MATHEMATICAL MODELS FOR IMPEDANCE SPECTROSCOPY

By

MORGAN S. HARDING

A DISSERTATION PRESENTED TO THE GRADUATE SCHOOL
OF THE UNIVERSITY OF FLORIDA IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

UNIVERSITY OF FLORIDA

2017

To my parents, KC, Ion and Eric

ACKNOWLEDGMENTS

I thank my advisor, Professor Mark Orazem, for all his support and guidance throughout my Ph.D. I would not have made it through such a mentally demanding and intellectually challenging program without his constant mentorship.

I would also like to thank my family. They encouraged me to take STEM courses throughout my education, which is the reason I am where I am today. I would like to thank my sister, KC, for her support and light competition throughout our entire lives. I would like to thank my friends from high school, undergrad, and grad school. I would like to thank Eric for being my partner, my best friend, my rock, and my biggest supporter. I would like to thank Ion, my dog, for making my life better.

I would like to thank Dr. Vincent Vivier for guiding me in my work in his lab in the summer of 2016 in Paris, France. He is an extraordinary experimentalist and has a great sense of humor. I would like to thank Dr. Bernard Tribollet for his mentorship and help.

I would like to thank my lab mates for all their companionship and help.

TABLE OF CONTENTS

APPENDIX

LIST OF TABLES

LIST OF FIGURES

LIST OF CODES

LIST OF SYMBOLS

Roman

A constant used for the velocity expansion for far from the electrode of a rotating disk, $A = 0.92486353$

a constant used for the velocity expansion for close to the electrode of a rotating disk, $a = 0.510232618867$

B constant used for the velocity expansion for far from the electrode of a rotating disk, $B = 1.20221175$

b constant used for the velocity expansion for close to the electrode of a rotating disk, $b = -0.615922014399$

C capacitance, F/cm^2 or F (1F = 1C/V)

C_{dl} double-layer capacitance, F/cm^2 or F (1F = 1C/V)

c_{i} volumetric concentration of species i, mol/cm^3

D_{i} diffusion coefficient for species i, cm^2/s

F dimensionless radial component of velocity for laminar flow to a disk electrode

F Faradays constant, 96,487 C/equiv

f frequency, $f = \omega/2\pi$, Hz

f parameter in velocity interpolation, equation (2–31)

G dimensionless angular component of velocity for laminar flow to a disk electrode

GA Gluconic acid species in a continuous glucose monitor

G Glucose species in continuous glucose monitor

GOx Glucose oxidase enzyme, oxidized version, in a continuous glucose monitor

GOx − H2O2 Glucose oxidase complex, participating in the second enzymatic regeneration step, in a continuous glucose monitor

GOx2 Glucose oxidase enzyme, reduced version, in a continuous glucose monitor

GOx2 − GA Glucose oxidase complex, participating in the first enzymatic reaction, in a continuous glucose monitor

H	dimensionless axial component of velocity for laminar flow to a disk electrode
H	mesh size of inner region of homogeneous reaction code and of outer (GLM) region of CGM code
h	mesh size in a general finite difference expression
HH	mesh size of reaction region of homogeneous reaction code and of inner region of CGM code
HHH	mesh size of reaction region of CGM code
H2O2	Hydrogen peroxide molecule, H_2O_2, in a continuous glucose monitor
I	Current, I
ΔI	amplitude of sinusoidal current signal
j	complex number, $\sqrt{-1}$
K	dimensionless frequency associated with the geometry of a disk electrode
K	rate constant for electrochemical reaction that excludes the exponential dependence on potential
k_{b}	backward rate constant for a chemical reaction
K_{eq}	equilibrium rate constant for a chemical reaction, $K_{\mathrm{eq}} = k_{\mathrm{f}}/k_{\mathrm{b}}$
k_{f}	forward rate constant for a chemical reaction
L	inductance, H ($1H = 1\mathrm{Vs}^2/\mathrm{C}$)
N_{i}	flux of species i, $\mathrm{mol/cm^2s}$
O2	Oxygen molecule, O_2, in a continuous glucose monitor
Q	CPE coefficient, $\mathrm{s}^{\alpha}/\Omega\mathrm{cm}^2$
r	radial direction in cylindrical coordinates for rotating disk
R	universal gas constant, 8.3143 J/molK
R_{e}	electrolyte or ohmic resistance, Ω or $\Omega\mathrm{cm}^2$
R_{i}	homogeneous reaction of species i, $\mathrm{mol/cm^2/s}$
R_{t}	charge-transfer resistance, $\Omega\mathrm{cm}^2$
R	resistance, $\Omega\mathrm{cm}^2$ or Ω ($1\Omega = 1\mathrm{Vs/C}$)

Sc	Schmidt number, $\mathrm{Sc} = \nu/D_\mathrm{i}$, dimensionless
T	temperature, K
t	time, s
T	Period
V	Potential, V
v	velocity, cm/s
ΔV	amplitude of sinusoidal potential signal
y	axial coordinate in cylindrical coordinates for rotating disk
Z	impedance, Ω or $\Omega\mathrm{cm}^2$, or if noted, dimensionless
z_i	charge associated with species i
Z_D	diffusion impedance, Ω or $\Omega\mathrm{cm}^2$
Z_F	faradaic impedance, Ω or $\Omega\mathrm{cm}^2$

Greek

α	constant used for the velocity expansion for far from the electrode of a rotating disk, $\alpha = 0.88447411$
α	CPE component, dimensionless
α	parameter in velocity interpolation, equation (2–32)
α	function of the forward and backward rate of a chemical reaction, equation (2–33)
δ_N	Nernst diffusion layer thickness
δ_r	reaction layer thickness
ζ	dimensionless position, $\zeta = y\sqrt{\Omega/\nu}$
ζ_0	parameter in velocity interpolation, equation (2–32)
μ	fluid viscosity, g/cms
μ_i	mobility of species i, $\mathrm{mol/cm^2/s/J}$
ν	kinematic viscosity, $\nu = \mu/\rho, \mathrm{cm}^2/\mathrm{s}$
ξ	dimensionless position
ρ	fluid density, $\mathrm{g/cm}^3$

