the top fitting in an inverted position (see Figures 4.1.1 and 4.1.2). A piece of the dialysis membrane is removed from the solution in which it has been soaking and stretched firmly by holding at each corner. The membrane was carefully stretched across the membrane retaining ring such that no wrinkles are left in the center region. The membrane was released from the users grasp and instead the corners of the retaining ring are used to hold both the membrane and the retaining ring itself. The retaining ring with the membrane clinging to it was then pressed carefully over the inverted top fitting such that the sample is disturbed as little as possible. By holding the retaining ring only, and not the membrane, the membrane is able to conform to the shape of the top fitting by slipping between the retaining ring and the sides of the top fitting. One must be careful to insure that no air bubbles remain between the membrane and the top fitting. The bulk flow channel in the diffusion chamber is filled with the initial solution (taken from the solution in which the membrane was soaking) and the top fitting pressed into place.

The entire chamber was then transferred to the microscope (a Zeiss Universal Research Microscope) and attached to an X-Y traversing mechanism. One 1/8 inch inside diameter Tygon tube is connected to the bulk flow outlet port at one end and the other end of the tube was inserted into a discharge vessel. Another tube is connected to a 35cc syringe and the syringe and tube are loaded with the new solution to be introduced into the chamber. All air bubbles are removed from both the syringe body and the tube. The free end of the tube is then connected to the bulk flow inlet port.

Provided that there are suitable liposomes in the sample used, one is now ready to generate a series of photomicrographs documenting the response of an individual liposome to an induced osmotic imbalance. A Chinon LED Promaster 35mm camera with an automatic winder was used to generate the photos used in the present study. ASA 125 black and white film was used with an exposure time of 1/30 of a second. The microscope illuminator was a 60W tungsten bulb with a voltage input of approximately 12V.

When the operator is ready to begin the stop-watch is started and the syringe plunger slowly depressed. Typically the syringe would be loaded with about 20cc of the new solution and the entire 20cc introduced in about 20-30 seconds. Once the new solution has been introduced into the bulk flow channel a diffusion process begins between the solution in the bulk flow region and that on the other side of the dialysis membrane in the sample region. This diffusion process creates an osmotic imbalance for the cells in the sample

region to which they will respond. Photos are then taken at predetermined intervals. Any special circumstances or observations are noted in a lab log book for future reference.

It was found to be easiest to use two people to generate a series of photographs. One person would monitor the chamber through the microscope keeping the liposome (or cell) of interest in view and in focus while the other would take the photos at predetermined intervals.

An effective technique was also developed by which a series of photos could be generated by a single individual. A computer program written by Tom Gielda, a member of the BTP lab group, when run on the DEC PDP 11/03 mini-computer in the BTP lab would cause the terminal bell to ring once every second. Using this program one could count time by the bells and at the same time keep the liposome of interest in view and in focus. A stop-watch was also used and checked periodically to insure that one did not lose count. Typically for the first 1.5-2 minutes a photo would be taken every 10 seconds so that the operator merely counts to ten and shoots. The time between shots was then increased to 20 seconds and after 3-4 minutes into the experiment extended to 40 seconds. A full series of photos would typically have 15-20 individual images taken over a total period of 6-8 minutes.

The film was then processed using standard procedures outlined in the documentation which comes with the film. Good results were obtained by placing 8 images on a single 8x10 print using a masking

kit. These photos were processed to yield volume as a function of time using the techniques described in Chapter 4 Section 4 of the present work. Figure 4.3.1 shows some typical photos of embryos undergoing a shrink-swell binary flow process with NaCl as the impermeable solute and glycerol as the permeable solute. Figure 4.3.2 shows some typical photos of liposomes generated using the diffusion chamber.

The facilities are also available in the lab to record the entire process on video-tape including a character generator which displays elapsed time directly onto the tape. The same procedure would be followed except that one need not worry about taking photos at specific times. The character generator would be initialized and set running at the beginning of the introduction of the new solution so that no stop-watch was required. This was the method used by Melkerson in his investigation of unilamellar liposome permeability, [26].

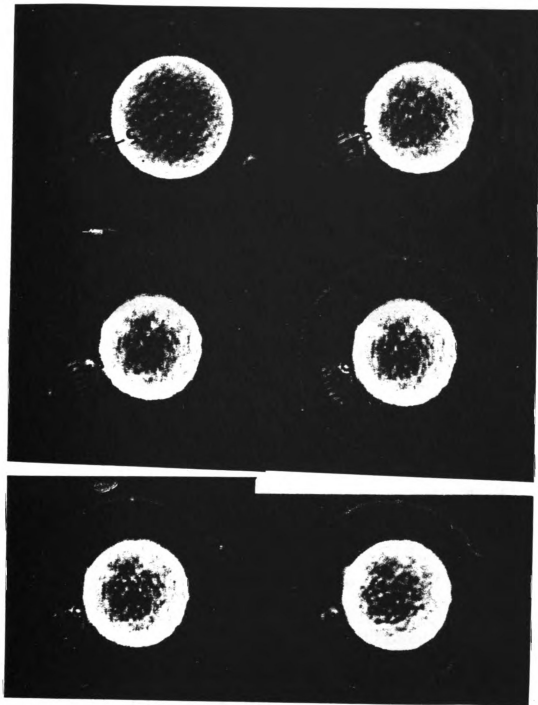


Figure 4.3.1 - Typical ovum photos from diffusion chamber

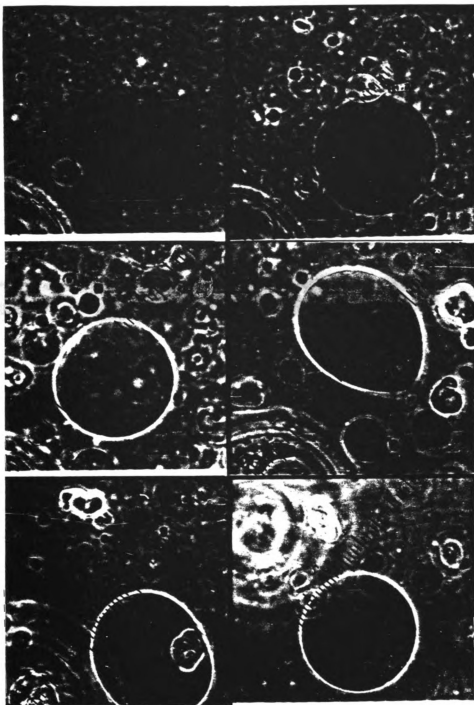


Figure 4.3.2 - Typical liposome photos from diffusion chamber.

4.4 Image Processing Techniques

Cells or liposome ("artificial" cells) were subjected to a pseudo-step change in extracellular concentration using the diffusion chamber described in the previous section. The response of the cell of interest to the non-equilibrium conditions created was documented through a series of photomicrographs taken at known intervals. These photo images were processed to yield the volume of the cell as a function of time. This volume information was then matched to the model of interest in order to estimate the values of the transport properties. There are many possible ways to perform the image processing task.

Shabana, [20], used slides, rather than prints, and projected the image of each cell onto a piece of thin tracing paper. The outline of each cell image was traced onto a separate sheet of paper. These outlines were then cut out of the paper and weighed. The ratio of the weights of each image to that of the first image (at time=0) was taken as the ratio of the area of the projected images. It was then assumed that the cell remained relatively spherical so that the radius could be calculated from the area and from the radius a volume was calculated. This method assumes that the density of the paper is constant. It is also very time-consuming as each image must be processed individually and by hand.

Another method developed as a part of the present work was to use a photo-enlarger to project the image from a black and white film

negative onto a large grid. The diameter of the cell image was then measured by hand in grid units several times for each image. The scale of the image could be calculated by measuring the distance between calibrated scale marks imprinted on each image at the same time the photo was taken. This method worked fairly well when the image on the negative was very distinct and the cell remained circular. Many cases were found in which the image produced by projection of the negative was not distinct enough to clearly define the boundary of the cell. This was particularly true for liposome (artificial cell) images very early in the sequence and for those very late in the sequence. During the majority of the experiment the difference in concentration inside and outside the liposome was large enough to produce a phase-contrast halo around the cell as viewed in the phase-contrast microscope which made the outline of the liposome easily identifiable. However, when the concentration was nearly in balance, as in the very early and very late times, this contrast was not present and the boundary of the cell was not easy to identify in the negative image.

This problem could be avoided by utilizing positive prints rather than the negatives. By manipulating the exposure time of the print the boundary of the cell or liposome could be made distinguishable in most cases. Unfortunately the images could not be made large enough to measure accurately without losing resolution in the photo. Instead the tools of the Computer Image Analysis Laboratory of Michigan State University administered by Professor Richard Dubes were utilized.

The photomicrographic images (typically 8 images to a single 8x10 glossy) were projected onto a video terminal display screen. The projected image could be made sufficiently large to fill the screen without losing significant resolution. A movable cursor was then positioned at several (typically 20) positions around the boundary of the cell and the X-Y coordinates of the cursor at each point were determined by the computer. All of the points entered will have first quadrant coordinates (positive X and positive Y).

An algorithm called CIRCLE (see Appendix E) was then used to perform an integration in radial coordinates to determine the projected area of the image as defined by the 20 input points. The equivalent radius of the image was then calculated as the radius of a perfect circle with the same area as that determined for the image.

The first step in this procedure was to change to an X-Y coordinate system whose origin is within the confines of the point set. The new origin is arbitrarily placed at the location (in terms of the original coordinate system):

$$X_0 = (X_{\max} + X_{\min})/2.0 \quad (4.4.1)$$

$$Y_0 = (Y_{\max} + Y_{\min})/2.0$$

where $X_{\max}$ and $Y_{\max}$ are the largest X and Y values respectively contained in the data set, and $X_{\min}$ and $Y_{\min}$ are the smallest such values. Thus if the new coordinates are designated X' and Y' then:

$$X' = X - X_0$$

(4.4.2)

$$Y' = Y - Y_0$$

These coordinates are then transformed into radial coordinates such that:

$$r^2 = X'^2 + Y'^2$$

(4.4.3)

$$\theta = \arctan(X'/Y')$$

(4.4.4)

For a planar arc the area bounded between the origin and the arc can be expressed as:

$$A = \int_{\theta_1}^{\theta_2} \int_0^r r' dr' d\theta$$

(4.4.5)

$$A = \int_{\theta_1}^{\theta_2} (1/2) r^2 d\theta$$

(4.4.6)

In this algorithm r is assumed to be a quadratic function of θ such that:

$$r = a + b\theta + c\theta^2$$

(4.4.7)

Substituting (4.4.7) into (4.4.6) and performing the integration on theta yields the expression:

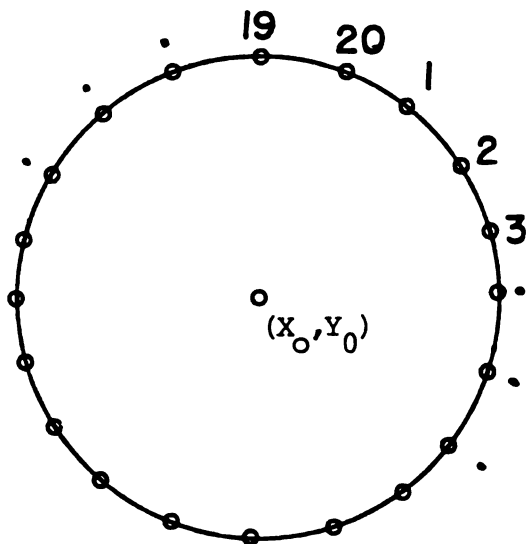
$$A = (1/2) \left[a^2\theta + ab\theta^2 + (2ac+b^2)\theta^3/3 + bc\theta^4/4 + c^2\theta^5/5 \right]_{\theta_1}^{\theta_2}$$

(4.4.8)

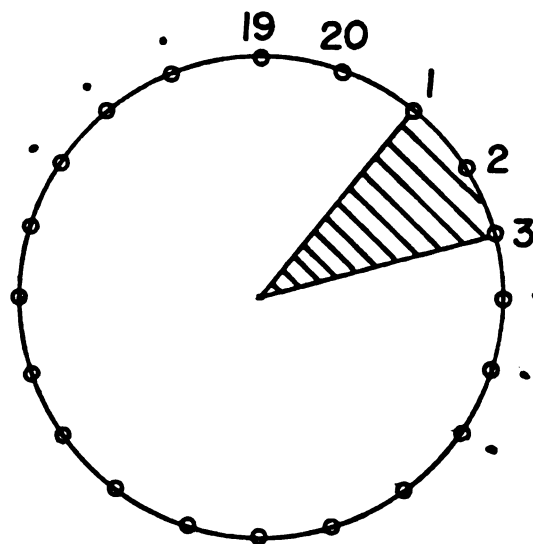
Successive groups of three consecutive points each are used to evaluate the constants a , b , and c and the value of the expression (4.4.8) is evaluated for each arc generated. In practice the (radial) coordinate axes are rotated prior to the calculation of a , b , and c such that $\theta'=0$ for the first point in each group. Thus the value of the lower limit in the integration on theta is always zero and equation (4.4.8) need only be evaluated at one value of theta for each point group (that of the new angle after rotation of the third point in the current group). The contributions to the area calculated for each arc generated from a group of three points are summed to yield the total area. In practice each successive pair of points is used in two arcs generated from the groups of three points (see Figure 4.4.1). Because of this overlapping of arcs the contribution for each arc is halved effectively averaging the contributions. Thus if 20 data points are entered then 20 arcs generated from three successive points each are used to calculate the total area of the image.

Thus the user must enter the data points at relatively equally spaced intervals around the circle, in a clockwise rotation, and in sequential order around the border of the image.

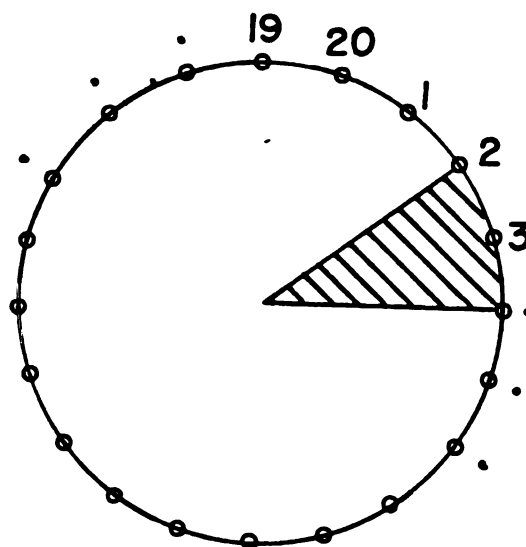
The scale of the image was determined by positioning the cursor at two points on the calibrated scale appearing in each image. This is done prior to entry of the points around the boundary of the image when using the routine TAKEPT (see Appendix E). The distance between these two points was then calculated and output with the data points. By developing all of the images at the same scale with the



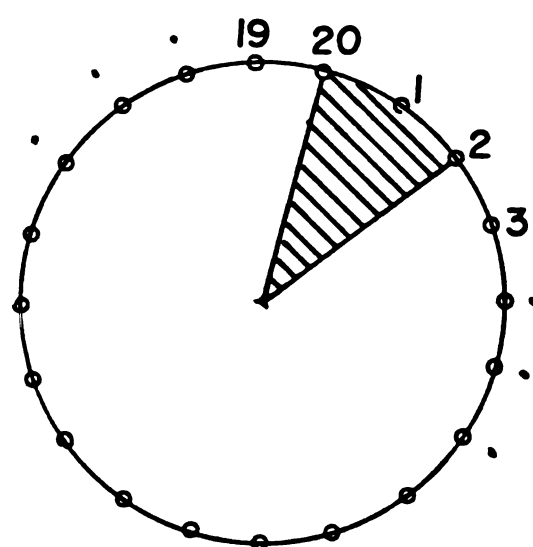
20 points are entered and an arbitrary "center" chosen.



First region integrated.



Second region integrated.



Final region integrated.

Figure 4.4.1- Point selection sequence and integration region selection sequence.

photo-enlarger and not changing the focal distance of the projected images the scales determined for each image should ideally be the same so that each of the individual scales can be averaged to produce the final scale for all images.

This process was accomplished through two sub-programs. The first is called TAKEPT and was used to take the data points from the image. The second is called CIRCLE and does the actual fitting of the points to the circles. These routines are presented in Appendix E of the present work. The output from TAKEPT consists of the two scale endpoints, the number of data points entered around the edge of the image (currently 20 by default), and the actual X-Y data points. This output is written into a file called POINTS.DAT. The routine CIRCLE will read all the values from this file, perform the integration to determine the area, and then output the area calculated, the equivalent radius, and the scale length to a file named POINTS.OUT.

This procedure seemed to work well for most cases. The results for a given image were reproducible to within 1 percent for a clear image and if the points were entered carefully. The times required for image processing using this method were much shorter than those required for hand processing. By processing each image 2-3 times an average value could be determined for the radius of each image and individual errors minimized. The fitting routine, CIRCLE, was very fast requiring about as much time to execute as it takes to enter the next command to the computer. Thus a series of 20 images would typically take about one hour to process using this procedure.