θ	angular direction in cylindrical coordinates for rotating disk
φ	phase difference between the potential and current
Φ	electrostatic potential, V
$\nabla\Phi$	gradient of electrostatic potential, negative electric field, V/cm
Ω	rotation speed, s^{-1}
ω	angular frequency, $\omega = 2\pi f$, s^-1

General Notation

$\mathrm{Im}\{X\}$	imaginary part of X
$\mathrm{Re}\{X\}$	real part of X
$\overline{X}$	steady-state or time-averaged part of $X(t)$

Subscripts

IJ	impinging jet
I	pertaining to current
i	pertaining to chemical species i
j	imaginary
r	real
r	pertaining to radial component
θ	pertaining to the angular component
i	pertaining to potential
y	pertaining to axial component

Abstract of Dissertation Presented to the Graduate School
of the University of Florida in Partial Fulfillment of the
Requirements for the Degree of Doctor of Philosophy

MATHEMATICAL MODELS FOR IMPEDANCE SPECTROSCOPY

By

Morgan S. Harding

May 2017

Chair: Mark E. Orazem
Major: Chemical Engineering

Diffusion or convective diffusion can influence the impedance response of a system associated with electrochemical reactions. Analysis for convective-diffusion impedance on both a rotating disk and a submerged impinging jet electrode are presented for finite Schmidt numbers. The convective-diffusion impedance simulations were performed in MATLAB, and the impedance is calculated using the oscillating concentration of the reacting species. The development of models for convective-diffusion impedance served as a foundation for the study of systems in which homogeneous reactions influence the impedance of electrochemical systems.

A mathematical model was developed for the impedance response associated with coupled homogeneous chemical and heterogeneous electrochemical reactions. The model included a homogeneous reaction in the electrolyte where species AB reacts reversibly to form A- and B+, and B+ reacts electrochemically on a rotating disk electrode to produce B. This model provides an extension to the literature by using a nonlinear expression for the homogenous reaction and unique diffusion coefficients for each species. The resulting convective-diffusion impedance had two asymmetric capacitive loops, one associated with convective-diffusion impedance, the other with the homogeneous reaction. Even though the assumption of a linear expression for the homogeneous reaction was relaxed, a modified Gerischer impedance was found to provide a good fit to the simulated data. The model was developed in FORTRAN, and a steady-state solution containing four variables

was solved followed by a solution in the frequency-domain involving eight variables. The oscillating concentration of B+ was used to obtain the impedance spectrum.

The development of a mathematical model for the impedance response of glucose oxidase electrochemical biosensors represents an extension of the model developed for a single homogenous reaction. In the biosensor, a process of enzymatic catalysis transforms glucose into hydrogen peroxide, which can be detected electrochemically. This model provides an extension to the relevant literature by considering four enzymatic reactions, two of which are nonlinear expressions, concentrations of the enzyme in an oxidized and reduced form, and the concentration of enzyme complexes formed as intermediates in the enzymatic reaction. A FORTRAN code was used to solve the steady-state equations for 12 variables which were used subsequently to solve the 24 frequency-domain equations. As before, the oscillating concentration of the electroactive species, hydrogen peroxide in this case, was used to obtain the impedance results.

CHAPTER 1
INTRODUCTION

Electrochemical impedance spectroscopy (EIS) plays an important role in electrochemistry and electrochemical engineering. Interpretation of impedance spectroscopy measurements requires a knowledge of reactions and transport properties and how these effect an electrochemical interface. EIS is nondestructive and noninvasive to the samples being tested. Since EIS is a transient technique, more details of an electrochemical system can be extracted then using steady-state methods alone. The popularity of impedance spectroscopy has grown in recent years. The number of journal articles referencing EIS is increasing dramatically and is approximately an exponential growth, as presented in Figure 1-1.

Impedance measurements are performed by inputting a small sinusoidal oscillating perturbation of potential (or current) to an electrochemical system and capturing the current (or potential) response. To obtain an impedance spectrum, the input oscillations are done over a range of frequencies. For a simulation, a typical frequency range goes from a dimensionless frequency of 10^{-5} to 10^{5}. For an experiment, a typical frequency range could be from 1 mHz to 1 GHz. A rotating disk electrode (RDE) embedded in an insulator is a common electrode to use when conducting EIS experiments. The developed fluid flow patterns in a rotating disk electrode system are well known.

EIS simulations allow interpretation of steady-state and impedance data on a level greater than can be achieved with experiments alone. It is not always practical to conduct EIS experiments. Electrochemical systems can exhibit many behaviors that overlap in EIS experimental results. Simulated EIS experiments allow these conflicting behaviors to be split up and discover their individual influence on the overall impedance results. In a rotating disk electrode, for example, the fluid flow is understood, see section 2.3.1, and the kinetics of an electrochemical system can be studied sufficiently without complicating the simulation with more dimensions. A one-dimensional model using Newman's BAND

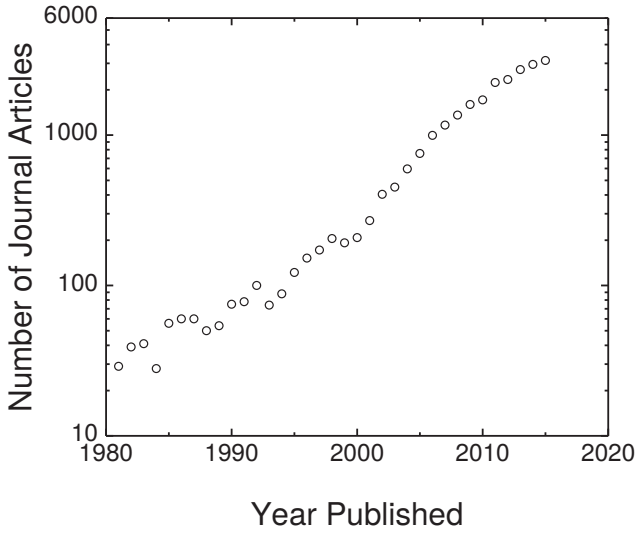


Figure 1-1. Number of journal articles that reference electrochemical impedance spectroscopy using Engineering Village® search engine. The keywords used were (impedance or admittance) and (electrochemical), journal articles only. Reprinted with permission from Orazem and Tribollet [3].

algorithm, discussed in length in Appendix A, is an ideal platform to study many electrochemical systems.

In this dissertation, fundamental concepts and numerical methods for EIS are described in Chapter 2. Background information of the convective-diffusion equation for EIS along with information for systems with convection, diffusion, and a homogeneous reaction effecting the system are also discussed. A velocity analysis for rotating disk electrodes, including an interpolation formula to describe the fluid flow over a large range is discussed. This chapter also contains history of the enzyme glucose oxidase and its use in an electrochemical sensor to measure blood glucose levels for type 1 diabetics. The current understanding of the kinetics in a glucose sensor are also discussed. Continuous glucose monitors (CGMs) are embedded bio-sensors, with a layer of glucose oxidase surrounding the electrode, that monitor blood glucose levels constantly. The use of EIS to monitor the state of health of a CGM is discussed in Chapter 2.

The results for the convective-diffusion equation for an electrochemical system with a reacting species on a rotating disk electrode[4] and on a submerged impinging jet electrode[5] have been published. Having a tabulated set of dimensionless impedance data for the three-expansion terms dependent on the Schmidt number can be incredibly useful. For example, the diffusion coefficient can be obtained after conducting an experiment and fitting to a model for the convective-diffusion equation. The mathematical development and simulation results for a finite Schmidt numbers for both the rotating disk electrode and submerged impinging jet electrode are presented in Chapter 3. The expression for an infinite Schmidt number, a simplifying assumption, and the error for assuming an infinite Schmidt number for cases where a finite Schmidt number is necessary is also discussed. The codes that were used to obtain the results in this chapter are listed in Appendices B and C.

The convective-diffusion equation is further complicated by considering a homogeneous, or chemical, reaction. Unlike when no homogeneous reaction is present, the steady-state simulation is required to obtain the impedance, so the solution becomes purely numerical. Considering the influence of a homogenous reaction on the impedance results also creates the necessity to add a velocity expansion to the system. This expansion and the interpolation between the expansion for close to the electrode are presented in more detail in section 2.3.1. The presentation of the mathematical models and results for a system effected by a faradaic and homogeneous reactions on a rotating disk electrode are presented in Chapter 4. The FORTRAN codes used to obtain the steady-state and oscillating solutions to the convective-diffusion equation with homogenous reactions as well as the MATLAB code to analyze the results are presented in Appendix D.

A mathematical model was developed for the impedance response of a glucose oxidase enzyme-based electrochemical biosensor. The model accounts for a glucose limiting membrane (GLM), which controls the amount of glucose participating in the enzymatic reaction. The glucose oxidase was assumed to be immobilized within a thin film adjacent

to the electrode. In the glucose oxidase layer, a process of enzymatic catalysis transforms the glucose into peroxide, which can be detected electrochemically. The background material and literature review related to the glucose biosensor is presented in section 2.3.3. The mathematical workup and results for the impedance response of a CGM are presented in Chapter 5. FORTRAN codes used to solve the steady and oscillating equations for the biosensor along with the MATLAB code used to plot the results are located in E.

A further parameter analysis for the impedance response of a CGM is needed to fully understand the influence of parameters on the CGM. A detailed workup of this suggested future work is presented in Chapter 7. Experiments were conducted to try to validate the coupling of faradaic and charging currents on EIS. The background to understand this phenomenon as well as simulations conducted by Wu et al. [2] and the preliminary results are presented in Chapter 7.

CHAPTER 2
BACKGROUND

Electrochemical Impedance Spectroscopy (EIS) is a popular technique to characterize electrode processes. A predominate feature of EIS is that it is non-invasive and therefor does not destroy the system. Electrochemical impedance spectroscopy simulations and models help develop a greater understanding of electrochemical systems.

2.1 Electrochemical Impedance Spectroscopy

Impedance measurements are conducted in the time domain and are performed by applying a small amplitude sinusoidal perturbation of potential to an electrochemical system and measuring the output response of current. A polarization curve is the steady-state current response as a function of potential for an electrochemical system. A point on a polarization curve is chosen to examine more and obtain EIS results. A sinusoidal perturbation about that point is imputed, for a range of frequencies, and the output is measured in order to obtain the EIS results, shown in Figure 2-1.

The input signal is represented by

$$V(t) = \overline{V} + |\triangle V|\cos(\omega t) \tag{2–1}$$

and the output as

$$I(t) = \overline{I} + |\triangle I|\cos(\omega t + \varphi) \tag{2–2}$$

The steady-state terms are $\overline{V}$ and $\overline{I}$ and the magnitude of the oscillating part of the signal is repressed by $|\triangle V|$ and $|\triangle I|$. The independent variable ω is the angular frequency, $\omega = 2\pi f$, and is commonly chosen as points per decade in impedance measurements. The phase lag between input and output is represented by φ and t is time. The calculation of the transfer function at a given frequency ω is presented schematically in Figure 2-2. An alternative way to write equations (2–1) and (2–2) is

$$V(t) = \overline{V} + \mathrm{Re}\{\widetilde{V}\exp(\mathrm{j}\omega t)\} \tag{2–3}$$

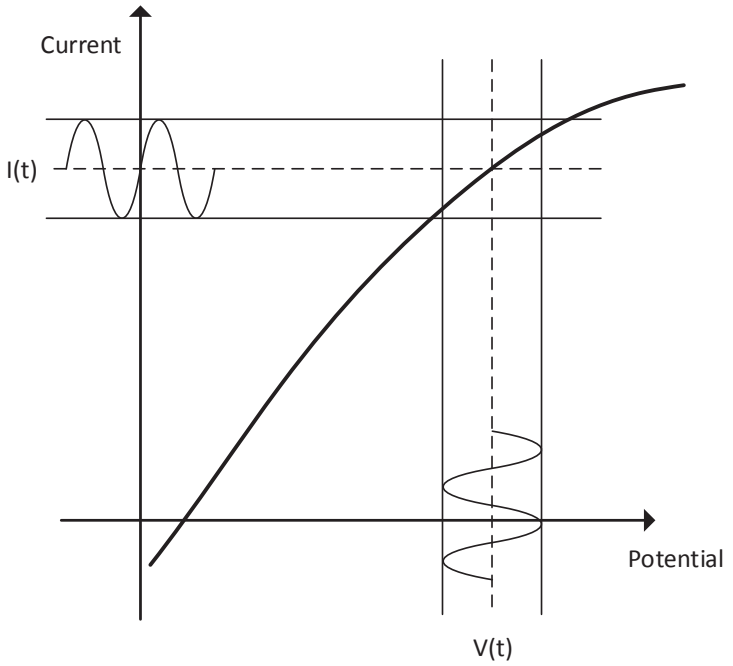


Figure 2-1. Sinusoidal perturbation of an electrochemical system at steady state, where $V(\omega)$ and $I(\omega)$ represent the potential and current, made up of a steady-state and oscillating part, at the frequency ω with a phase difference of φ [1].