5.1 Osmotic Shrinkage of Hamster Embryos

The results reported here were generated using the program MARBOX to process data reported by Shabana, [20], on the osmotic shrinkage of hamster embryos. These results serve to demonstrate the workability of MARBOX in determining the permeability of a membrane to water using the simplified Kedem and Katchalsky model. The raw data reported by Shabana are tabulated in Appendix F of the present work. The data are reported in this form in order to provide a set of reference values for use by other investigators in testing of other computer routines. In the course of this work one of the problems encountered was the lack of tabulated data available with which to test the programs developed.

Shabana reported numerical values for the nondimensional volume, $V^*=V/V_0$, where V is the volume of the cell as a function of time and V_0 is the initial cell volume, of an individual cell as a function of time. Four embryos were tested in separate experiments. These embryos were made to undergo an osmotic shrinkage using the diffusion chamber described in Chapter 4, Section 1 of the present work. A step change in the concentration of the solute sodium chloride, to which the cells are impermeable, was introduced in the bulk flow region of the chamber and diffused into the sample region through an Enka dialysis membrane, type 80pm. All of the experiments were conducted at room temperature.

Shabana processed the data by hand using a closed-form solution

to the simplified Kedem and Katchalsky equation for this case developed by Terwilliger and Solomon, [21]. This solution assumes that the cells experience a step change in extracellular solute concentration at time equal to zero. With the diffusion chamber this assumption does not reflect the true conditions as the dialysis membrane, which separates the bulk flow region from the sample region, will cause the step-change induced in the bulk flow region to be gradually introduced into the sample region. Terwilliger and Solomon's solution also assumes that the surface area of the membrane remains constant throughout the experiment and is set to the initial value.

The effects of these two assumptions on the permeability values calculated were explored using MARBOX and the data for cell 1. The program MARBOX has the ability to simulate the conditions assumed in Terwilliger and Solomon's solution. It can also take into consideration the lag introduced by the presence of the dialysis membrane in the diffusion chamber. The concentration of the sample region as a function of time is calculated using the permeability of the dialysis membrane, w_d , to the solute present and the algorithm derived in Chapter 4, Section 1 of the present work. The routine MARBOX can also be made to calculate the surface area of the membrane as a function of the volume of the cell assuming a spherical shape for the cell. These results are summarized in Table 5.1.1.

Note that the values for the permeability of the cell membrane to water, L_p , are reported in units of (μ /sec). The conversion to these units from units typical of the Kedem and Katchalsky definition of

Table 5.1.1 - Results for Shabana's cell #1.*

Run #	conditions	L_p	Final sum of squares
1	Shabana's value	16.17	-
2	Area constant $\omega_d = 1000.$	16.69	8.93×10^{-3}
3	Area constant $\omega_d = 1.167 \times 10^{-3}$	25.69	5.63×10^{-3}
4	Area = function(V) $\omega_d = 1000.$	18.72	10.81×10^{-3}
5	Area = function(V) $\omega_d = 1.167 \times 10^{-3}$	28.51	5.48×10^{-3}

* ω_d reported in units of cm/sec, L_p in microns/sec.

solute permeability, ($\text{cm}^3/\text{dyne-sec}$), (see equation 2.3.24) is made by using the expression:

$$L_p(\mu/\text{sec}) = L_p(\text{cm}^3/\text{dyne-sec}) [RT/\bar{v}_w] \times 10^4 \quad (5.1.1)$$

The first value reported in Table 5.1.1 is the value calculated by Shabana. The second value was calculated using MARBOX and assuming that the surface area remained constant and by setting the dialysis membrane permeability, w_d , to a value of 1000(cm/s). Using the present version of the computer routine which calculates the concentration of the sample region if the dialysis membrane permeability is set to a value higher than 998.0 (cm/min) the sample region concentration for all times is set to the value of the concentration of the new bulk-flow solution. This simulates a step-change in the concentration of the sample region. The actual value of the dialysis membrane permeability is typically 10^{-1} to 10^{-3} (cm/min) so that it is safe to assume that when the user inputs a value as large as 10^3 that they wish to simulate a step-change response. These conditions match those assumed in the solution used by Shabana. Comparison of these two values show only a 3% difference between the calculated permeabilities. This slight difference can be attributed to the difference in the weighting of the data applied in the two methods of minimization and to numerical evaluation of the Kedem and Katchalsky equations in MARBOX. In Terwilliger and Solomon's solution the data is linearized and a least squares linear regression used to fit a straight line to the data. In MARBOX a least squares is performed directly on the data

with no linearization. The closeness of these results support the conclusion the MARBOX is indeed executing properly.

The third value in Table 5.1.1 differs from the second value only in that the third value was generated using the manufacturer's recommended permeability value for the dialysis membrane of $\omega_d = 1.1667 \times 10^{-3}$ (cm/sec). The surface area was again assumed to remain constant. The results show a 54% increase in the permeability value calculated for the cell membrane. This result is consistent with the expected results as if one assumes a step-change in extracellular concentration then one will be consistently over-predicting the actual extracellular concentration due to the lag induced by the dialysis membrane. This will force the routine to under-predict the cell membrane permeability in order to maintain the same flux rate at a higher concentration difference. The magnitude of the change was somewhat surprising, however, as initial investigations on the diffusion chamber seemed to indicate that the permeability of the dialysis membrane to the solute sodium chloride was high enough so as to have little effect on the response of the cells. These results show that a small effect on the cell volume response can induce a large change in the calculated permeability values. Thus accurate characterization of the dialysis membrane permeability to various solutes will be vital to the future successful use of the diffusion chamber system.

The final values in Table 5.1.1 include the calculation of the surface area of the membrane as a function of cell volume assuming spherical geometry. The fourth value used the step-change simulation

of the sample region concentration, and the fifth value used the manufacturer's recommended value for the dialysis membrane permeability. These results are consistent with the observation that by assuming constant surface area one will consistently over-predict the actual area and hence will cause an under-prediction of the cell membrane permeability.

It is interesting to note that the final predicted values of the nondimensional volume over time for run 3 and for run 5 were virtually identical. These two runs both used the manufacturer's permeability value for the dialysis membrane but for run 3 the surface area was assumed to remain constant and for run 5 the surface area was calculated as a function of volume. While the permeability values calculated differed significantly the final fit of the model to the data was not significantly different. This would imply that one should maintain consistency when calculating predicted responses from permeability values. That is if the permeability values were calculated assuming constant surface area then the predicted responses should also be calculated assuming constant surface area. If the permeability is calculated assuming a varying surface area then so should the predicted responses. One should be aware of this effect when utilizing published permeability values in simulating cell responses. As long as one remains consistent with the method used to generate the permeability values a good fit should result.

The sum of the squares values reported in Table 5.1.1 can be used as a relative measure of the closeness of the fit between the data

points and the predicted curve. A lower sum of the squares value implies a better fit. From these values it is apparent that the permeability of the dialysis membrane will have a major effect on the fit of the model to the data. The best fit for cell 1 was obtained in run 5 which has the cell surface area being calculated as a function of the volume and uses the manufacturer's value for the dialysis membrane permeability. The predicted response for runs 4 and 5 are presented in Figure 5.1.1. This figure illustrates the effect of the dialysis membrane on the predicted response.

In an attempt to check the value of the dialysis membrane permeability recommended by the manufacturer the modeling subroutine in MARBOX was modified to allow the dialysis membrane permeability to float as a second parameter. Each of the four data sets reported by Shabana were processed using this modified routine. The results are presented in Table 5.1.2. Note that for cell 4 the program was unable to meet the convergence criterion after 60 iterations. This implies that this data set does not have a distinct minimum for this model. The parameter search was varying the parameter values in the third significant digit around the values reported. One can assume that the minimum exists somewhere in the near neighborhood of these values if it exists at all.

The final values calculated for ω_d , the dialysis membrane permeability, were reasonably close to the manufacturer's value and fell to either side of that value. This would tend to support the validity of the manufacturer's values for permeability as well as the sample

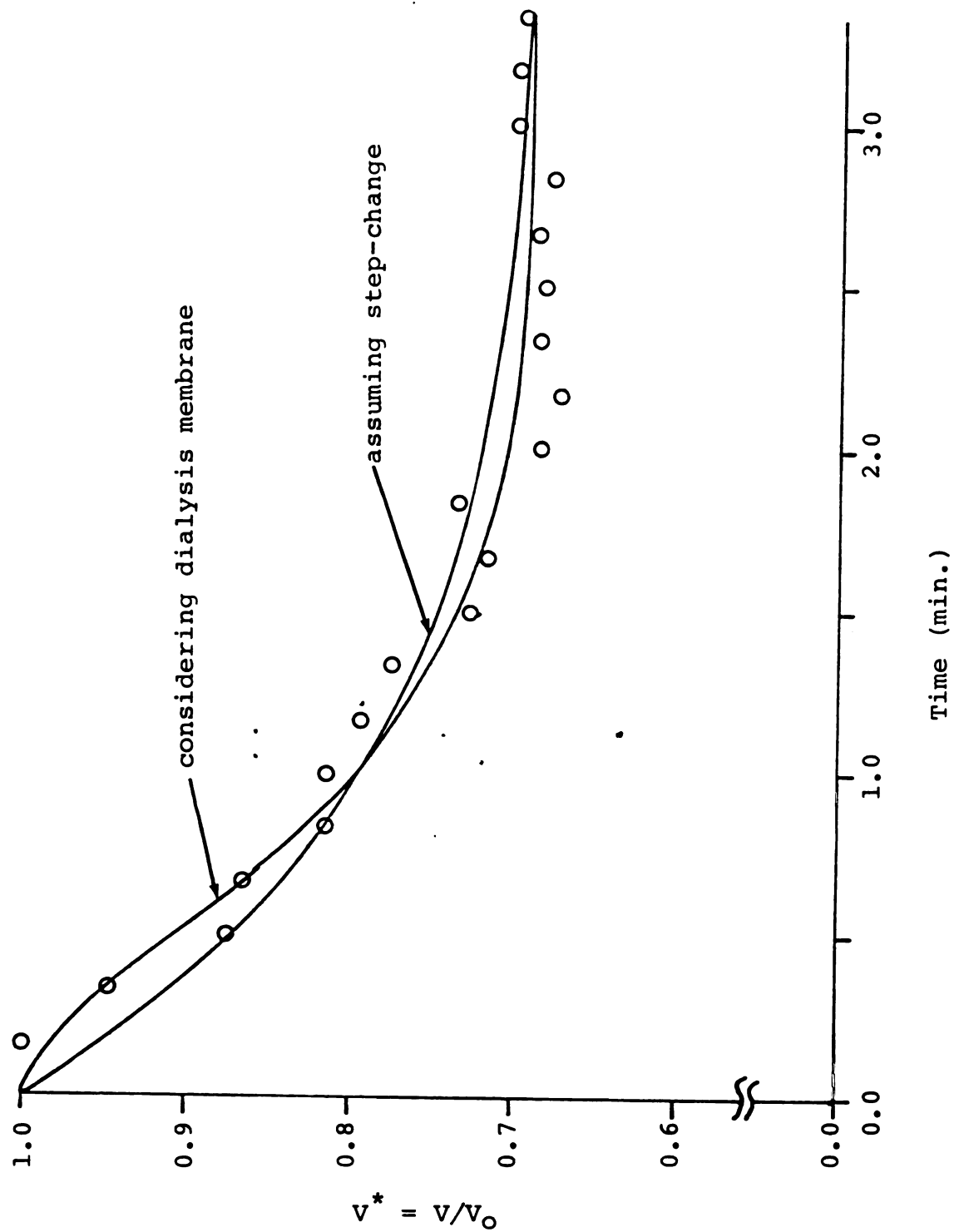


Figure 5.1.1- Measured and predicted response of cell #1.

Table 5.1.2 - Results letting ω_d float as parameter.*

	$\omega_d = 1.167 \times 10^{-3}$	ω_d as parameter #2
cell #1	$L_p = 28.51$	$\omega_d = 1.247 \times 10^{-3}$ $L_p = 27.63$
cell #3	$L_p = 39.13$	$\omega_d = 0.770 \times 10^{-3}$ $L_p = 61.52$
cell #4	$L_p = 27.91$	$\omega_d = 0.97 \times 10^{-3}$ $L_p = 31.81$

* ω_d reported in units of cm/sec, L_p in microns/sec.

Table 5.1.3 - Final permeability values as determined by parameter estimation routine.**

	Value reported by Shabana	Value calculated using MARBOX
cell #1	16.39	28.51
cell #2	20.00	46.72
cell #3	19.17	39.13
cell #4	16.17	27.91

** All L_p values reported in units of microns/sec.

region concentration algorithm derived and used as a part of the present work. The effect on the calculated permeability of the cell membrane was very dramatic. This demonstrates again the need to carefully characterize the dialysis membrane permeability for the solutes of interest.

The final recommended permeability values for each of the four cells as determined by the parameter estimation routine are presented in Table 5.1.3. These values were generated using the manufacturer's recommended permeability values for the dialysis membrane and by calculating the surface area of the cell as a function of cell volume assuming a spherical geometry for the cell. It is interesting to note the fairly wide variation in the permeability values calculated. These cells appear to be identical visually and yet their permeability to water varies greatly. This points out the advantage of the diffusion chamber in the investigation of population distribution information through the observation of individuals within the population. The final simulated response for cells 2, 3, and 4 are shown in Figures 5.1.2 through 5.1.4.

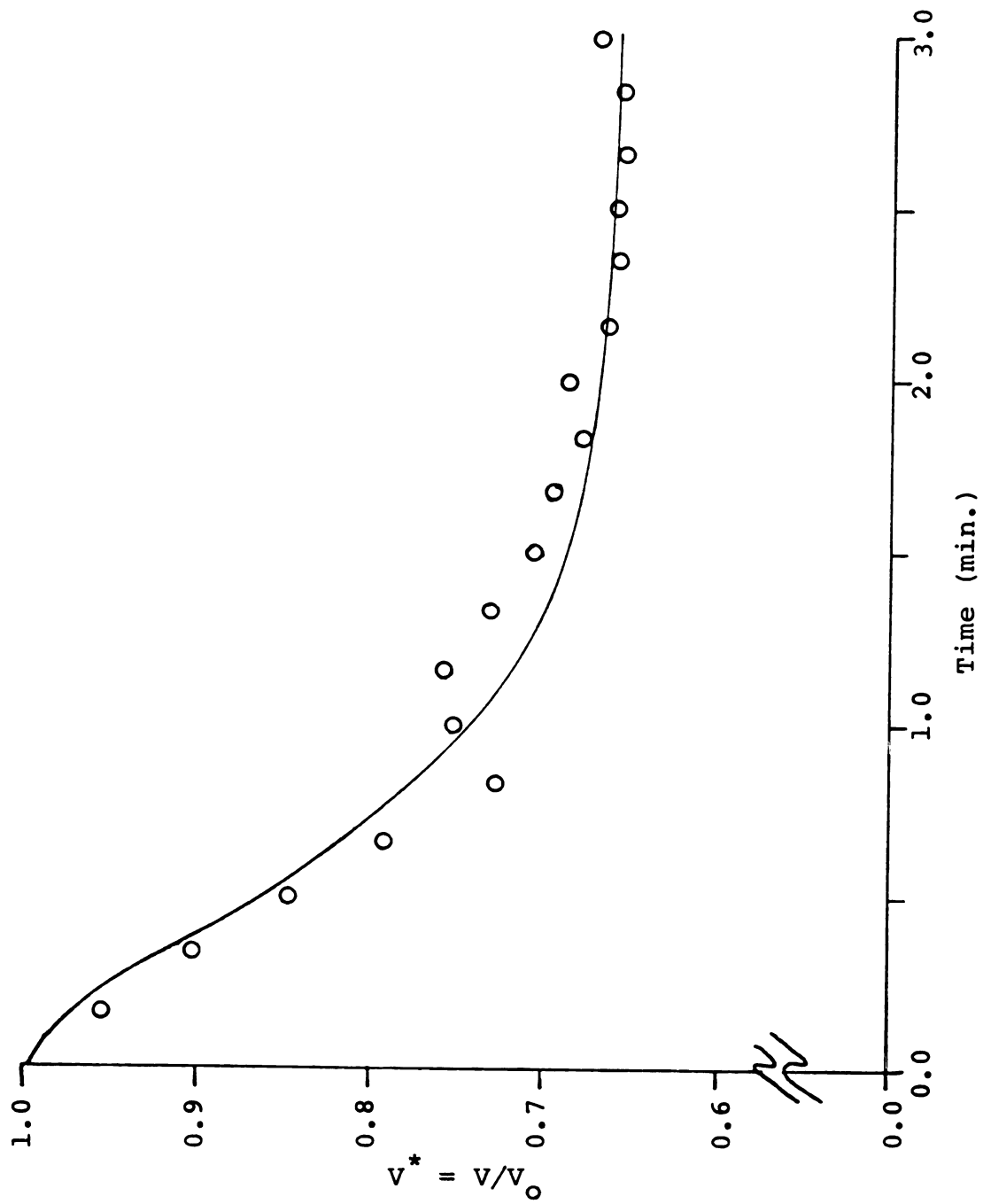


Figure 5.1.2- Measured and predicted response of cell #2.

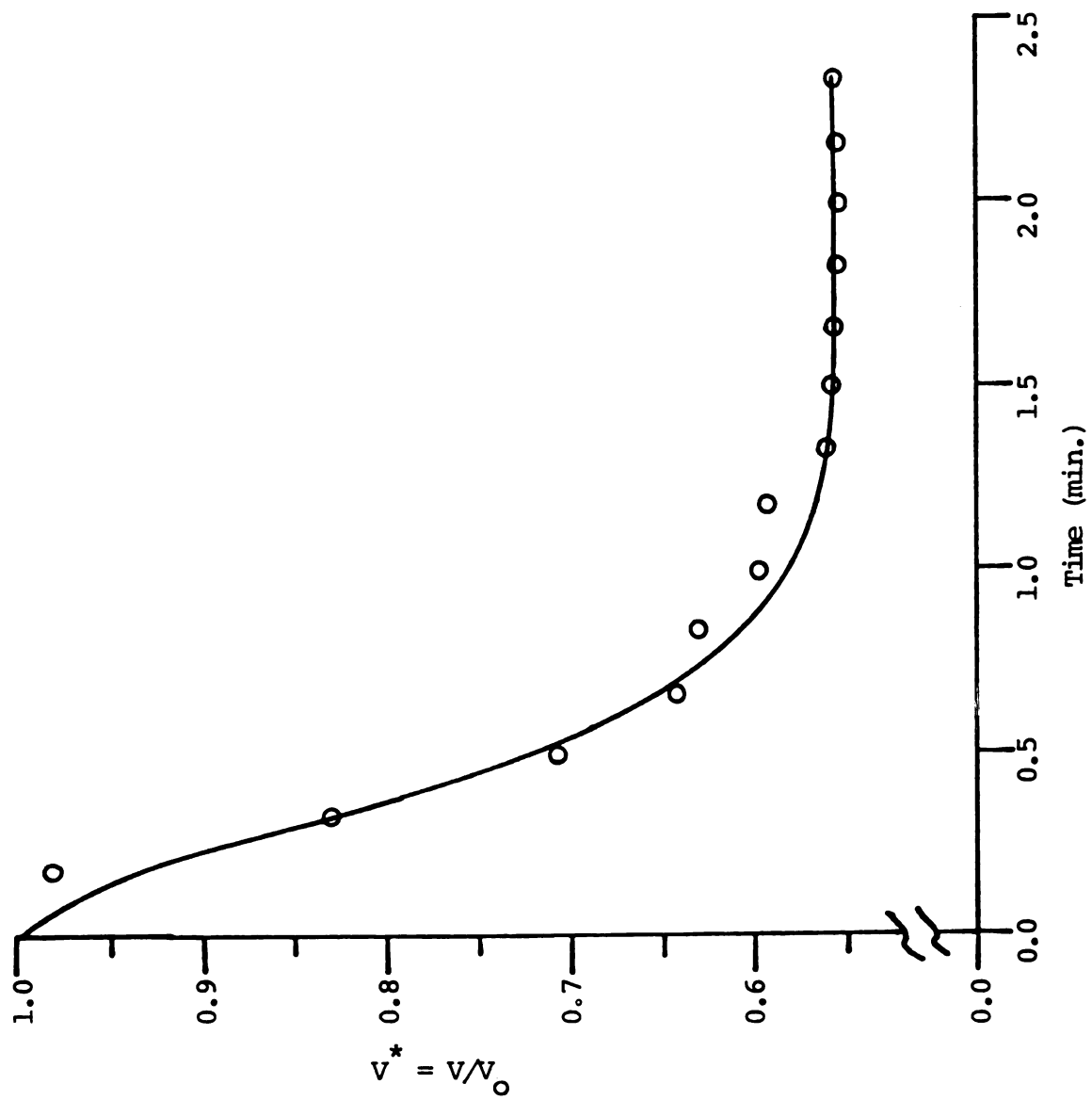


Figure 5.1.3- Measured and predicted response of cell #3.

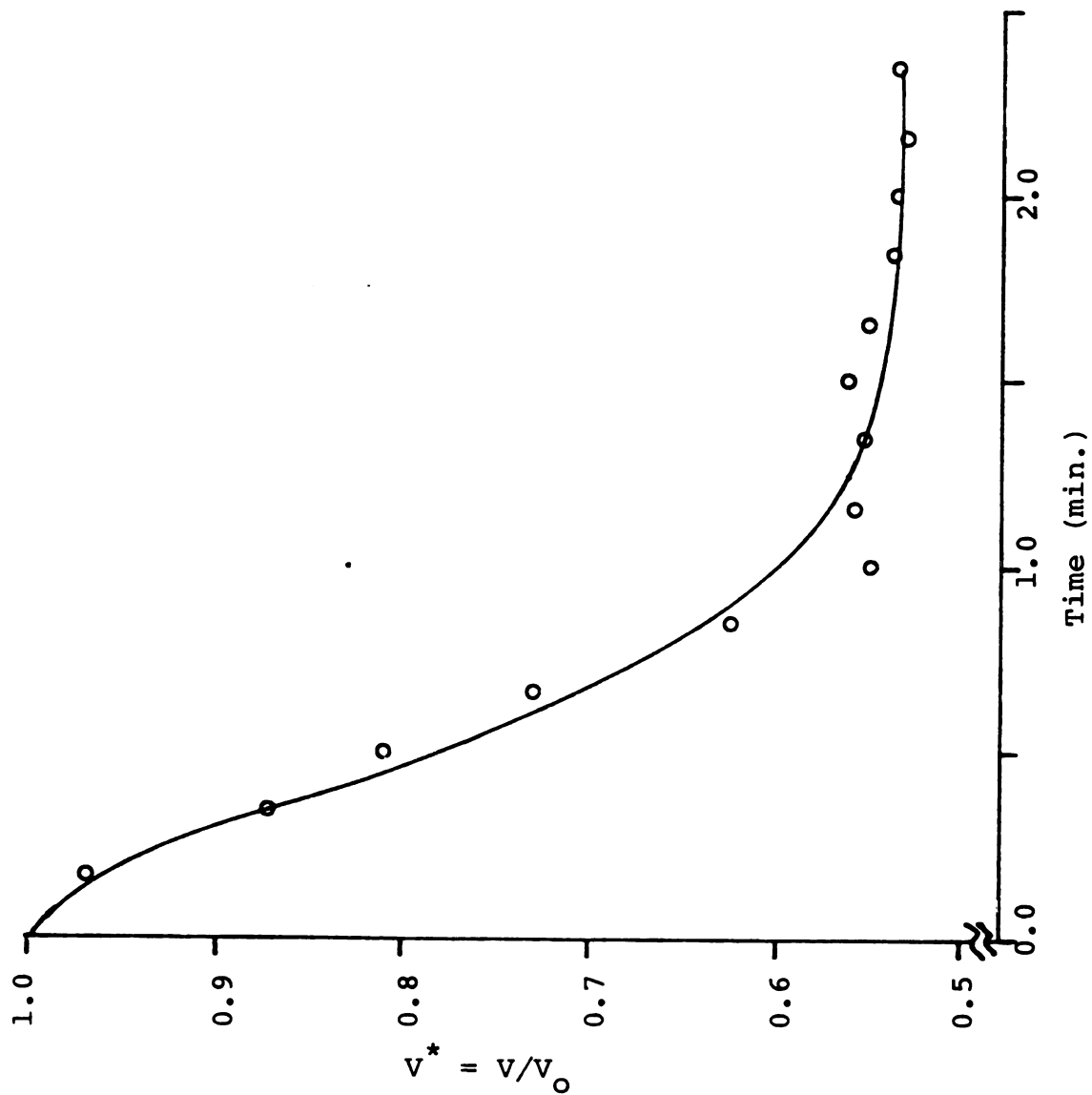


Figure 5.1.4- Measured and predicted response of cell #4.

5.2 Binary Flow in Unfertilized Hamster Ova

Photomicrographic images were taken of two unfertilized hamster ova undergoing a process of binary flow. Each ovum was taken from separate sample groups and tested in separate experiments using the diffusion chamber. Both of the experiments were conducted at room temperature and within 3 hours of removal of the ova from the host hamster. The ovum designated 5 was subjected to a change in extracellular concentration of the permeable solute glycerol from 0.0 to 0.20 (osmol/kg). Ovum 6 was subjected to a change in glycerol concentration from 0.0 to 0.25 (osmol/kg). Both ova were initially in a 0.3 (osmol/kg) saline solution.

The resulting photomicrographic images were processed using the computer image analysis techniques described in Chapter 4 Section 4 of the present work to yield the volume of each ovum as a function of time. These data were then used to test the execution of the parameter estimation routine for the three parameter case. The parameter routine was able to converge for both sets of data. Figure 5.2.1 shows the measured and "best fit" predicted response for these two ova.

Two additional ova were tested by Tu, [29], and recorded on video-tape. Both ova were subjected to a change in glycerol concentration from 0.0 to 0.25 (osmol/kg). Diameters of the ova were determined by direct measurement of the projected image from a television screen. One of these data sets was successfully processed using

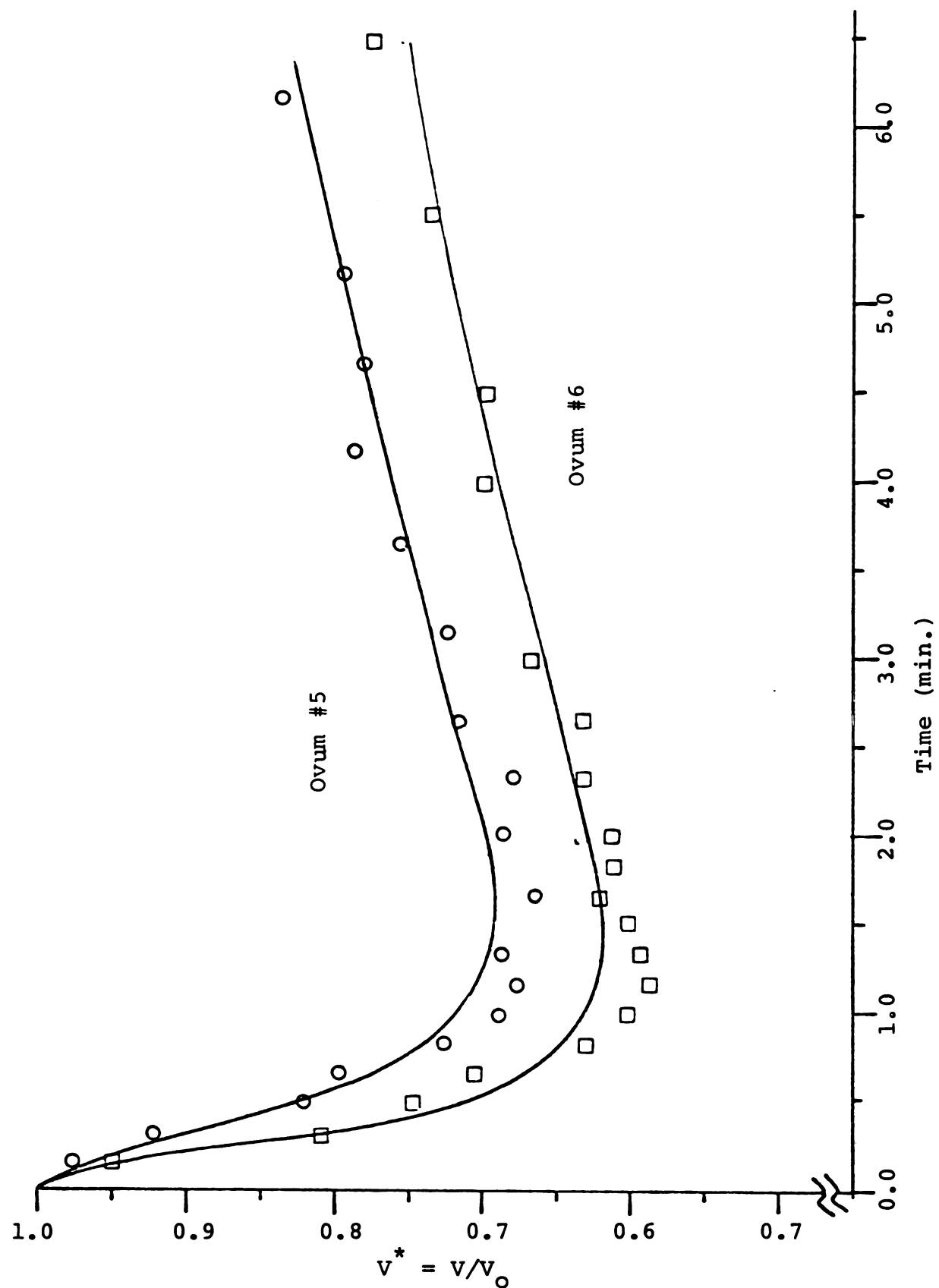


Figure 5.2.1 - Measured and predicted response of ova #5 and #6.

the parameter estimation routine. Figure 5.2.2 shows the measured and "best fit" predicted responses for that ovum, designated ovum 8.

In the processing of each data set using the parameter estimation routine the parameter search would consistently drive the value of the hydraulic permeability to the upper bound set for that parameter. Thus the significance of the calculated hydraulic permeability values must be questioned. For this case it is believed that the dialysis membrane itself is rate limiting on the system, due to its relatively low permeability to glycerol (3.8 microns/sec), and that the ovum remained in a quasi-equilibrium state, with respect to the water content, during the initial period of rapid shrinkage. It should be noted that this should not effect the validity of the calculated glycerol permeability values as the solute permeability will be primarily determined by the rate at which the cell recovers during the later stages of the experiment. Since the ova solute permeability values are much lower than the solute permeability of the dialysis membrane the model should produce an accurate prediction of the solute transport process for all times. It is only the water transport that is limited by the slow response of the dialysis membrane and an upper bound to the rate of water transport is quickly approach for reasonable hydraulic permeability values (30-40 microns/sec).

The solute permeability values calculated were 4.2×10^{-2} , 3.0×10^{-2} , and 2.9×10^{-2} (microns/sec) for ova 5, 6, and 8 respectively. Jackowski, et al, [30], reported glycerol permeability values for mouse ova at room temperature of 1.7×10^{-2} (microns/sec). This

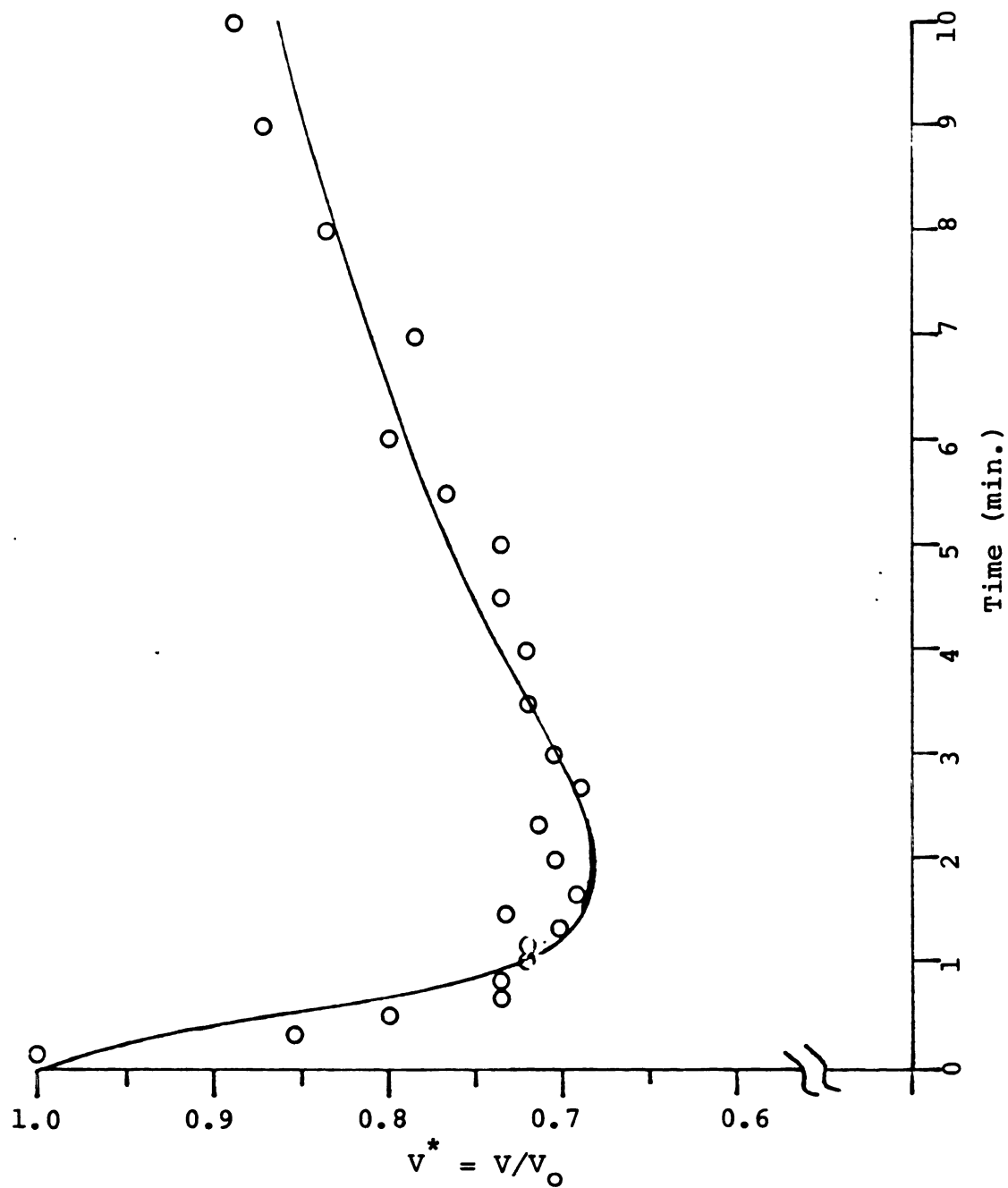


Figure 5.2.2 - Measured and predicted response of cell #8.

would indicate that the hamster ova are much more permeable to glycerol than are the mouse ova. This is somewhat surprising as one would expect these two systems to be quite similar. Further work should be conducted in order to substantiate these findings.