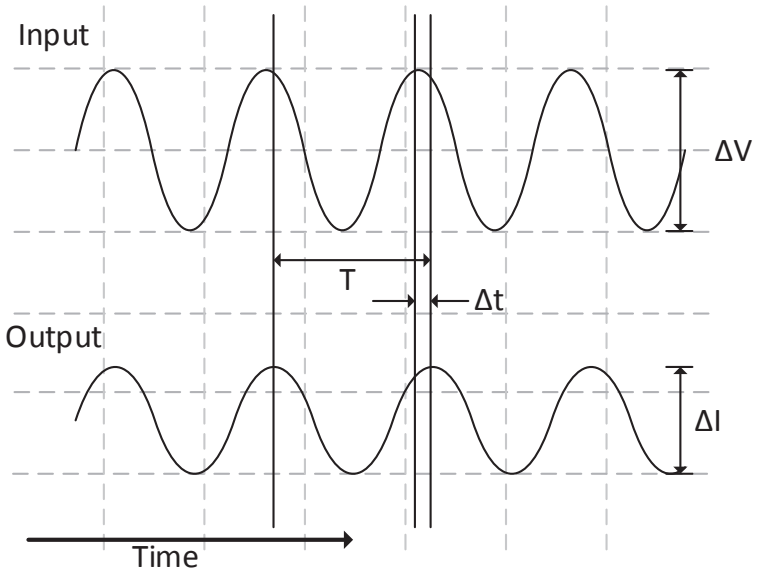


Figure 2-2. Schematic representation of the calculation of the transfer function for a sinusoidal input at frequency ω. The time lag between the two signals is $\triangle t$ and the period of the signals is T. [3]

and

$$I(t) = \overline{I} + \mathrm{Re}\{\widetilde{I}\exp(\mathrm{j}\omega t)\} \tag{2–4}$$

where $\widetilde{V}$ and $\widetilde{I}$ are complex quantities that are functions of frequency but independent of time. The complex value is represented as j, which equals $\sqrt{-1}$; j is used instead of i to avoid confusion with current density.

In EIS the oscillating magnitude is sufficiently small so the response is linear and will have the same form of the input and occur at the same frequency. There is a phase lag between the input and output, which is represented by φ. The phase lag in units of radians can be obtained as

$$\varphi(\omega) = 2\pi\frac{\triangle t}{T} \tag{2–5}$$

If $\triangle t = 0$ or if $\triangle t = T$ the phase angle is equal to zero. As shown in Figure 2-2, the output lags the input, and the phase angle has a positive value. The output response is a complex quantity, with real and imaginary components, that is a function of frequency. Impedance is therefore a complex value defined as the ratio of potential and current, see equation (2–6).

$$Z(\omega) = \frac{\widetilde{V}(\omega)}{\widetilde{I}(\omega)} = Z_{\mathrm{r}} + \mathrm{j}Z_{\mathrm{j}} \tag{2–6}$$

where $\widetilde{V}(\omega)$ and $\widetilde{I}(\omega)$ come from equations (2–3) and (2–4) and are represented as

$$\widetilde{V}(\omega) = |\triangle V|\exp(\mathrm{j}\varphi_{\mathrm{V}}) \tag{2–7}$$

and

$$\widetilde{I}(\omega) = |\triangle I|\exp(\mathrm{j}\varphi_{\mathrm{I}}) \tag{2–8}$$

where the phase difference between the potential and current is $\varphi = \varphi_{\mathrm{V}} - \varphi_{\mathrm{I}}$. In equation (2–6), Z_{r} and Z_{j} are the real and imaginary parts of the impedance, respectively.

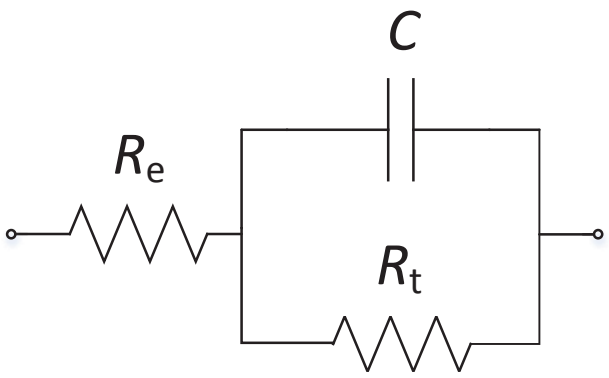


Figure 2-3. Schematic representation of circuit elements used to model EIS results.

2.1.1 Modeling Electrochemical Impedance Spectroscopy

If there is no phase difference between potential and current, then the impedance is a real number as a resistance.

$$Z_{\text{res}}(\omega) = R \tag{2–9}$$

If the phase difference lags by 90° then the impedance is completely imaginary, and a capacitance.

$$Z_{\text{cap}}(\omega) = \frac{1}{\mathrm{j}\omega C} \tag{2–10}$$

A 90° lag can also represent an inductor.

$$Z_{\text{ind}}(\omega) = \mathrm{j}\omega L \tag{2–11}$$

From this, impedance can be modeled using circuit elements. A common circuit for impedance is shown in Figure 2-3. The first resistor, R_e, is the ohmic resistance, and represents the resistance in the electrolyte. The parallel combination of a resistor and a capacitor is called a Voigt element, or RC element. The resistor in the RC element represents the charge-transfer-resistance, R_t and the capacitor represents a surface capacitance, commonly a double-layer-capacitance, C. Following the approach of using circuit elements, equations (2–9) - (2–11) can be used to represent or model impedance.