For both ova 5 and 6 the calculated value of the reflection coefficient, σ , was 1.0. For ovum 8 the calculated value of σ was 0.76. The validity of these values is questionable as the reflection coefficient characterizes solvent-solute interactions and the hydraulic permeability, and hence true rate of solvent transport, is unknown due to the rate limiting problem.

One significant difference between the data generated using the techniques of image analysis and that generated by direct measurement of cell diameter can be seen quite clearly in Figures 5.2.1 and 5.2.2. That is that the image analysis technique produces "smoother" data. The fluctuation in the data obtained by direct measurement is due to the difficulty in determining the diameter of a cell which deviates even slightly from a spherical shape and to the poor resolution obtained with use of a ruler. The image analysis technique performs an integration to determine the projected area of the image and determines an equivalent radius for a circle with the same area. This method thereby accounts for deviations from a spherical shape more accurately than direct measurement. The resolution of the grid on the computer display terminal used to enter points around the cell boundary is also much greater (approximately 1 part in 400 for a typical image). This results in more accurate and "smoother" data.

It should be noted that the data obtained using image analysis were checked by using a photo-enlarger to project several of the photo images onto a grid and measuring the diameter directly. These results were consistent with those obtained using image analysis and indicated that no biasing of the data occurred through the image analysis procedure itself.

The tabulated values of the nondimensional volume of ova 5 through 8 are presented in Appendix H of the present work. Also included in this appendix are the input conditions for the parameter estimation routine used to generate the results presented above.

The results presented here confirm that the parameter estimation routine is capable of handling the three parameter problem, and that the image analysis methods yielded accurate data. It will be left to future investigators to increase the data base and to investigate the implications of these results.

CHAPTER 6

Conclusions

The results of this study demonstrate the workability of the diffusion chamber system in determining the passive transport properties of an individual cell membrane. This work has also shown that the parameter estimation computer algorithm is capable of handling models of 1, 2, or 3 parameters. The routine has reached convergence on several sets of experimental data. The techniques of computer image analysis have also been shown to work well on well defined photo images yielding accurate, high resolution data.

Four separate data sets reported by Shabana, [20], for individual unfertilized hamster ova subjected to an osmotic shrinkage process have been successfully processed using the parameter estimation routine. One conclusion drawn from these results was that the effects of the dialysis membrane on the concentration history of the sample region must be taken into consideration when processing data obtained from the diffusion chamber. Shabana processed the same data sets by hand using a closed-form solution to the simplified Kedem and Katchalsky equations. This solution assumes that the cells experience a step change in extracellular concentration. When using the diffusion chamber this assumption does not reflect the true experimental condi-

tions. The presence of the dialysis membrane between the bulk flow region and the sample region of the chamber introduces a lag in the response of the sample region to the step change in concentration induced in the bulk flow region. As expected it was found that when processing the same data sets using the relationship derived in Chapter 4 Section 1 of the present work to calculate the concentration of the sample region as a function of time the resulting hydraulic permeability values calculated were consistently higher than those calculated using the closed-form solution. For the four data sets processed Shabana reported hydraulic permeability values in the range of 16.2 to 20.0 (microns/sec) using the closed-form solution. Using the parameter estimation routine with the concentration algorithm included the same data sets yielded hydraulic permeability values in the range of 27.9 to 46.7 (microns/sec).

In order to test the parameter estimation routine on a model of three parameters and to test the image analysis methods as well binary flow experiments on unfertilized hamster ova were conducted. Two ova were subjected to changes in extracellular concentration of the permeable solute glycerol and their responses were documented via a series of photomicrographic images. A third ovum was tested by another investigator also using the diffusion chamber. The response of this third ovum was documented on video-tape.

The photo images were processed using the image analysis techniques described in Chapter 4 Section 4 of the present work. The accuracy of these data was checked by directly measuring the diameter

of several of the photo images and comparing the resulting predictions of volume at various times. The two methods gave similar results although the image analysis results were much more consistent and showed the expected trends with much less fluctuation. From this it was concluded that the image analysis technique yielded more accurate and more consistent data than did the direct measurement method. The time involved in using the image analysis techniques was longer however due to the need to process the film. The actual measurement process for the image analysis was approximately the same as the direct measurement process.

Each of these data sets was successfully processed using the parameter estimation routine linked to the Kedem and Katchalsky model for coupled binary flow in a membrane. In each case it is believed that the permeability of the ovum to water was so high that the dialysis membrane permeability to the solute glycerol became the rate limiting factor in the diffusion chamber system. This could be observed in that as the hydraulic permeability value increased above a value of approximately 30.0 (microns/sec) there was no significant effect on the predicted curve. Thus the values of the hydraulic permeability calculated for these data sets have no significance.

This rate limiting of the solvent transport should not have a significant effect on the calculated solute permeability values as the dialysis membrane is significantly more permeable to the solute than are ova. The solute permeability values calculated will be primarily determined by the rate of volume increase during the later stages of

the experiment. The glycerol permeability values calculated ranged from 2.9×10^{-2} to 4.2×10^{-2} (microns/sec) for the three cells processed.

CHAPTER 7

Suggestions for Future Work

The results of the present work show great promise for utilization of the diffusion chamber in conjunction with various computer routines in the study of the permeability characteristics of cell membranes. During the course of this work several points which should be investigated in the future were revealed.

The first of these is the characterization of the dialysis membrane permeability. In the processing of data obtained using the diffusion chamber the permeability value of the dialysis membrane to the solutes used was found to play a key role in the resulting cell membrane permeabilities calculated as describe in Chapter 5 of the present work. Thus it will be vital to know accurately the permeability of the dialysis membrane to each of the solutes of interest. As future research plans include modification of the diffusion chamber to accommodate a temperature regulating system one will need to know the permeability characteristics of the dialysis membrane as a function of temperature as well.

The modification of the diffusion chamber to accommodate a temperature regulating system is, in itself, another project which should

be undertaken in the future. As the overall research effort in the BTP lab centers on the investigation of the effects of cryopreservation procedures on various cell types, investigation of cell transport properties as a function of temperature becomes a key concern. In order to accurately predict the response of a particular type of cell to a particular freezing protocol one will need to know the cell's permeability characteristics over the entire range involved in the freezing protocol.

Another project which should be undertaken is the modification of the bulk flow system. The present system of pressurized bottles has several drawbacks. One is that as flow is introduced into the diffusion chambers bulk flow channel the initial surge can cause severe disturbance in the sample region. This can make it difficult to keep the cell of interest in view and in focus. This problem might be solved by lowering the pressure in the bottles and regulating it more closely. This would require that a low pressure regulator be installed in the air supply line. Another problem with the present bottle system is that due to the way the bottles are interconnected a relatively long section of tubing exists between the Y-junction which connects the two bottles to a single pump and the bulk flow inlet port of the diffusion chamber. This creates a lag of unknown duration between the time the pump is activated and the time the new solution actually enters the bulk flow channel. This makes it rather difficult to clearly define time equal to zero in an experiment. A Y-junction closer to the inlet port, and use of a separate pump for each bottle, would probably resolve this problem. While the technique utilizing a

manually depressed syringe does not encounter either of these problems it is more cumbersome to work with and was found by Melkerson, [26], to be less reproducible. Thus a modified bottle system is the preferable option to pursue.

During several runs using the diffusion chamber it was found that a sudden rippling of the dialysis membrane would occasionally cause the entire contents of the sample region to be flushed out of the sample region. It was also found that when working with the relatively small liposomes that they had a tendency, on occasion, to drift about and sometimes squeeze between the dialysis membrane and the top fitting of the diffusion chamber. A modification of the membrane retaining ring would probably solve both of these problems. It might be useful to replace the retaining ring with a cup-like fitting which would then sandwich the dialysis membrane firmly between two rigid supports. A small hole in the center of the cup corresponding to the hole in the spacer ring (which makes up the sample region) would allow the bulk flow to come into direct contact with the dialysis membrane. The base of the cup should be made as thin as possible in order to prevent the region within the hole from becoming an isolated pocket bypassed by the bulk flow. As the hole in the center will allow viewing of the sample region this cup-retainer could be made from most any material, opaque or transparent. A retainer such as this would prevent rippling of the dialysis membrane and press the membrane firmly against the top fitting preventing leakage from the sample region.

Many points regarding the statistical nature of the parameter

search exist which should be explored in the future. To the best knowledge of the author very little investigation into these aspects of the parameter search applied to the problem of passive cell membrane transport has been performed. Enough work relating to this area potentially exists to justify devotion of an entire thesis project to just this problem. For instance it would be interesting to apply a sequential method of minimization to this problem. That is, a method which adds data points one at a time and adjusts the parameter values with each additional point. This type of routine can often provide interesting insights into both the data being processed and the model being used to simulate the data. It would also be interesting to map out the sensitivity coefficients as a function of time for the final parameter values as this can often provide insights into the model which are not obvious from inspection of the modeling equations. For instance Papanek's observation (quoted in Chapter 3 Section 7 of the present work) that in a binary flow shrink-swell situation the initial rate of shrinkage will be almost entirely dependent on the value of solvent permeability, and that the shape of the minimum will be primarily dependent on the value of the interaction coefficient, σ , and that the rate of swelling during the final stages will be dependent on the solute permeability value, should be reflected in the sensitivity coefficients. Thus one would expect the sensitivity coefficient relating to solvent permeability to have relatively high values during the period of initial shrinkage and to drop off thereafter. Similarly for the other sensitivity coefficients.

The investigation of models other than the Kedem and Katchalsky

permeability model used in the present work is an obvious area of research for the future. This is particularly true with regards to using the Johnson and Wilson power series solution, outlined in Chapter 3 Section 6 of the present work, as a generator of starting estimates to the parameter values. Use of this model should improve the initial estimates and hence reduce the total computational effort required. The Papanek model, described in Chapter 3 Section 7 of the present work, also needs to be investigated. This could be done in conjunction with the investigation of the statistical nature of the parameter search applied to the passive transport problem. Various methods exist by which one can statistically compare two or more models. It would be interesting and very useful to perform such a comparison between the Papanek model and the more commonly used K-K model. It would also be interesting to apply the Kedem and Katchalsky resistance model, particularly to the dialysis membrane where one could assume a capillary structure and thereby simplify the model considerably.

A final area which needs further work is the development of image analysis methods. Reasonable results were obtained in the present work using very simple methods of analysis. More sophisticated methods with better reproducibility should be developed as the computer image analysis techniques have the potential for becoming an elegant and easily utilized method of photo data processing. This project in itself could become a major undertaking.

The present work has demonstrated the workability of the diffu-

sion chamber as an experimental tool, and the computer programs written as a part of that work as analytical tools. It is now up to future investigators to refine and expand these techniques.

APPENDICES

APPENDIX A

Subprogram Unit MARBOX

The ordinary least squares based parameter estimation routines main driving program is contained in a file name MARBOX.FOR. This file also contains three support routines called only from .MAIN. which will also be described here. These support routines are BOUNDS, MINVER, and DETERM.

The input to the routine is through a data file. Input is read from logical unit 2 so that under the RT11 operating system the input file has the name FIN2.DAT (see RT11 FORTRAN IV Users Guide page 3-5). All input is in the free format mode so that no special formatting is necessary. Input values on a single line should be separated by commas. The order of input and the definition of each of the input variables is included in the program listing.

The main routine embodies a variety of parameter estimation routines all based on the ordinary least squares (OLS) method. The form of the routine to be executed is set by the user through the input variables BOXFLG, and MFLAG. By setting one of these variables to 1 the user introduces modifications to the basic OLS routine. If both of the flags are set to 0 then the routine is the OLS method with

a modification which checks for violation of the parameter bounds set by the user during input. The OLS routine will have the fastest execution time per iteration on the parameter values but will probably require more iterations to reach convergence than the other forms.

The first modification allows the user to set upper and lower limits on the values of each of the parameters. In the current version of the program this option is invoked by default. Within the input section of the program there is a set of statements which can be easily modified to allow the user to make this option selectable via the input variables. This is not recommended however as this option adds very little to the execution time of the program and will prevent the search from diverting to unrealistic values of the parameters (such as $\sigma < 0$ or $\sigma > 1.0$). For instance in the K-K equations negative values for any of the parameters will result in meaningless solutions. Thus the user should set a lower limit on each of the parameters of no less than 0.0. After each iteration the newly calculated parameters are compared to the bounds set by the user. If the new value violates either of the bounds then the value of the parameter is set to the value of the violated bound and execution continues. The checking is done in subroutine BOUNDS.

The second option is invoked by setting MFLAG=1. This causes the routine to execute as Marquardt's method (see Chapter 4 Section 5 of the present work). Marquardt's method modifies both the size and direction of the step in parameter values taken in each iteration.

The final option is invoked by setting BOXFLG to one. This causes the routine to execute under the Box-Kanemasu minimization method.

Note that the Box-Kanemasu method is not compatible with Marquardt's method so that one should not set both BOXFLG and MFLAG equal to one.

The convergence of the routine is based on the change in the parameter values through the input variable TOLER. Execution is terminated and final values of the cell volume with time calculated when the condition:

$$\Delta\beta_i/\beta_i < \text{TOLER}$$

is met for all of the parameters ($i=1,2,\dots,P$).

The support routine MINVER is a subroutine which will invert an ($n \times n$) matrix where ($n \leq 3$). Subroutine DETERM calculates the determinant of an ($n \times n$) matrix where ($n \leq 3$).

In utilizing this routine the user should have patience. The results in a parameter estimation routine can be unexpected and the corrective actions required to enable convergence are often learned only through experience. By utilizing the various options available the user can run a particular data set under several different procedures. For instance the user can "trick" the routine into manipulating only one of the parameters in the three parameter model

by setting the upper and lower bounds of the other two parameters to the value of the initial estimate and invoking the bounding option. Thus one can "zero in" on the final parameter values one at a time. This is often helpful when the initial estimates are not very good and the routine has trouble converging. By allowing only one parameter to vary at a time the user can often get an idea of the direction in which to change the initial estimates in order to approach the solution. Even this method will not work in every case, however.

C PROGRAM REFERENCE: BECK, JAMES V.; ARNOLD, KENNETH J.;
 C PARAMETER ESTIMATION IN ENGINEERING
 C AND SCIENCE, JOHN WILEY _SONS, INC.,
 C 1977, CALL TA340.B39.

C THIS PROGRAM IS A PARAMETER ESTIMATION ROUTINE BASED ON
 C BASED ON THE ORDINARY LEAST SQUARES ROUTINE. IT ALSO HAS
 C AN OPTION FOR CONSTRAINING THE PARAMETER SEARCH.
 C IT IS ALSO SETUP TO RUN AS MARQUARDT'S ROUTINE OR AS THE
 C BOX-KANEMASU METHOD.

C INPUT TO THIS ROUTINE IS THROUGH A FILE NAMED FIN2.DAT
 C THIS FILE SHOULD CONTAIN IN THE FOLLOWING ORDER:

C DEBUG, MFLAG, BOXFLG
 C ACONFL, P
 C II
 C T(1),Y(1)
 C T(2),Y(2)
 C . .
 C . .
 C T(II),Y(II)
 C BETA(1),LBDETA(1),UBBETA(1)
 C . . .
 C BETA(P),LBBETA(P),UBBETA(P)
 C TOLER
 C DT
 C CIZERO, CISTEP
 C CSZERO, CSSTEP
 C DPERMI, DPERMS, DEPTH
 C RADIUS
 C VDEAD
 C (IF MFLAG=1 (MARQUARDTS METHOD) THEN:)
 C LAMBDA, FACLAM

C WHERE:
 C DEBUG INTEGER DEBUG OUTPUT CONTROL CHARACTER
 C BOXFLG INTEGER FLAGS FOR BOX-KANEMASU METHOD
 C 1 = YES
 C 0 = NO
 C MFLAG INTEGER FLAGS FOR MARQUARDTS METHOD
 C 1-YES
 C 0-NO
 C ACONFL INTEGER FLAGS FOR ASSUMPTION OF CONSTANT MEMBRANE
 C SURFACE AREA
 C 0 = CONSTANT AREA
 C 1 = NON-CONSTANT AREA
 C P INTEGER NUMBER OF PARAMETERS IN MODEL
 C II INTEGER NUMBER OF DATA POINTS
 C T REAL(II) TIME DATA IN VECTOR FORM IN (SEC)
 C Y REAL(II) DEPENDENT VARIABLE IN VECTOR FORM
 C BETA REAL(P) PARAMETER ESTIMATES
 C LBBETA REAL(P) LOWER BOUND ON PARAMETERS
 C UBBETA REAL(P) UPPER BOUND ON PARAMETERS
 C DT REAL APPROXIMATE VALUE OF TIME INCREMENT FOR

```

C          INTEGRATION STEP SIZE (SINCE INCREMENT
C          MAY NOT FIT TOTAL TIME INTERVAL EVENLY
C          ACTUAL STEP SIZE IS CALCULATED IN "RK4"
C      CIZERO  REAL      INITIAL CONCENTRATION OF IMPERMEABLE SOLUTES
C      CISTEP  REAL      NEW IMPERMEABLE SOLUTE CONCENTRATION
C      CSZERO  REAL      INITIAL CONCENTRATION OF PERMEABLE SOLUTE
C      CSSTEP  REAL      NEW PERMEABLE SOLUTE CONCENTRATION
C          NOTE- CONCENTRATIONS IN (MOLES/CM**3)
C      DPERMI  REAL      DIALYSIS MEMBRANE PERMEABILITY TO SOLUTE I (CM/SEC)
C      DPERMS  REAL      DIALYSIS MEMBRANE PERMEABILITY TO SOLUTE S (CM/SEC)
C      DEPTH   REAL      DEPTH OF SAMPLE REGION IN DIFFUSION CHAMBER (CM)
C                      DEPTH = (TRANSFER AREA) / (SAMPLE VOLUME)
C      RADIUS  REAL      CELL RADIUS IN cm
C      VDEAD   REAL      FRACTION OF INITIAL VOLUME ATTRIBUTED TO OSMOTIC
C                      DEAD SPACE, THAT IS SPACE NOT INVOLVED IN
C                      ACTIVE TRANSPORT. ( 0.0 <= VDEAD => 1.0 )
C      LAMBDA  REAL      SEE MARQUARDT'S METHOD
C      FACLAM  REAL      SEE MARQUARDT'S METHOD

```

```

C      DEBUG IS A MULTY-LEVEL DEBUGGING CONTROL CHARACTER. ITS VALUE WILL
C      DETERMINE THE LEVEL OF DEBUGGING PRINTOUTS GENERATED BY THE PROGRAM
C      DURING EXECUTION. A HIGHER VALUE RESULTS IN MORE EXTENSIVE PRINTOUTS.
C      THIS IS VERY HANDY FOR PRINTING OUT INTERMEDIATE VALUES OF VARIABLES
C      NOT NORMALLY DESIRED IN THE OUTPUT OR FOR PRINTING A MESSAGE PRIOR
C      TO A SUBROUTINE CALL IN ORDER TO DETERMINE WHERE A PROGRAM IS RUNNING
C      INTO TROUBLE. A STATEMENT OF THE FOLLOWING TYPE IS RECOMENDED:

```

```

C      IF (DEBUG .GT. 2) PRINT . . . .

```

```

C      THIS PROGRAM IS SET UP FOR PROBLEMS OF UP TO 3 PARAMETERS
C      ALTHOUGH IT MAY VERY EASILY BE MODIFIED TO WORK WITH ANY
C      NUMBER OF PARAMETERS. TO DO THIS THE USER MUST CHANGE
C      THE DIMENSION OF ARRAYS BETA, BSTORE, LBBETA, UBBETA, DELB,
C      H, X, XTX, AND CHANGE.
C      ONE MUST ALSO SUPPLY ROUTINE TO CALCULATE THE DETERMINANT AND
C      INVERSE OF THE LARGER MATRIX (PXP).

```

```

C      THE USER MUST SUPPLY A SUBROUTINE NAMED "DEQMOD" IN THE
C      FOLLOWING FORMAT:

```

```

C      SUBROUTINE DEQMOD(Y1,Y2,T,DY1DT,DY2DT,BETA)
C      REAL DY1DT,DY2DT,T,Y1,Y2,BETA(3)
C      ( OPTIONAL COMMANDS: )
C      INTEGER DEBUG
C      COMMON/BLOCK9/VDEAD
C      COMMON/BLOC10/ACONFL
C      COMMON/BUG/DEBUG

```

```

C      WHERE:

```

```

C      Y1      - FIRST DEPENDENT VARIABLE (TYPICALLY V*)
C      Y2      - SECOND DEPENDENT VARIABLE (TYPICALLY NS*)
C      T       - REPRESENTS THE INDEPENDENT VARIABLE (TIME)
C      DY1DT   - DERIVITIVE OF Y1 W.R.T. TIME
C      DY2DT   - DERIVITIVE OF Y2 W.R.T. TIME
C      BETA    - ESTIMATED VALUE OF PARAMETERS

```

0

0

0

0

0

0

1

1

1

1

```

C      VDEAD - REAL VALUE INDICATING FRACTION OF CELL DEVOTED TO
C              OSMOTIC DEAD SPACE ( 0 <= VDEAD < 1.0 )
C      ACONFL - INTEGER CONTROL VARIABLE TO SET AREA CONSTANT
C              ACONFL=0 THEN A=CONSTANT
C              ACONFL=1 THEN A=F(VOLUME)
C      DEBUG - INTEGER DEBUG CONTROL VARIABLE (SEE ABOVE)
C
C      THE VALUES OF Y1, Y2, TIME, AND BETA ARE INPUT TO THE SUBROUTINE
C      AND SHOULD NOT BE ALTERED.
C      THE VARIABLES DY1DT AND DY2DT ARE THE SUBROUTINE OUTPUT.
C
C      PROGRAMED BY:      STEVE NOWLEN
C                        3/83
C
C
C
C
0001      DIMENSION BETA(3),LBBETA(3),UBBETA(3),BSTORE(3)
C              ,CHANGE(3),DELB(3),E(50),ETA(50),H(3)
C              ,PR(3,3),TIME(50),X(50,3),XTX(3,3),XTXM(3,3)
C              ,Y(50)
0002      INTEGER I,J,II,P,KOUNT,L,SFLAG,CDIFF,CFLAG
C              ,BOXFLG,BFLAG,MFLAG,DEBUG,MODNUM,ACONFL
0003      REAL PR,LBBETA,MODEL,MINVER,LAMBDA
0004      COMMON/BLOCK1/BETA,LBBETA,UBBETA,BSTORE
0005      COMMON/BLOCK2/TIME,Y,ETA,X
0006      COMMON/BLOCK3/DELB
0007      COMMON/BLOCK4/DT
0008      COMMON/BLOCK5/RADIUS
0009      COMMON/BLOCK6/CSZERO,CSSTEP,CIZERO,CISTEP,DPERMS,DPERMI,DEPTH
0010      COMMON/BLOCK8/P,CDIFF
0011      COMMON/BLOCK9/VDEAD
0012      COMMON/BLOCK10/ACONFL
0013      COMMON/BUG/DEBUG
C
C      SET ITERATION LOOP COUNTER
0014      KOUNT=0
C
C      SET CDIFF TO INITIALLY USE A FORWARD DIFFERENCE
C
0015      CDIFF=0
C      * * * * *
C
C      DATA INPUT SECTION:
C
0016      READ (2,*)DEBUG,MFLAG,BOXFLG
C
C              DEFAULT VALUE CHECKS FOR PARAMETER
C              BOUNDS VIOLATION
C      BFLAG      INTEGER      FLAGS FOR BOUNDS CHECK ON PARAMETERS
C                                1 = CHECK FOR VIOLATION
C                                0 = NO CHECK
0017      BFLAG=1
0018      READ (2,*)ACONFL,P
C

```

```

0019      READ (2,*)II
0020      DO 10 I=1,II
0021          READ (2,*)TIME(I),Y(I)
0022  10    CONTINUE
0023      READ (2,*)(BETA(I),LBBETA(I),UBBETA(I),I=1,P)
0024      READ (2,*)TOLER
      C
      C  SET TOLER2 FOR USE IN CHECKING FOR NEAR CONVERGENCE
      C  AND SWITCH TO CENTRAL DIFF DERIV.
      C
0025      TOLER2=10.*TOLER
      C
      C  READ IN NUMBER OF STEPS FOR R-K ROUTINE INTEGRATION
0026      READ (2,*)DT
      C
      C  READ IN EXPERIMENTAL CONDITIONS
0027      READ (2,*)CIZERO,CISTEP
0028      READ (2,*)CSZERO,CSSTEP
0029      READ (2,*)DPERMI,DPERMS,DEPTH
0030      READ (2,*)RADIUS
0031      READ (2,*)VDEAD
      C
      C  IF MARQUARDT METHOD READ IN LAMBDA AND DL
0032      IF(MFLAG.EQ.1)READ(2,*)LAMBDA,FACLAM
      C
      C  * * * * *
      C
      C  DATA VERIFICATION BLOCK:
      C
0034      PRINT 900
0035  900  FORMAT(1H1,' SPECIFIED RUN CONDITIONS:')
0036      PRINT 901,II
0037  901  FORMAT(' NUMBER OF DATA POINTS = ',I3)
0038      PRINT 902
0039  902  FORMAT(/,' INDEPENDENT          DEPENDENT')
0040      PRINT 903,(TIME(I),Y(I),I=1,II)
0041  903  FORMAT(2E15.6)
0042      PRINT *,' INITIAL PARAMETER ESTIMATES:'
0043      PRINT 904,(I,BETA(I),I=1,P)
0044  904  FORMAT(' BETA',I1,' = ',E15.6)
0045      IF(BFLAG.NE.1) GOTO 19
0047      PRINT 921,(LBBETA(I),I,UBBETA(I),I=1,P)
0048  921  FORMAT(E15.6,' < BETA',I1,' < ',E15.6)
0049      GOTO 17
0050  19    CONTINUE
0051      PRINT 922
0052  922  FORMAT(' NO CONSTRAINT ON PARAMETERS')
0053  17    CONTINUE
0054      PRINT 934,TOLER
0055  934  FORMAT(//,' TOLERANCE FOR CONVERGENCE = ',E10.4)
0056      PRINT 926
0057  926  FORMAT(///,' EXPERIMENTAL CONDITIONS:',//)
0058      PRINT 927,CIZERO,CSZERO
0059  927  FORMAT(' INITIAL CONCENTRATIONS:',/,
      _  ' INITIAL IMPERMEABLE SOLUTE CONCENTRATION = '

```



```

0086      CALL MODEL(II, TIME, ETA, X, BETA, UBBETA, LBBETA, 1)
C
C  CALCULATE AND STORE SUM OF SQUARES
C
0087      SQUAR=0.0
0088      DO 30 I=1, II
0089          E(I)=Y(I)-ETA(I)
0090          SQUAR=SQUAR+E(I)**2
0091  30    CONTINUE
0092      SSTORE=SQUAR
0093      PRINT 905, SQUAR
0094  905  FORMAT(/, ' SUM OF SQUARES FUNCTION FOR THESE VALUES = ', E15.6)
C
C  BECK'S EQUATION:
C
0095      DO 31 I=1, P
0096      DO 31 J=I, P
0097          XTX(I, J)=0.0
0098      DO 32 L=1, II
0099          XTX(I, J)=XTX(I, J)+X(L, I)*X(L, J)
0100  32    CONTINUE
0101          XTX(J, I)=XTX(I, J)
0102  31    CONTINUE
C
C  SET UP XTX DUMMY FOR MARQUARDT'S MODIFICATIONS
0103      DO 33 I=1, P
0104      DO 33 J=1, P
0105          XTXM(J, I)=XTX(J, I)
0106  33    CONTINUE
C
C  SET COUNTER FOR MARQUARDT'S METHOD
0107      MKOUNT=0
C
C  ***** TOP OF MARQUARDT LOOP *****
C
0108  34    CONTINUE
0109          MKOUNT=MKOUNT+1
0110          IF(MFLAG.NE.1) GOTO 36
0111          DO 35 I=1, P
0112              XTXM(I, I)=XTX(I, I)*(1.0+LAMBDA)
0113  35    CONTINUE
0114  36    CONTINUE
0115          CALL MINVER(XTXM, PR, P)
0116          DO 37 I=1, P
0117              H(I)=0.0
0118          DO 38 L=1, II
0119              H(I)=H(I)+X(L, I)*E(L)
0120  38    CONTINUE
0121  37    CONTINUE
0122          DO 39 I=1, P
0123              DELB(I)=0.0
0124          DO 40 L=1, P
0125              DELB(I)=DELB(I)+PR(I, L)*H(L)
0126  40    CONTINUE
0127  39    CONTINUE
0128

```

```

0129      DO 41 I=1,P
0130          BETA(I)=BSTORE(I)+DELB(I)
0131      41  CONTINUE
C          CHECK FOR OUT OF BOUNDS
0132      IF(BFLAG.EQ.1) CALL BOUNDS(P)
C
C *****
C
C CHECK FOR BOX-KANEMASU METHOD AND BYPASS THIS SECTION IF NOT
C
0134      IF(BOXFLG.NE.1) GOTO 50
0136          ALPHA=1.0
0137          A=1.1
0138      42  CALL MODEL(II, TIME, ETA, X, BETA, UBBETA, LBBETA, 0)
0139          SALPHA=0.0
0140          DO 43 I=1, II
0141              SALPHA=SALPHA+(Y(I)-ETA(I))**2.0
0142      43  CONTINUE
0143          IF(SALPHA.LT.SSTORE) GOTO 46
0145          IF(ALPHA.LE.0.01) GOTO 45
0147          ALPHA=ALPHA/2.0
0148          DO 44 I=1, P
0149              BETA(I)=BSTORE(I)+ALPHA*DELB(I)
0150      44  CONTINUE
0151          GOTO 42
0152      45  CONTINUE
C          ERROR CONDITION
0153          PRINT *, ' UNABLE TO REDUCE SUM OF SQUARES IN BOX METHOD'
0154          PRINT *, ' EXECUTION CONTINUING AT LAST VALUES OF ALPHA'
0155      46  CONTINUE
0156          G=0.0
0157          DO 47 I=1, P
0158              DO 47 J=1, P
0159                  G=G+DELB(J)*XTX(I, J)*DELB(J)
0160      47  CONTINUE
0161          TEST=SSTORE-ALPHA*G*(2-(1/A))
0162          HBOX=A*ALPHA
0163          IF(SALPHA.LT.TEST) GOTO 48
0165          HBOX2=G*(ALPHA**2.0)/(SSTORE-SALPHA+(2.0*ALPHA*G))
0166          IF(HBOX2.LT.HBOX) HBOX=HBOX2
0168      48  CONTINUE
0169          DO 49 I=1, P
0170              DELB(I)=HBOX*DELB(I)
0171              BETA(I)=BSTORE(I)+DELB(I)
0172      49  CONTINUE
C
C END OF BOX-KANEMASU METHOD BLOCK
C
C *****
C
0173      50  CONTINUE
C
C MARQUARDT'S METHOD BLOCK:
C
C AGAIN CHECK FOR MARQUARDT'S METHOD FLAG:

```

```

C
0174      IF(MFLAG.NE.1)GOTO 54
0176      CALL MODEL(II,TIME,ETA,X,BETA,UBBETA,LBBETA,0)
0177      SQUAR=0.0
0178      DO 51 I=1,II
0179          SQUAR=SQUAR+(Y(I)-ETA(I))**2
0180 51      CONTINUE
0181      IF(SQUAR.LT.SSTORE)GOTO 53
0183      IF(MKOUNT.GT.10)GOTO 52
0185          IF(DEBUG.GE.1)PRINT 998,FACLAM
0187 998      FORMAT(' INCREASING LAMBDA BY FACTOR OF ',E10.4)
0188          LAMBDA=LAMBDA*FACLAM
0189      GOTO 34

C
C ***** BOTTOM OF MARQUARDT LOOP *****
C
C ERROR CONDITION OUTPUT BLOCK:
C
0190 52      CONTINUE
0191          PRINT 912
0192 912      FORMAT(/,' UNABLE TO REDUCE SUM OF SQUARES WITH LAMBDA',/,
-          ' AFTER 10 ITERATIONS, CONTINUING EXECUTION')
0193 53      CONTINUE
0194          IF(DEBUG.GE.1)PRINT 999,FACLAM
0196 999      FORMAT(' REDUCING LAMBDA BY FACTOR OF ',E10.4)
0197          LAMBDA=LAMBDA/FACLAM
0198 54      CONTINUE

C
C ***** BOTTOM OF ESTIMATION BLOCK *****
C
C BEGIN CHECK FOR CONVERGENCE:
C
C IF CONSTRAINT PROBLEM CHECK PARAMETERS FOR OUT OF BOUNDS:
C
0199      IF(BFLAG.EQ.1) CALL BOUNDS(P)
C
C ITERATION COMPLETE, CHECK FOR CHANGE IN PARAMETER VALUES:
C
0201      DO 60 I=1,P
0202          CHANGE(I)=ABS(DELB(I))/(ABS(BSTORE(I))+1E-16)
0203 60      CONTINUE
C
C BE SURE CDIFF .EQ. 1 SO THAT AT LEAST ONE ITERATION USED CENTRAL
C DIFFERENCE APPROXIMATION FOR DERIVATIVES
C
0204      SFLAG=1
0205      DO 65 I=1,P
0206          IF(CDIFF.NE.1.OR.CHANGE(I).GT.TOLER)SFLAG=0
0208 65      CONTINUE
C
C CHECK FOR NEAR CONVERGENCE AND SWITCH TO CENTRAL DIFF IF TRUE
C
0209      CFLAG=1
0210      DO 66 I=1,P
0211          IF(CHANGE(I).GT.TOLER2)CFLAG=0

```

```

0213 66 CONTINUE
0214 IF(CFLAG.EQ.1)CDIFF=1
C
C OUTPUT NEW PARAMETER VALUES:
C
0216 PRINT 906,KOUNT
0217 906 FORMAT(/,' NEW PARAMETER VALUES AFTER ',I3,' ITTERATIONS:')
0218 PRINT 907,(I,BETA(I),CHANGE(I),I=1,P)
0219 907 FORMAT(' BETA',I1,' =',E15.6,10X,'% CHANGE =',E15.6)
C
C CHECK IF CONVERGENCE MET:
C
0220 80 IF(SFLAG) GOTO 500
C
C TOLERANCE NOT MET, RE-ITERATE:
C
0222 GOTO 20
C
0223 500 CONTINUE
C
C END PROGRAM SEQUENCE:
C
0224 CALL MODEL(II, TIME, ETA, X, BETA, UBBETA, LBBETA, 0)
0225 PRINT 908
0226 908 FORMAT(1H1,' FINAL ESTIMATED PARAMETER VALUES:',//)
0227 PRINT 909,(I,BETA(I),I=1,P)
0228 909 FORMAT(' BETA',I1,' =',E15.6)
0229 SQUAR=0.0
0230 DO 70 I=1,II
0231 SQUAR=SQUAR+(Y(I)-ETA(I))**2.0
0232 70 CONTINUE
0233 PRINT 935,SQUAR
0234 935 FORMAT(//,' FINAL SUM OF THE SQUARES FUNCTION = ',E15.6)
0235 PRINT 910
0236 910 FORMAT(//,' INDEPENDENT',8X,'DEPENDENT VARIABLE',/,
_4X,'VARIABLE',7X,'MEASURED',8X,'MODEL',//)
0237 PRINT 911,(TIME(I),Y(I),ETA(I),I=1,II)
0238 911 FORMAT(3E15.6)
0239 STOP
0240 END

```

[illegible]

[illegible]

```

C
C C C C C C C C C C C C C C C C C C C C C C C C C C C C C
C
0001      FUNCTION DETERM(A,N)
0002      DIMENSION A(3,3)
0003      IF(N.NE.1) GOTO 10
0005      DETERM=A(1,1)
0006      RETURN
0007  10    IF(N.NE.2) GOTO 20
0009      DETERM=A(1,1)*A(2,2)-A(2,1)*A(1,2)
0010      RETURN
0011  20    IF(N.NE.3) GOTO 30
0013      DETERM= A(1,1)*(A(2,2)*A(3,3)-A(2,3)*A(3,2))
          -      -A(1,2)*(A(2,1)*A(3,3)-A(2,3)*A(3,1))
          -      +A(1,3)*(A(2,1)*A(3,2)-A(2,2)*A(3,1))
0014      RETURN
0015  30    TYPE *, ' ERROR - MATRIX IN DETERM CALL LARGER THAN 3X3'
0016      STOP
0017      END

```

APPENDIX B

Subprogram Unit MODEL

This subprogram handles the calculation of the predicted values using the user supplied set of differential equations. It also calculates the matrix of sensitivity coefficients. In support of this routine the user must supply a subroutine called DEQMOD in the form:

```
SUBROUTINE DEQMOD(V,NS,TIME,DVDT,DNSDT,BETA)
REAL BETA(3),DNSDT,DVDT,NS,TIME,V
      OPTIONAL COMMANDS DEPENDING ON NEED:
      . INTEGER DEBUG,ACONFL
      REAL CISTEP,CIZERO,CSSTEP,CSZERO,RADIUS,VDEAD
COMMON/BUG/DEBUG
COMMON/BLOCK5/RADIUS
COMMON/BLOCK6/CSZERO,CSSTEP,CIZERO,CISTEP
COMMON/BLOCK9/VDEAD
COMMON/BLOC10/ACONFL
```

An example of the type of routine required is contained in Appendix D of the present work. As illustrated in this example the user can supply additional support routines such as the one shown which calculates the concentration of the sample region using Equation 4.1.16 of the present work. The input variable are defined in these routines.

Note that this routine can be made to use a one-directional derivative (forward or backward difference) or a central difference chosen via the variable CDIFF. In the present version the routine always uses the central difference unless the parameter is at the

value of one of the user specified bounds. If this occurs then the routine will use a one directional difference approximation in the direction towards the center of the bounded region. The statements needed to change this to a selectable option are included in the program but in the current version are nulled out via a comment indicator.

```

C
C   THIS SUBROUTINE EVALUATES THE VOLUME-TIME HISTORY FOR A GIVEN
C   SET OF PARAMETER VALUES AND EXPERIMENTAL CONDITIONS.  IT ALSO
C   EVALUATES THE MATRIX OF PARTIAL DERIVITIVES WITH RESPECT TO EACH
C   OF THE THREE PARAMETERS USING A CENTRAL DIFFERENCE APPROXIMATION
C   IF AND ONLY IF  XFLAG=1  IN THE ROUTINE CALL.
C
C   SUBTOUINE INPUT IS:
C       N           INTEGER      NUMBER OF DATA POINTS
C       TIME        REAL(50)     VECTOR OF INDEPENDENT VARIABLE VALUES
C       BETA        REAL(3)      VECTOR OF PARAMETER VALUES:
C                               DEFINED BY SUBROUTINE DEQMOD
C       XFLAG       INTEGER      CONTROL VARIABLE FOR SUPPRESSION OF
C                               DERIVATIVE CALCULATION.
C
C   SUBROUTINE OUTPUT IS:
C       ETA        REAL(50)     VECTOR OF CALCULATED DEPENDENT VARIABLE
C                               VALUES
C       X          REAL(50,3)    MATRIX OF PARTIAL DERIVITIVES
C
C   PROGRAMED BY:  STEVE NOWLEN
C                   4/ 83
C
0001   SUBROUTINE MODEL(N, TIME, ETA, X, BETA, UBBETA, LBBETA, XFLAG)
0002   DIMENSION TIME(50), ETA(50), X(50,3), BETA(3)
0003   1 , BETAD(3), DELB(3), UBBETA(3), LBBETA(3)
0004   INTEGER XFLAG, N, DEBUG, CDIFF, P
0005   REAL NS, NSZERO, LBBETA
0006   COMMON/ BUG/DEBUG
0007   COMMON/ BLOCK3/DELB
0008   COMMON/ BLOCK4/DT
0009   COMMON/ BLOCK6/CSZERO, CSSTEP, CIZERO, CISTEP, DPERMS, DPERMI, DEPTH
0010   COMMON/ BLOCK8/P, CDIFF
C
C       SET INITIAL CONDITIONS
C
0010   NSZERO=0.0
0011   IF(CSZERO.NE.0.0)NSZERO=CSZERO/CSEXT(0.0)
0013   Y1=1.0
0014   NS=NSZERO
C
C       CALCULATE PREDICTED VALUES FROM
C       USER MODEL FOR GIVEN PARAMETER VALUES
C
0015   CALL RK4(0.0, TIME(1), Y1, NS, DT, BETA)
0016   ETA(1)=Y1
0017   NN=N-1
0018   DO 10 I=1, NN
0019       CALL RK4(TIME(I), TIME(I+1), Y1, NS, DT, BETA)
0020       ETA(I+1)=Y1
0021   10 CONTINUE
C
C       CHECK IF DERIVITIVES ARE NEEDED
C
0022   IF(XFLAG.NE.1)RETURN

```

```

C
C      IF CDIFF.EQ.1 THEN CALCULATE X
C      USING CENTRAL DIFFERENCE
C      IF CDIFF.NE.1 THEN USE FORWARD
C      DIFFERENCE
C
0024      K=2
C      IF(CDIFF.EQ.1)K=2
C
C      ZERO OUT DERIVATIVE MATRIX:
C
0025      DO 15 J=1,3
0026          DO 15 I=1,N
0027              X(I,J)=0.0
0028      15  CONTINUE
C
C      SET UP DUMMY PARAMETER VECTOR
C
0029      DO 18 I=1,P
0030          BETAD(I)=BETA(I)
0031      18  CONTINUE
C
C      EVALUATE dV()/dBETA()
C
0032      DO 30 L=1,P
0033          DP=0.01*ABS(DELB(L))
0034          IF(DP.EQ.0.0)DP=0.001*BETA(L)
0036          IF(L.EQ.3.AND.DP.EQ.0.0)DP=0.001
C
C      CHECK TO SEE IF CURRENT PARAMETER
C      IS AT ONE OF PARAMETER BOUNDS AND
C      ROUTE ACCORDINGLY
C
0038          IF(BETA(L).LE.LBBETA(L))GOTO 27
0040          IF(BETA(L).GE.UBBETA(L))GOTO 26
C
C      THIS SECTION FOR FORWARD OR CENTRAL
C      DIFFERENCE CALCULATION AS SET BY CDIFF
C
0042      DO 25 J=1,K
0043          BETAD(L)=BETA(L)+DP
0044          Y1=1
0045          NS=NSZERO
0046          CALL RK4(0.0,TIME(1),Y1,NS,DT,BETAD)
0047          X(1,L)=X(1,L)+((Y1-ETA(1))/DP)/K
0048          DO 20 I=1,NN
0049              CALL RK4(TIME(I),TIME(I+1),Y1,NS,DT,BETAD)
0050              X(I+1,L)=X(I+1,L)+((Y1-ETA(I+1))/DP)/K
0051      20  CONTINUE
0052          DP=-DP
0053      25  CONTINUE
0054      GOTO 29
C
C      THIS SECTION FOR SPECIAL CASE
C      REQUIRING FORWARD OR BACKWARD

```

```

      C          DIFF IF AT PARAMETER BOUNDS
      C
0055  26      CONTINUE
      C          IF AT UPPER LIMIT USE BACK-DIFF.
0056          DP=-DP
0057  27      CONTINUE
0058          BETAD(L)=BETA(L)+DP
0059          Y1=1.0
0060          NS=NSZERO
0061          CALL RK4(0.0,TIME(1),Y1,NS,DT,BETAD)
0062          X(1,L)=X(1,L)+((Y1-ETA(1))/DP)/K
0063          DO 28 I=1,NN
0064              CALL RK4(TIME(I),TIME(I+1),Y1,NS,DT,BETAD)
0065              X(I+1,L)=X(I+1,L)+((Y1-ETA(I+1))/DP)
0066  28      CONTINUE
      C
      C          END SPECIAL SECTION
      C
0067  29      BETAD(L)=BETA(L)
0068  30      CONTINUE
0069          RETURN
0070          END

```

APPENDIX C

Subroutine RK4

Subroutine RK4 is a numerical integration routine for two simultaneous ordinary differential equations. It is based on a variation of the fourth order Runge-Kutta method developed by Gill, [27], and presented in detail by Romanelli, [28].

The user inputs the values of the lower and upper limits of integration, T1 and T2 respectively, the values of the two dependent variables at the lower limit, Y1 and Y2, and the approximate size of the time step increment to be used. The program will calculate the actual number of integration steps from T1 to T2, NSTEPS, since the user specified increment may not divide evenly into the total time (T2-T1). The routine will then evaluate the value of the dependent variables at the time T2 and place these values in Y1 and Y2 before returning to the calling routine. All other input variables are unchanged.

The routine requires an external user supplied subroutine named DEQMOD of the form:

```
SUBROUTINE DEQMOD(Y1,Y2,TIME,DY1DT,DY2DT,BETA)
```

This subroutine should evaluate the derivatives of the dependent variables, Y1 and Y2, with respect to the independent variable, TIME, based on the current values of Y1, Y2, TIME, and the vector of parameter values BETA. These values are then placed in DY1DT and DY2DT as the subroutine output. The values of Y1, Y2, TIME, and BETA should not be altered in the routine.

Subroutine RK4 contains two debugging statements which will flag the call to and successful return from the user supplied subroutine DEQMOD. The level for output of these flags is DEBUG.GE.1.

```

0001      SUBROUTINE RK4(T1,T2,Y1,Y2,DT,BETA)
C
C      THIS SUBROUTINE IS A RUNGE-KUTTA INTEGRATION ROUTINE OF THE
C      FORTH ORDER FOR TWO SIMULTANIOUS ORDINARY DIFFERENTIAL
C      EQUATIONS.
C
C      INPUTS TO THE ROUTINE ARE:
C      T1,T2      REAL      THE LIMITS OF INTEGRATION
C      Y1,Y2      REAL      THE INITIAL FUNCTION VALUES AT T1
C      DT         REAL      APPROXIMATE TIME INCREMENT (STEP SIZE)
C      BETA       REAL(3)   VECTOR OF PARAMETER VALUES
C
C      OUTPUT FROM THE ROUTINE IS:
C      Y1 AND Y2 ARE REPLACED BY THE NEW VALUES AT TIME T2
C      T1 AND T2 ARE UNCHANGED
C      DT IS UNCHANGED
C
C      EXTERNAL FUNCTION REQUIREMENTS ARE FOR A SUBROUTINE OF THE FORM:
C
C      SUBROUTINE DEQMOD(Y1,Y2,TIME,DY1DT,DY2DT)
C
C      THIS EXTERNAL SUBROUTINE EVALUATES THE DERIVITIVES OF Y1 AND Y2
C      WITH RESPECT TO TIME AND RETURNS THEM IN DY1DT AND DY2DT
C      RESPECTIVELY. THE VALUES OF Y1, Y2, AND TIME SHOULD NOT BE
C      MODIFIED IN THE ROUTINE.
C
0002      DIMENSION A(4),B(4),C(4)
0003      REAL A,B,C,DT,DY1DT,DY2DT,H,QX,Q1,Q2,X,Y1,Y2
0004      1 ,BETA(3)
0004      INTEGER FLAG,I,J,NSTEPS,DEBUG
0005      COMMON/BUG/DEBUG
0006      DATA FLAG/0/
C
C      NOTE THAT FOR SPEED OF EXECUTION IN REPETED CALLS TO RK4 THE
C      VALUES OF A, B, AND C ARE EVALUATED ONLY THE FIRST TIME THROUGH
C      THE ROUTINE. THEY WILL REMAIN UNCHANGED THERAFTER.
C
C      IF FIRST TIME THROUGH EVALUATE A, B, AND C
C
0007      IF(FLAG.NE.0)GOTO 10
0009      A(1)=0.5
0010      SR=SQRT(0.5)
0011      A(2)=1-SR
0012      A(3)=1+SR
0013      A(4)=1.0/6.0
0014      B(1)=2.0
0015      B(2)=1.0
0016      B(3)=1.0
0017      B(4)=2.0
0018      C(1)=0.5
0019      C(2)=A(2)
0020      C(3)=A(3)
0021      C(4)=0.5
0022      FLAG=1
0023      10  CONTINUE

```

```

C
C  EVALUATE ACTUAL STEP SIZE
C  APPROX STEP SIZE, DT, MAY NOT FIT INTERVAL
C  EVENLY SO MUST RECALCULATE
C
0024      NSTEPS=INT((T2-T1)/DT)
0025      H=(T2-T1)/FLOAT(NSTEPS)
0026      X=T1
0027      DO 20 I=1,NSTEPS
0028          QX=0.0
0029          Q1=0.0
0030          Q2=0.0
0031          DO 30 J=1,4
0032              IF(DEBUG.GE.1)TYPE 901
0033              CALL DEQMOD(Y1,Y2,X,DY1DT,DY2DT,BETA)
0034              IF(DEBUG.GE.1)TYPE 902
0035              X=X+H*A(J)*(1.0-B(J)*QX)
0036              QX=QX+3.0*A(J)*(1.0-B(J)*QX)-C(J)
0037              Y1=Y1+H*A(J)*(DY1DT-B(J)*Q1)
0038              Q1=Q1+3.0*A(J)*(DY1DT-B(J)*Q1)-C(J)*DY1DT
0039              Y2=Y2+H*A(J)*(DY2DT-B(J)*Q2)
0040              Q2=Q2+3.0*A(J)*(DY2DT-B(J)*Q2)-C(J)*DY2DT
0041          30 CONTINUE
0042      20 CONTINUE
0043      RETURN
0044      901 FORMAT(' CALLING DEQMOD')
0045      902 FORMAT(' BACK FROM DEQMOD')
0046      END

```


APPENDIX D

Subprogram Unit DEQMOD and CEXT

The routines described here are examples of the routines which the user must supply when installing a new transport model for use with MARBOX, MODEL, and RK4. These examples are also viable for calculation purposes. The model is that of Kedem and Katchalsky formulated for binary flow problems (see Chapter 3 Section 3 of the present work). It is a very simple problem to modify this model for the case of osmotic shrinkage only by setting $\omega_d=0$ and $\sigma=1.0$ at the top of the routine.