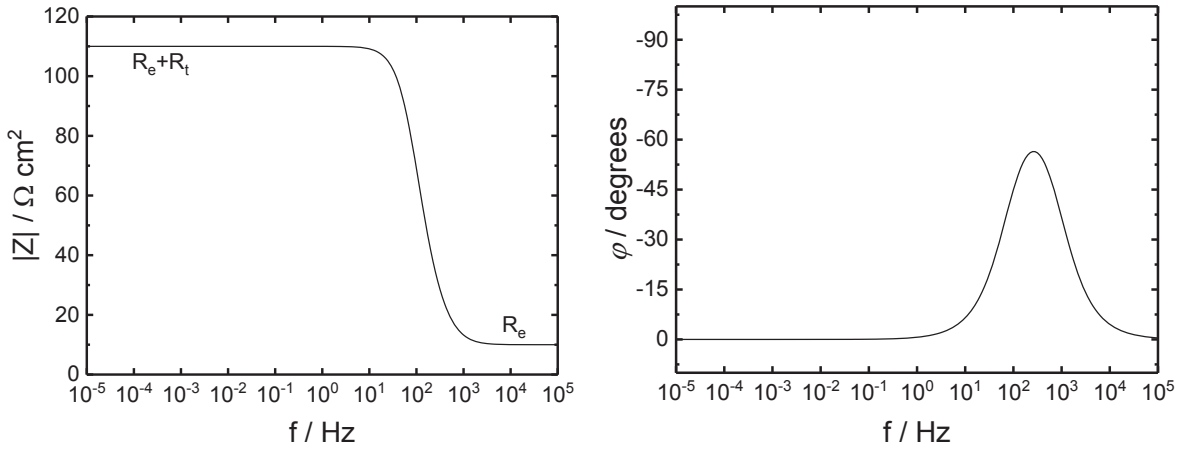


Figure 2-4. Bode representation of impedance data shown in Figure 2-6: a) magnitude and b) phase angle.

The impedance for Figure 2-3 can be represented in equation form by

$$Z(\omega) = R_{\mathrm{e}} + \frac{R_{\mathrm{t}}}{1 + j\omega C R_{\mathrm{t}}} \tag{2–12}$$

The real part of the impedance for Figure 2-3 can be represented by

$$Z_{\mathrm{r}}(\omega) = R_{\mathrm{e}} + \frac{R_{\mathrm{t}}}{1 + \omega^2 C^2 R_{\mathrm{t}}^2} \tag{2–13}$$

and the imaginary impedance by

$$Z_{\mathrm{j}}(\omega) = \frac{-j\omega C R_{\mathrm{t}}^2}{1 + \omega^2 C^2 R_{\mathrm{t}}^2} \tag{2–14}$$

2.1.2 Representation of Electrochemical Impedance Spectroscopy

Impedance is a complex value and frequently shown in a complex plane, called a Nyquist plot. The data is presented as a locus of points, where each point represents a different frequency measurement. The Nyquist plot hides the frequency dependence, hence it is advantageous to label some points with their corresponding frequency. Figure 2-6 shows a typical Nyquist plot, with the imaginary part of the impedance on the y axis and the real part of the impedance on the x axis. In order to show impedance is a function

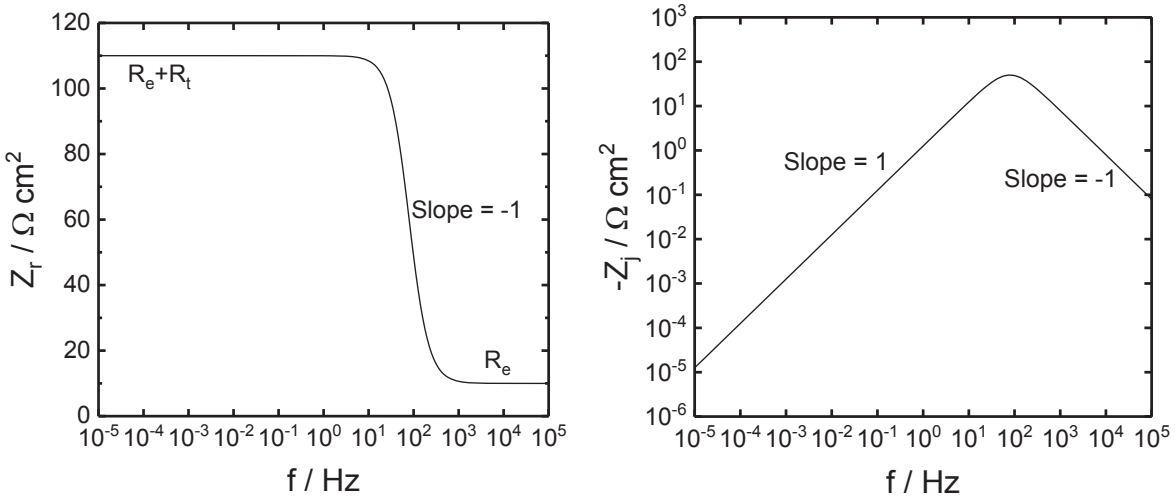


Figure 2-5. Representation of impedance data as a function of frequency for: a) real part of impedance b) imaginary part of impedance.

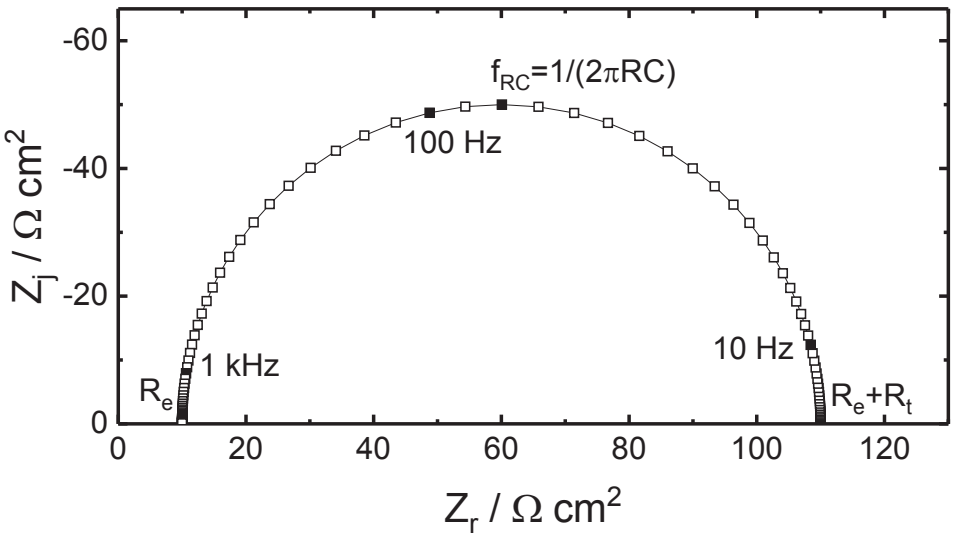


Figure 2-6. Nyquist plot of impedance data corresponding to a RC circuit and CPE circuit of $R_{\mathrm{t}} = 100\Omega\mathrm{cm}^2$, $R_{\mathrm{e}} = 10\Omega\mathrm{cm}^2$ and $C_{\mathrm{o}} = 20\mu\mathrm{F/cm}^2$