The support routine CEXT uses Equation 4.1.16 to calculate the concentration of the sample region based on the permeability of the dialysis membrane and the thickness of the region. CSEXT is for the permeable solute (relative to the cell or liposome membrane) and CIEXT is for the impermeable solute (again relative to the cell or liposome membrane).

Note that it is in routine DEQMOD that the definitions of each of

the parameters in the vector BETA are defined relative to the modeling equations. As far as MARBOX is concerned the parameters are not defined implicitly. The user chooses which parameter will be assigned to BETA(1) and which to BETA(2) etcetera. Thus the user must maintain consistency between the input and the modeling routine.

```

0001      SUBROUTINE DEQMOD(V,NS,TIME,DVDT,DNSDT,BETA)
      C
      C      THIS SUBROUTINE EVALUATES THE K-K SET OF PERMEABILITY
      C      EQUATIONS IN DIFFERENTIAL FORM FOR THE VALUES OF DV*/DT
      C      AND DNS*/DT WHERE THE STAR IMPLIES NONDIMENSIONAL VALUES.
      C
      C      NOTE THAT IN THIS VERSION OF THE ROUTINE THE TIME FACTOR
      C      HAS NOT BEEN NONDIMENSIONALIZED. THIS LEAVES THE EQUATIONS
      C      IN A PARTIALLY NONDIMENSIONAL FORM ONLY.
      C      THIS VERSION THEREFORE REQUIRES THE INITIAL RADIUS OF
      C      THE CELL TO BE PASSED THROUGH COMMON BLOCK 5.
      C      THIS ROUTINE ALSO CONTAINS THE FLAG FOR CONSTANT AREA
      C      ASSUMPTION
      C
      C      INPUT TO THE ROUTINE ARE THE VALUES OF N* AND V* AS WELL
      C      AS THE TIME AT WHICH THE SOLUTION IS DESIRED.
      C
      C      NOTE THAT THE TIME IS ONLY REQUIRED TO EVALUATE THE
      C      EXTERNAL SOLUTION CONCENTRATION.
      C
      C      NOTE ALSO THAT THE K-K SET USED INCLUDES THE TERMS TO
      C      HANDLE IMPERMEABLE SOLUTES AS WELL.
      C
      C      IT SHOULD ALSO BE NOTED THAT IT IS IN THIS ROUTINE THAT
      C      THE PARAMETERS ARE DEFINED. IN THIS VERSION:
      C          BETA(1) = Pw (SOLVENT PERMEABILITY) (CM**4/MOLE-SEC)
      C          BETA(2) = Ps (SOLUTE PERMEABILITY) (CM/SEC)
      C          BETA(3) = SIGMA (REFLECTION COEFFICIENT)
      C
0002      REAL BETA(3),NS,PW,PS,SIGMA
      C
      C      DEBUG IS AN INTEGER DEBUGGING VARIABLE
      C      WITH MULTY LEVEL CAPABILITY
      C
0003      INTEGER DEBUG,ACONFL
0004      COMMON/BUG/DEBUG
0005      COMMON/BLOCK5/RADIUS
0006      COMMON/BLOCK6/CSZERO,CSSTEP,CIZERO,CISTEP,DPERMS,DPERMI,DEPTH
0007      COMMON/BLOCK9/VDEAD
0008      COMMON/BLOC10/ACONFL
      C          CHECK FOR NEGITIVE VOLUME
0009      IF(V.LT.0.0)GOTO 99
      C          EVALUATE EXTERNAL CONCENTRATIONS
0011      CSOUT=CSEXT(TIME)
0012      CIOUT=CIEXT(TIME)
      C          SET UP PARAMETERS FROM BETA
0013      PW=BETA(1)*18.015*1E-4
0014      PS=BETA(2)*1E-4
0015      SIGMA=BETA(3)
      C          EVALUATE DV/DT
0016      DVDT=-(PW*3.0/RADIUS)*
      1      (SIGMA*CSOUT*(1.0-(NS/(V-VDEAD))))
      1      +CIOUT-(CIZERO*(1-VDEAD)/(V-VDEAD)))
      C          CHECK FOR NON-CONSTANT AREA
0017      IF (ACONFL.EQ.1)DVDT=DVDT*(V**(2./3.))

```

```

      C          EVALUATE DNS/DT
0019      X1=((1.0-(NS/(V-VDEAD)))*PS*3.0)/RADIUS
      C          CHECK FOR NON-CONSTANT AREA
0020      IF (ACONFL.EQ.1) X1=X1*(V**(2./3.))
0022      DNSDT=X1+(DVDT*(1.0-SIGMA)*(1.0+(NS/(V-VDEAD)))/2.0)
      C
0023      IF(DEBUG.GE.2)PRINT 903,DVDT,DNSDT,CSOUT
0025      RETURN
0026      99      CONTINUE
      C          THIS SECTION FOR NEGATIVE VOLUME
0027      PRINT 904,PW,PS,SIGMA
0028      STOP
0029      901     FORMAT(' CALLING CSEXT')
0030      902     FORMAT(' BACK FROM CSEXT')
0031      903     FORMAT(' DVDT = ',E15.7,'    DNSDT = ',E15.7,'    CSOUT = ',F10.4)
0032      904     FORMAT(' *** ERROR CONDITION IN DEQMOD ***',/,
1          ' NEGATIVE VOLUME ENCOUNTERED, PARAMETER VALUES:',/,
2          ' PW      = ',E15.6,/,', PS      = ',E15.6,/,', SIGMA = ',E15.6)
0033      END

```

[illegible]

APPENDIX E

Subprogram Units TAKEPT and CIRCLE

C THIS PROGRAM UNIT INTEGRATES A SERIES OF ARCS DEFINED
C BY THREE POINTS EACH. THE FINAL AREA IS THE SUM OF THE
C INTEGRATED PARTS DIVIDED BY TWO. THIS PROCEDURE IS
C DOCUMENTED IN CHAPTER 4 SECTION 4 OF THIS THESIS WORK.

C INPUT TO THIS ROUTINE IS THROUGH A FILE NAMED POINTS.DAT
C THIS FILE SHOULD CONTAIN IN THE FOLLOWING ORDER:

C END1X,END1Y
C END2X,END2Y
C II
C X(1),Y(1)
C X(2),Y(2)
C .
C .
C X(II),Y(II)

C WHERE:

C	END1X	INTEGER	X COORDINATE OF FIRST SCALE ENDPOINT
C	END1Y	INTEGER	Y COORDINATE OF FIRST SCALE ENDPOINT
C	END2X	INTEGER	X COORDINATE OF SECOND SCALE ENDPOINT
C	END2Y	INTEGER	Y COORDINATE OF SECOND SCALE ENDPOINT
C	II	INTEGER	NUMBER OF DATA POINTS
C	X	INTEGER	SET OF X COORDINATES
C	Y	INTEGER	SET OF Y COORDINATES

C THE PROGRAM OUTPUT IS WRITTEN INTO A DATA FILE NAMED
C POINTS.OUT THE PROGRAM WILL DESTORY ANY EXISTING FILE
C WITH THIS NAME DURING EXECUTION SO THAT ANY FILE TO BE
C SAVED MUST BE RENAMED BEFORE RUNNING CIRCLE AGAIN.

C PROGRAMED BY: STEVE NOWLEN
C 3/ 83

0001 DIMENSION X(20),Y(20),XX(3),YY(3),SUBA(3,3)
1 SUBB(3,3),SUBC(3,3),COEF(3,3),R(20),THETA(3)
0002 INTEGER I,J,II,P,KOUNT,L,SFLAG,M
 _,DEBUG
0003 COMMON/BUG/DEBUG

C * * * * *

C

C DATA INPUT SECTION:

C

0004 OPEN(UNIT=2,NAME='DK:POINTS.DAT',TYPE='OLD')
0005 READ (2,*)IEND1X,IEND1Y
0006 READ (2,*)IEND2X,IEND2Y
0007 F1=ABS(FLOAT(IEND1X-IEND2X))
0008 F2=ABS(FLOAT(IEND1Y-IEND2Y))
0009 SCALE=SQRT(F1**2+F2**2)
0010 READ (2,*)N
0011 DO 10 I=1,N

```

0012      READ (2,*)IX,IY
0013      X(I)=FLOAT(IX)
0014      Y(I)=FLOAT(IY)
0015 10    CONTINUE
0016      CLOSE(UNIT=2,DISPOSE='SAVE')
0017      OPEN(UNIT=4,NAME='DK:POINTS.OUT',TYPE='NEW')
C
C * * * * *
C
C      BEGIN CALCULATION PROCEDURE:
C
C      ***** TOP OF MAIN PROGRAM LOOP *****
C
0018      XMAX=0.0
0019      XMIN=1000.0
0020      YMAX=0.0
0021      YMIN=1000.0
0022      DO 19 I=1,N
0023          IF(X(I).GT.XMAX)XMAX=X(I)
0025          IF(X(I).LT.XMIN)XMIN=X(I)
0027          IF(Y(I).GT.YMAX)YMAX=Y(I)
0029          IF(Y(I).LT.YMIN)YMIN=Y(I)
0031 19    CONTINUE
0032      X0=(XMAX+XMIN)/2.0
0033      Y0=(YMAX+YMIN)/2.0
0034      DO 20 I=1,N
C          CONVERT TO RADIAL
0035          R(I)=SQRT((X(I)-X0)**2.+(Y(I)-Y0)**2.)
0036 20    CONTINUE
0037      AREA=0.0
0038      DO 21 I=1,N
C          ROTATE COORDS AND CALC THETA
0039          THETA(1)=ATAN((Y(I)-Y0)/(X(I)-X0))
0040          THETA(2)=ATAN((Y(I+1)-Y0)/(X(I+1)-X0))-THETA(1)
0041          THETA(2)=ATAN((Y(I+2)-Y0)/(X(I+2)-X0))-THETA(1)
0042          THETA(1)=0.0
C          CALCULATE A, B, AND C:
0043      DO 22 J=1,3
0044          COEF(J,1)=1.0
0045          COEF(J,2)=THETA(J)
0046          COEF(J,3)=THETA(J)**2.
0047 22    CONTINUE
0048      DETX=(COEF(2,2)*COEF(3,3))-(COEF(2,3)*COEF(3,2))
0049      DO 23 J=1,3
0050          SUBA(J,1)=R(I+J-1)
0051          SUBA(J,2)=COEF(J,2)
0052          SUBA(J,3)=COEF(J,3)
0053          SUBB(J,1)=COEF(J,1)
0054          SUBB(J,2)=R(I+J-1)
0055          SUBB(J,3)=COEF(J,3)
0056          SUBC(J,1)=COEF(J,1)
0057          SUBC(J,2)=COEF(J,2)
0058          SUBC(J,3)=R(I+J-1)
0059 23    CONTINUE
0060      A=DETERM(SUBA)/DETX

```



```

0061      B=DETERM(SUBB)/DETX
0062      C=DETERM(SUBC)/DETX
0063      T=THETA(3)
0064      AREA = AREA + 0.5* ( T*(A**2.) + A*B*(T**2.)
1          + (2.*A*C+(B**2.))* (T**3.)/3.
2          + B*C*(T**4.)/2. + (C**2.)*(T**5.)/5.)
0065  21  CONTINUE
      C                                END PROGRAM SEQUENCE
0066      WRITE(4,905)AREA
0067  905  FORMAT(' AREA = ',E15.8,' UNITS')
0068      PI=4.0*ATAN(1.0)
0069      RADIUS=SQRT(AREA/PI)
0070      WRITE(4,906)RADIUS
0071  906  FORMAT(' RADIUS = ',F10.4,' UNITS')
0072      WRITE(4,907)SCALE
0073  907  FORMAT(' SCALE = 1 : ',F10.4)
0074      STOP
0075      END

```

C

```
0001 FUNCTION DETERM(X)
0002 DIMENSION X(3,3)
0003 DETERM=X(1,1)*(X(2,2)*X(3,3)-X(2,3)*X(3,2))
      1 - X(1,2)*(X(2,1)*X(3,3)-X(2,3)*X(3,1))
      2 + X(1,3)*(X(2,1)*X(3,2)-X(2,2)*X(3,1))
0004 RETURN
0005 END
```

0001 PROGRAM POINTS

```

C
C   THIS PROGRAM UNIT IS LOCAL TO THE IMAGE ANALYSIS
C   LABORATORY OF MICHIGAN STATE UNIVERSITY. ITS PURPOSE
C   IS TO READ IN A SERIES OF X-Y COORDINATE POINTS. THE
C   USER SIGNIFIES THAT THE CURSOR IS PROPERLY POSITIONED
C   FOR THE NEXT POINT BY DEPRESSING THE BACK-SPACE KEY ON
C   THE CONSOLE TERMINAL. THE PROGRAM WILL THEN RING THE
C   TERMINAL BELL WHEN READY FOR ANOTHER POINT.
C
C   THIS PROGRAM MUST BE LINKED TO THE LIBRARY TVLIB.OBJ
C   ON THE SYSTEM VOLUME. THIS LIBRARY CONTAINS THE EXTERNAL
C   ROUTINE CURSOR. THE PROGRAM IS CURRENTLY SET UP TO READ
C   FIRST:
C       TWO SCALE ENDPOINTS FOR REFERENCE SCALE
C       20 DATA POINTS AROUND THE EDGE OF A CIRCULAR IMAGE
C
C   OUTPUT FROM THE ROUTINE CONSISTS OF THE DATA POINTS IN
C   INTEGER FORMAT (I5) AND THE NUMBER OF DATA POINTS (20) IN
C   PROPER FORMAT FOR DIRECT ENTRY INTO THE ROUTINE CIRCLE.FOR
C   THE OUTPUT FILE NAME IS POINTS.DAT WHICH IS THE INPUT
C   FILE NAME FOR ROUTINE CIRCLE.FOR.
C
C   PROGRAMED BY: STEVE NOWLEN
C                   319 EB
C
0002       INTEGER I,K,X,Y
0003       EXTERNAL OPEN,CURSOR
0004       OPEN(UNIT=3,NAME='DK:POINTS.DAT',TYPE='NEW')
0005       TYPE 905
0006       905 FORMAT(' ENTER SCALE END POINTS NOW')
0007       CALL CURSOR(X,Y)
0008       WRITE(3,901)X,Y
0009       901 FORMAT(2I5)
0010       CALL CURSOR(X,Y)
0011       WRITE(3,901)X,Y
0012       WRITE(3,902)
0013       902 FORMAT(' 20')
0014       TYPE 903
0015       903 FORMAT(' ENTER 20 DATA POINTS NOW')
0016       DO 10 I=1,20
0017       CALL CURSOR(X,Y)
0018       WRITE(3,901)X,Y
0019       10 CONTINUE
0020       CLOSE(UNIT=3,DISPOSE='SAVE')
0021       STOP
0022       END

```

APPENDIX F

Subprogram Unit CURGEN

This program unit is for use in generating predicted curves from specific parameter values independent of the parameter estimation routine. The routine requires the external routines RK4 and DEQMOD as described in the previous sections. The input-output format is described in the program listing.

C THIS ROUTINE IS A DRIVER FOR GENERATING SOLUTIONS
 C TO A SET OF DIFFERENTIAL EQUATIONS FOR THE TRANSPORT
 C OF MATERIALS ACROSS A CELL MEMBRANE. THE ROUTINE
 C SOLVES THE MODELING EQUATIONS FOR A TIME SOLUTION OF
 C VOLUME AND SOLUTE CONTENT.

C
 C INPUT TO THE ROUTINE IS THROUGH LOGICAL UNIT 3 SO
 C THAT UNDER RT-11 THE INPUT FILE SHOULD BE NAMED
 C "FIN3.DAT".