with respect to frequency a Bode plot is used. A Bode plot shows magnitude, equation (2–15), and phase angle, equation (2–16), as a function of frequency, shown in Figure 2-4.

$$|Z| = \sqrt{Z_{\mathrm{r}}^2 + Z_{\mathrm{j}}^2} \tag{2–15}$$

$$\varphi = \tanh^{-1}\left(\frac{Z_{\mathrm{j}}}{Z_{\mathrm{r}}}\right) \tag{2–16}$$

Another common way to display impedance data is the real impedance and imaginary impedance as a function of frequency, shown in Figure 2-5. The real impedance and imaginary impedance, calculated from equations (2–13) and (2–14), are presented in a log scale as a function of frequency, shown in Figures 2-5A and 2-5B, respectively.

2.1.3 Characteristic Frequency in Electrochemical Impedance Spectroscopy

The characteristic frequency of an impedance spectrum is the maximum of the absolute value of the imaginary part of the impedance. Much can be learned about a system from its characteristic frequency. For example, for the data presented in Figures 2-4-2-6 has a characteristic frequency equal to

$$f_{\mathrm{RC}} = 1/(2\pi R_{\mathrm{t}} C) \tag{2–17}$$

For a rotating disk electrode, a reacting electrochemical system influenced by convection, the dimensionless characteristic frequency is equal to 2.5. The dimensionless frequency for this system, affected by convection and diffusion, is

$$K = \frac{\omega \delta_{\mathrm{i}}^2}{D_{\mathrm{i}}} \tag{2–18}$$

where δ_{i} is the characteristic length for mass transfer and D_{i} is the diffusion coefficient of species i. The characteristic frequency can help an experimentalist estimate the capacitance of a system or measure the diffusion coefficient of a species.

2.2 Numerical Simulations for Electrochemical Impedance Spectroscopy

It is not always practical to conduct EIS experiments. Electrochemical systems can exhibit many behaviors that overlap in EIS experimental results. Simulated EIS

experiments allow these conflicting behaviors to be split up and discover their individual influence on the overall impedance results. One-dimensional models are of particular practically for electrochemical systems. In a rotating disk electrode, for example, the fluid flow is understood, see section 2.3.1, and the kinetics of an electrochemical system can be studied sufficiently without complicating the simulation with more dimensions. A one-dimensional model using Newman's BAND algorithm, discussed in length in Appendix A, is an ideal platform to study electrochemical systems.

2.2.1 Finite-Difference Methods

Differential equations describing the behavior in an electrochemical system can be solved using finite-difference methods. A differential equation can be discretized using a finite-difference method. For example a first derivative can be written as

$$\frac{\mathrm{d}c}{\mathrm{d}y} = \frac{c(x+h) - c(x-h)}{2h} + \mathcal{O}(h^2) \tag{2–19}$$

and a second derivative can be written as

$$\frac{\mathrm{d}^2c}{\mathrm{d}y^2} = \frac{c(x+h) - 2c(x) + c(x-h)}{h^2} + \mathcal{O}(h^2) \tag{2–20}$$

where the accuracy of the discretization is the order of the mesh squared.

2.2.2 Convergence Methods

Finite-difference methods in conjunction with Newman's BAND method are used in Chapters 4 and 5. A technique to help the convergence of the code was first developed by Orazem [6]. The technique is referred to as "BIGs". After the differential equation has been discretized, the biggest term in the equation is multiplied by EBIG=1×10^{-10} and is compared to G(I). If the absolute value of G(I) is less then the absolute value of the biggest term multiplied by EBIG, then G(I) is set to zero. The non-linear code converges easier with the implementation of BIGs.

2.3 Impedance Models for Rotating Disk Electrode

Due to the popularity of the rotating disk electrode, the impedance response for electrochemical systems with a rotating disk electrode has been examined by many in the electrochemical community [7, 8, 9, 10, 11, 12, 13, 14, 15].

Homsy and Newman [16] investigated an asympototic solution for the Warburg Impedance.

2.3.1 Fluid flow for a Rotating Disk

The steady flow created by an infinite disk rotating at a constant angular velocity in a fluid with constant physical properties was first studied by von Kármán. [17] The solution was sought by using a separation of variables using a dimensionless distance

$$\zeta = y\sqrt{\Omega/\nu} \tag{2–21}$$

and dimensionless radial velocity

$$v_r = r\Omega F(\zeta) \tag{2–22}$$

dimensionless angular velocity

$$v_\theta = r\Omega G(\zeta) \tag{2–23}$$

and dimensionless axial velocity

$$v_y = \sqrt{\nu\Omega}H(\zeta) \tag{2–24}$$

where ν is the kinematic viscosity in cm^2/s. Ω is the rotation speed in rad/s. y represents the axial distance from the electrode. For the system illustrated here a value of ν=0.01 cm^2/s and Ω=209.4 rad/s which corresponds to 2000 RPM were used.

The Navier-Stokes equations can be solved numerically when equations (2–22), (2–23), and (2–24) are inserted. As shown by Cochran [18], the variables F, G and H can be written as two sets of series expansions, one close to the electrode ($\zeta \leftarrow 0$), and one where $\zeta \rightarrow \infty$. For small values of ζ, or close to the electrode, the expansions are

$$F = a\zeta - \frac{1}{2}\zeta^2 - \frac{1}{3}b\zeta^3 + ... \tag{2–25}$$

$$G = 1 + b\zeta + \frac{1}{3}\zeta^3 + ... \tag{2–26}$$

and

$$H = -a\zeta^2 + \frac{1}{3}\zeta^3 + \frac{b}{6}\zeta^4 + ... \tag{2–27}$$

where a=0.5102326189 and b=-0.6159220144. A graph of F, G and H close to the electrode versus dimensionless distance is shown in Figure 2-7A.