C INPUT IS IN THE FREE FORMAT FORM AND IN THE ORDER:

C
 C DEBUG
 C DTINT, DTOUT, TMAX
 C CIZERO, CISTEP
 C CSZERO, CSSTEP
 C DPERMI, DPERMS, DEPTH
 C PW, PS, SIGMA
 C RADIUS
 C VDEAD
 C ACONFL

C WHERE:

C	DEBUG	INTEGER	VARIABLE LEVEL DEBUG OUTPUT
C			CONTROL VARIABLE
C	DTINT	REAL	STEP SIZE FOR INTEGRATION (SEC)
C	DTOUT	REAL	INTERVAL FOR PRINTING OF
C			VOLUME AND SOLUTE CONTENT (SEC)
C	TMAX	REAL	LENGTH OF TIME TO INTEGRATE OVER (SEC)
C	CIZERO	REAL	INITIAL IMPERMEABLE SOLUTE
C			CONCENTRATION
C	CISTEP	REAL	NEW IMPERMEABLE SOLUTE CONC.
C	CSZERO	REAL	INITIAL PERMEABLE SOLUTE CONC.
C	CSSTEP	REAL	NEW PERMEABLE SOLUTE CONC.
C	***** NOTE ALL CONC. IN UNITS (MOLES/CC) *****		
C	DPERMI	REAL	DIALYSIS MEMBRANE PERMEABILITY
C			TO IMPERMEABLE SOLUTE (CM/SEC)
C	DPERMS	REAL	DIALYSIS MEMBRANE PERMEABILITY
C			TO PERMEABLE SOLUTE (CM/SEC)
C	DEPTH	REAL	DEPTH OF SMPLE REGION IN DIFFUSION
C			CHAMBER (CM)
C	PW	REAL	HYDRAULIC OR SOLVENT (WATER)
C			PERMEABILITY OF CELL (MICRONS/ SEC)
C	PS	REAL	SOLUTE PERMEABILITY OF CELL (MICRONS/ SEC)
C	SIGMA	REAL	INTERACTION TERM
C	RADIUS	REAL	INITIAL CELL RADIUS (CM)
C	VDEAD	REAL	FRACTION OF INITIAL VOLUME DEVOTED
C			OSMOTICALLY INACTIVE VOLUME
C	ACONFL	INTEGER	FLAGS FOR AREA AS FUNCTION OF VOLUME
C			ACONFL = 0 IMPLIES CONSTANT AREA
C			ACONFL = 1 IMPLIES AREA=F(V)

0001 DIMENSION BETA(3), VSTORE(50), TIME(50)
 0002 REAL NS, PW, PS, SIGMA
 0003 INTEGER DEBUG, ACONFL
 0004 COMMON/ BLOCK5/ RADIUS

```

0005      COMMON/ BLOCK6/ CSZERO, CSSTEP, CIZERO, CISTEP, DPERMS, DPERMI, DEPTH
0006      COMMON/ BLOCK9/ VDEAD
0007      COMMON/ BLOC10/ ACONFL
0008      COMMON/ BUG/ DEBUG
      C          OPEN FILE FOR OUTPUT
0009      OPEN (UNIT=6, NAME='DY1:OUTPUT.DAT', TYPE='NEW')
      C          DATA ENTRY BLOCK
0010      READ(3,*)DEBUG
0011      READ(3,*)DTINT, DTOUT, TMAX
0012      READ(3,*)CIZERO, CISTEP
0013      READ(3,*)CSZERO, CSSTEP
0014      READ(3,*)DPERMI, DPERMS, DEPTH
0015      READ(3,*)PW, PS, SIGMA
0016      READ(3,*)RADIUS
0017      READ(3,*)VDEAD
0018      READ(3,*)ACONFL
      C          ECHO OUTPUT OF EXPERIMENTAL CONCITIONS
0019      WRITE(6,905) CIZERO, CISTEP, CSZERO, CSSTEP
0020      WRITE(6,906) DPERMI, DPERMS, DEPTH
0021      WRITE(6,904) PW, PS, SIGMA
0022      WRITE(6,907) RADIUS, VDEAD
      C          BEGIN CALCULATION PROCEDURE
0023      NSTEPS=INT(TMAX/DTOUT)
0024      IF(NSTEP.GT.50) WRITE(6,911)
0026      IF(NSTEPS.GT.50) NSTEPS=50
0028      BETA(1)=PW
0029      BETA(2)=PS
0030      BETA(3)=SIGMA
0031      NS=0.0
0032      IF(CSZERO.NE.0.0) NS=CSZERO/CSEXT(0)
0034      TIME2=0.0
0035      V=1.0
0036      WRITE(6,900)
0037      WRITE(6,901) TIME2, V, NS
0038      IFLAG=0
0039      KOUNT=0
0040      DO 10 I=1, NSTEPS
0041          TIME1=TIME2
0042          TIME2=TIME1+DTOUT
0043          IF(DEBUG.GE.1) TYPE 903
0045          CALL RK4(TIME1, TIME2, V, NS, DTINT, BETA)
0046          VSTORE(I)=V
0047          TIME(I)=TIME2
0048          IF(DEBUG.GE.1) TYPE 902
0050          WRITE(6,901) TIME2, V, NS
0051          CHANGE=ABS(VSTORE(I)-VSTORE(I-1))
0052          KOUNT=KOUNT+1
0053          IF(CHANGE.LT.0.0001) IFLAG=IFLAG+1
0055          IF(IFLAG.GE.3) GOTO 15
0057      10  CONTINUE
0058      15  CONTINUE
0059      WRITE(6,908)
0060      VMIN=1.0
0061      DO 16 I=1, KOUNT
0062          IF(VSTORE(I).LT.VMIN) VMIN=VSTORE(I)

```

```

0064 16  CONTINUE
0065      DO 20 I=1,KOUNT
0066          VSTAR=(VSTORE(I)-VMIN)/(1.0-VMIN)
0067          RADRAT=(ABS(VSTORE(I)))**(1./3.)
0068          WRITE(6,909)TIME(I),VSTAR,RADRAT
0069 20  CONTINUE
0070      CLOSE(UNIT=6)
0071      STOP
0072 901  FORMAT(E15.4,10X,E15.6,10X,E15.8)
0073 900  FORMAT(/,' NOTE: NSTAR IS DEFINED AS THE NUMBER OF MOLES'
1  ,/, ' PERMEABLE SOLUTE INSIDE DIVIDED BY THE EXTERNAL'
2  ,/, ' PERMEABLE SOLUTE CONCENTRATION AT TIME t DIVIDED'
3  ,/, ' BY THE INITIAL VOLUME',/
4  ,/,5X, ' TIME (SEC)',9X, 'VOLUME(t)/VOLUME(0)',10X, 'NSTAR')
0074 902  FORMAT(' BACK FROM RK4')
0075 903  FORMAT(' CALLING RK4')
0076 904  FORMAT(/,' HYDRAULIC PERMEABILITY = ',E15.6,' MICRONS/ SEC'
1  ,/, ' SOLUTE PERMEABILITY = ',E15.6,' MICRONS/ SEC'
2  ,/, ' SIGMA = ',F6.4)
0077 905  FORMAT(' INITIAL IMPERMEABLE SOLUTE CONCENTRATION = ',E15.6,
1  ' MOLES/CC',
2  ,/, ' NEW IMPERMEABLE SOLUTE CONCENTRATION = ',E15.6
3  ' MOLES/CC',
4  ,/, ' INITIAL PERMEABLE SOLUTE CONCENTRATION = ',E15.6
5  ' MOLES/CC',
6  ,/, ' NEW PERMEABLE SOLUTE CONCENTRATION = ',E15.6
7  ' MOLES/CC')
0078 906  FORMAT(/,' DIALYSIS MEMBRANE PERMEABILITY:',/,
1  ' TO IMPERMEABLE SOLUTE = ',E15.6,' (CM/SEC)',/,
2  ' TO PERMEABLE SOLUTE = ',E15.6,' (CM/SEC)',/,
3  /, ' DEPTH OF SAMPLE REGION = ',E15.6,' (CM)')
0079 907  FORMAT(/,' INITIAL CELL RADIUS = ',E15.6,' (CM)',/,
1  ' OSMOTICALLY INACTIVE VOLUME FRACTION = ',F6.4)
0080 908  FORMAT(/,/,5X, ' TIME (SEC)',7X, '(V(t)-Vmin)/(V(0)-Vmin)',8X,
1  'DSTAR=D(t)/D(0)')
0081 909  FORMAT(2X,E15.6,8X,E15.6,8X,E15.6)
0082 911  FORMAT(/,' EXCEEDED MATRIX DIMENSIONS, WILL MAKE 50 STEPS',/,)
0083      END

```

APPENDIX G**Tabulated Osmotic Shrinkage Data****Experimental Conditions**

- 1.) All cells initially in 0.3 (osmol/kg) saline solution.
- 2.) For cells 1 and 2 the saline concentration increased to 0.5 (osmol/kg). For cells 3 and 4 the saline concentration increased to 0.8 (osmol/kg).
- 3.) Initial radius as follows:
 - cell 1: $r = 39.5$ (microns)
 - cell 2: $r = 40.4$
 - cell 3: $r = 38.0$
 - cell 4: $r = 38.5$
- 4.) Osmotic inactive volume as fraction of initial volume:
 - cell 1: 0.219
 - cell 2: 0.156
 - cell 3: 0.292
 - cell 4: 0.260
- 5.) All experiments conducted at room temperature using the diffusion chamber. Depth of sample region = 228.3 (microns). Cuprophane 80 μ m membrane used.

Time (sec)	$V^* = V/V_0$			
	cell 1	cell 2	cell 3	cell 4
0.0	1.0	1.0	1.0	1.0
10.0	1.0	0.964	0.9787	0.9703
20.0	0.9479	0.903	0.8320	0.8710
30.0	0.8747	0.847	0.7086	0.8096
40.0	0.8651	0.791	0.6416	0.7290
50.0	0.8147	0.729	0.6352	0.6250
60.0	0.8417	0.752	0.5961	0.5514
70.0	0.7933	0.758	0.5950	0.5595
80.0	0.7752	0.730	0.5604	0.5574
90.0	0.7227	0.706	0.5546	0.5615
100.0	0.7170	0.693	0.5575	0.5514
110.0	0.7342	0.677	0.5585	0.5374
120.0	0.6836	0.686	0.5575	0.5374
130.0	0.6726	0.662	0.5575	0.5314
140.0	0.6863	0.659	0.5575	0.5374
150.0	0.6836	0.662		
160.0	0.6863	0.653		
170.0	0.6781	0.658		
180.0	0.7001	0.672		
190.0	0.7001	0.667		
200.0	0.6974	0.667		

APPENDIX H

Tabulated Binary Flow Data

Experimental Conditions

- 1.) All cells initially in 0.3 (osmol/kg) saline solution. New solution also 0.3 (osmol/kg) saline content.
- 2.) New solution contains solute glycerol. For cell 5 the concentration is 0.2 (osmol/kg). For all others the concentration is 0.25 (osmol/kg).
- 3.) Initial cell radius as follows:
 - cell 5: $r = 39.3$ (microns)
 - cell 6: $r = 37.0$
 - cell 7: $r = 33.2$
 - cell 8: $r = 33.5$
- 4.) All experiments run at room temperature using the diffusion chamber. Cuprophane type 80pm membrane used.
- 5.) Depth of sample region in diffusion chamber = 228.6 (microns).
- 6.) Osmotic inactive volume unknown in all cases.

Time (sec)	$V^* = V/V_0$			
	cell 5	cell 6	cell 7	cell 8
0.0	1.0	1.0	1.0	1.0
10.0	0.9765	0.9504	0.98	1.0
20.0	0.9224	0.8106	0.96	0.852
30.0	0.8196	0.7354	0.94	0.800
40.0	0.7998	0.7064	0.90	0.735
50.0	0.7255	0.6301	0.848	0.735
60.0	0.6876	0.5991	0.83	0.719
70.0	0.6764	0.5834	0.779	0.719
80.0	0.6823	0.5941	0.729	0.704
90.0	———	0.5976	0.713	0.735
100.0	0.6571	0.6178	0.713	0.689
110.0	———	0.6025	———	———
120.0	0.6865	0.6102	0.681	0.704
140.0	0.6764	0.6267	0.697	0.712
160.0	0.7092	0.6315	0.666	0.689
180.0	———	0.6575	0.681	0.704
190.0	0.7192	———	———	———

Time (sec)	$V^* = V/V_0$			
	cell 5	cell 6	cell 7	cell 8
210.0	-----	0.6453	0.666	0.719
220.0	0.7498	-----	-----	-----
240.0	-----	0.6999	0.697	0.719
250.0	0.7865	-----	-----	-----
270.0	-----	0.7056	0.666	0.735
280.0	0.7861	-----	-----	-----
300.0	-----	-----	0.713	0.735
310.0	0.8001	-----	-----	-----
330.0	-----	0.7379	0.697	0.766
340.0	0.7873	-----	-----	-----
360.0	-----	-----	0.729	0.800
370.0	0.8360	-----	-----	-----
390.0	-----	0.7755	-----	-----
400.0	0.8686	-----	-----	-----
420.0	-----	-----	0.713	0.784
480.0	-----	-----	0.745	0.835
540.0	-----	-----	-----	0.870
600.0	-----	-----	-----	0.888

BIBLIOGRAPHY

BIBLIOGRAPHY

- 1 Jacobs, M.H., in Barron, E.S.G., The Permeability of Natural Membranes, University Press, Cambridge Mass., 1952.
- 2 Zethen, E., and Prescott, D.M., Acta Physiol. Scand., 28(1953),77.
- 3 Kedem, O., and Katchalsky, A., Biochimica Et Biophysica Acta, "Thermodynamic Analysis of the Permeability of Biological Membranes to Non-Electrolytes", 27(1958),229.
- 4 Onsager, Physic Rev., 37(1931),405; 38(1931),2265.
- 5 Levin, R.L., Membrane Biology, "Water Permeability of Yeast Cells at Sub-Zero Temperatures", 46(1979),91-124.
- 6 Staverman, A.J., Recl. Trav. Chim. Pays-Bas Belg., 70,344.
- 7 Kedem, O., and Katchalsky, A., The Journal of General Physiol., "A Physical Interpretation of the Phenomenological Coefficients of Membrane Permeability", 45(1961),143.
- 8 Johnson, J.A., and Wilson, T.A., Journal of Theoretical Biology, "Osmotic Volume Changes Induced by a Permeable Solute", 17(1967),304-311.
- 9 Katchalsky, A., and Curran, P.F., Nonequilibrium Thermodynamics in Biophysics, University Press, Cambridge Mass, 1965.
- 10 Stusnick, E., Numerical Methods for Determining Membrane Permeability Parameters, Ph.D Dissertation, State Univ. of New York at Buffalo, (unpublished) 1971.
- 11 Stusnick, E., and Hurst, R.P., Journal of Theoretical Biology, "Numerical Determination of Membrane Permeability Parameters", 37(1972),261-271.
- 12 Papanek, T.H., The Water Permeability of Human Erythrocyte in the Temperature Range +25C to -25C, Ph.D Dissertation, Mass. Inst. Tech., (unpublished) 1971.
- 13 Fenstermacher, J., and Johnson, J.A., American Journal of Physiol., 211(1966),341.
- 14 Willbrandt, W., Pflugers Arch. ges. Physiol., 254(1941),1.

- 15 Beck, J.V., and Blackwell, B., Solution Methods for Inverse Heat Conduction Problems, (unpublished as of Aug. 1983).
- 16 Beck, J.V., and Arnold, K.J., Parameter Estimation in Engineering and Science, Wiley, New York, 1977.
- 17 Bard, Y., Nonlinear Parameter Estimation, Academic Press Inc., New York, 1974.
- 18 Stewart, D.R., and Jacobs, M.H., Journal Cell. Comp. Physiol., 1(1932),83.
- 19 Ligon, R., Personal Communication, 1982.
- 20 Shabana, M., Personal Communication, 1983.
- 21 Terwilliger, T.C., and Solomon, A.K., Journal of General Physiol., 77(1981),549-570.
- 22 Lovelock, J.E., "Physical Instability of Human Red Blood Cells", Biochem. J., 60:692-696.
- 23 Mazur, P., "The Role of Intracellular Freezing in the Death of Cell Cooled at Supra-Optimal Rates", Cryobiology, 14(1977):251.
- 24 Rayleigh, J.W.S., The Theory of Sound, Dover Publ., New York, 1877 (reprinted 1945).
- 25 Dupuis, C.A., "A Preliminary Study of the Diffusion Through Cuprophane Membranes of Sodium Chloride and Sucrose", (unpublished) 1983.
- 26 Melkerson, M.N., "The Determination of Dialyzing Membranes Effects on Liposome Response Characteristics", (unpublished) 1983.
- 27 Gill, S., Proc. Cambridge Philos. Soc. 47(1951),96-108.
- 28 Romanelli, M.J., in Ralston, A., Mathematical Methods for Digital Computers, "Runge-Kutta Methods for the Solution of Ordinary Differential Equations", Wiley, New York, 1960.
- 29 Tu, Sui-Ming, Personal communication, 8/83.
- 30 Jackowski, S., Leibo, S.P., Mazur, P., The Journal of Experimental Zoology, 212(1980):329-341.