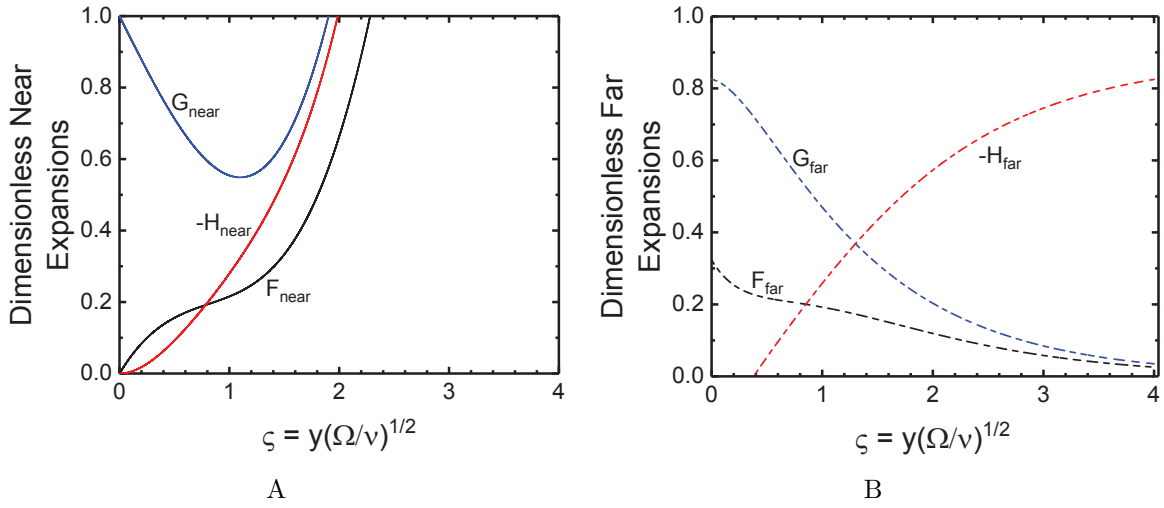


Figure 2-7. Velocity expansion for fluid flow of a rotating disk electrode for a) small values of ζ and b) large values of ζ

Far from the electrode, when ζ is large, the expansion equations become

$$F = A\exp(-\alpha\zeta) - \frac{A^2 + B^2}{\alpha^2}\exp(-2\alpha\zeta) + \frac{A(A^2 + B^2)}{4\alpha^4}\exp(-3\alpha\zeta) + ... \tag{2–28}$$

$$G = B\exp(-\alpha\zeta) - \frac{A(A^2 + B^2)}{12\alpha^4}\exp(-3\alpha\zeta) + ... \tag{2–29}$$

$$H = -\alpha + \frac{2A}{\alpha}\exp(-\alpha\zeta) - \frac{A^2 + B^2}{2\alpha^3}\exp(-2\alpha\zeta) - \frac{A(A^2 + B^2)}{6\alpha^5}\exp(-3\alpha\zeta) + ... \tag{2–30}$$

where $\alpha = 0.88447441$, $A = 0.93486353$, and $B = 1.2021175$. The expansion for when ζ is large for each species are plotted versus dimensionless position in 2-7B.

To solve the mass charge conservation equations, a velocity profile is required to describe the fluid flow in the whole domain. A weighting function, shown in equation 2–32, and an interpolation function, equation 2–31, are used to accomplish this, first developed

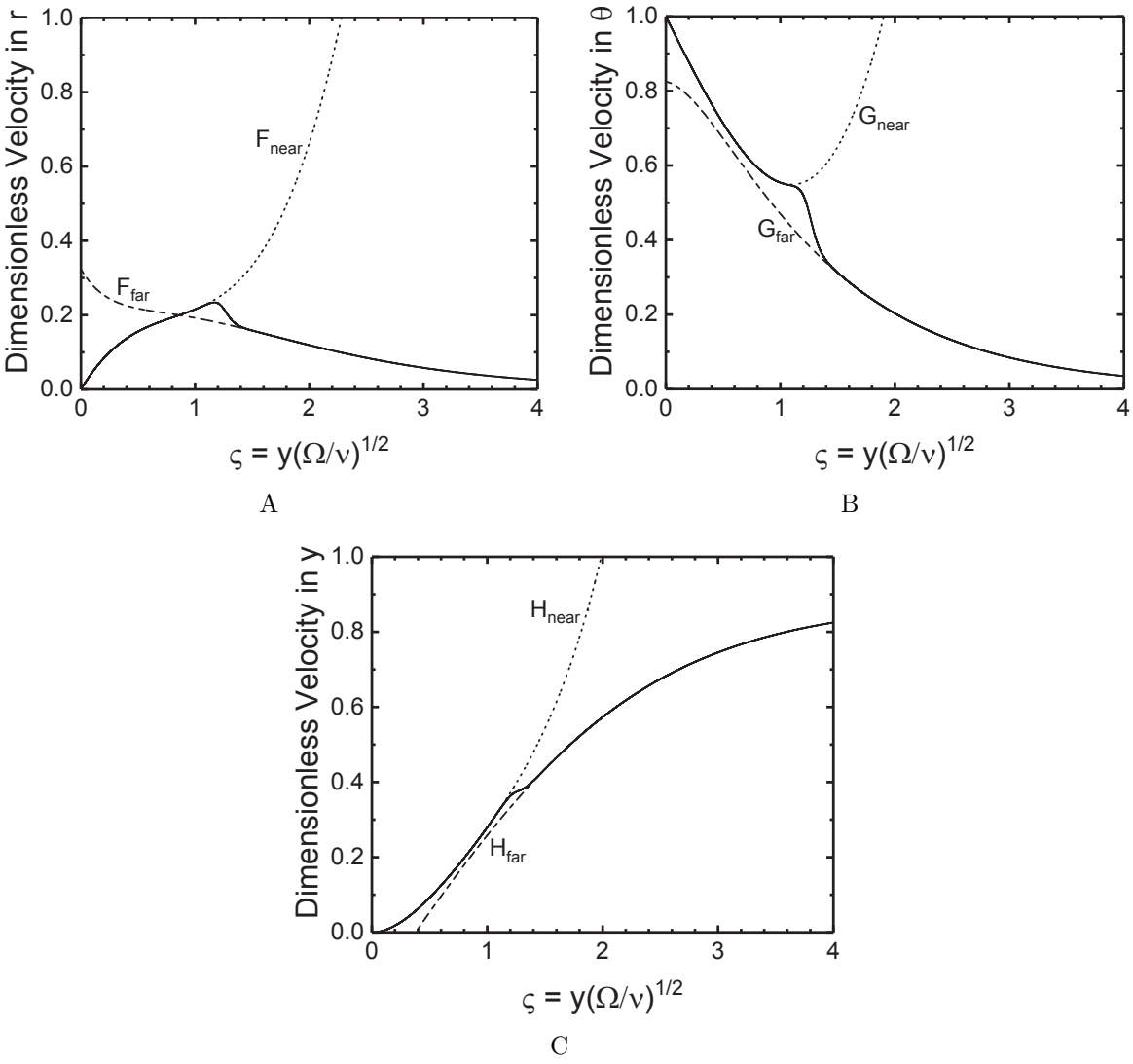


Figure 2-8. Velocity expansions and interpolation velocity for a rotating disk electrode for small values of ζ and large values of ζ for a) the r–direction, b) the θ–direction, and c) the y–direction

by Wu [1]. This interpolation function is similar to the Fermi-Dirac function applied in quantum mechanics for describing the distribution of fermions.

$$v_{\mathrm{i}} = (1-f)v_{\mathrm{i},\zeta\to 0} + f v_{\mathrm{i},\zeta\to\infty} \tag{2–31}$$

where i, can be r, y, or θ to represent the direction of the velocity flow and

$$f = \frac{1}{1+\mathrm{e}^{-\alpha(\zeta-\zeta_0)}} \tag{2–32}$$

In the present calculation, we used $\alpha = 20$ and $\zeta_0 = 1.25$ as the interpolation constants. The axial components of the dimensionless velocity as a function of dimensionless distance from the electrode surface are shown in figure 2-8C. The radial and angular dimensionless velocities are shown in figures 2-8B and 2-8A.The velocity expression applying the interpolation function satisfies the velocities for small and large values of ζ and shows a transition in the medium distance from the disk surface. The interpolation function needs to be as accurate as possible near the electrode and this creates a non–smooth transition from the near expansion to the far expansion. This non–smooth transition is not significant because the impedance depends only on the results very close to the electrode. Therefore, the use of the velocity profile in equation 2–32 on a RDE is justified.

2.3.2 Convective Diffusion Impedance Models

A one-dimensional numerical model for the impedance of a rotating disk with a finite Schmidt number was presented by Tribollet and Newman [4]. They used a two term expansion in the axial direction to describe the motion in the system.

2.3.3 Impedance Models with Homogeneous Reactions

The impedance response at an electrode surface may be influenced by homogeneous reactions in the electrolyte. In 1951, Gerischer [19] published the first formal treatment of steady-state and AC chemical-electrochemical reactions at an inert electrode in an aqueous electrolyte solution. He considered a homogeneous reaction in the electrolyte influenced only by two species and derived equations for the steady-state concentrations and the AC impedance response. Gerischer's treatment of homogeneous reactions set a standard for mathematical models and analytic solutions to problems with an electrochemical reaction influenced by a chemical reaction. Lasia [20], in his book, steps through the mathematical development of the Gerischer impedance. In 1957, Koutecky and Levich [21, 22, 23] developed a mathematical approach of kinetic and catalytic electrochemical processes on a rotating disk electrode. They found a homogeneous reaction length, what they called the

thickness of the kinetic layer, could be represented by

$$\delta_r = \sqrt{\frac{D}{\alpha}} \tag{2–33}$$

where α is a function of the forward and backward rates of the homogeneous reaction, and D is the diffusion coefficient. Smith [24, 25] used AC Polarography to study different homogeneous reaction mechanisms, including first-order preceding, following, and catalytic chemical reactions coupled with electrochemical reactions. Unlike Gerischer or Smith, Jurczakowski and Polczynski [26] developed an AC model with coupled homogenous and heterogeneous reactions accounting for cases where diffusion coefficients are not considered to be equal. The above Electrochemical-Chemical theories assume simplified homogeneous reactions, with a maximum of two species considered. Timmer et al. [27] claimed that explicit solutions could only be obtained for systems controlled by charge transfer, diffusion, and chemical reactions, if one was neglected or the chemical reactions were simplified.

Since Gerischer first developed the mathematical treatment of chemical-electrochemical reactions, many scientists have contributed to this field. Tribollet and Newman[28] calculated the distribution of concentration of each species in a concentrated solution with homogeneous and heterogeneous reactions. The velocity Tribollet and Newman used was a mass-averaged velocity. They compared the difference between the impedance derived from the infinite dilution theory and the impedance from the concentrated solution theory. Hauser and Newman [29, 30] discussed the influence of homogenous consumption of cuprous ion on the impedance response associated with dissolution of a copper rotating disk electrode. They considered a simplified homogeneous reaction where only two species are considered, one of which is the electroactive species. Bossche et al. [31] investigated the steady-state solution for an electrochemical system controlled by diffusion, migration and convection and homogeneous reactions. The convection term uses a three term expansion for close to the electrode surface calculated analytically by

von Kármán [17]. Deslouis et al. [32] used a submerged impinging jet cell to measure interfacial pH during the reduction of dissolved oxygen. Remita et al. [33] have shown that, for a deaerated aqueous electrolyte containing dissolved carbon dioxide, hydrogen evolution is enhanced by the homogeneous dissociation of CO_2. Vazquez-Arenas and Pritzker[34, 35] studied the deposition of cobalt ions on a cobalt rotating disk. They used a model that included convection and diffusion and statistically fit it to experimental EIS data. Chapman and Antaño[36] studied the effect of an irreversible homogeneous reaction on finite-layer diffusion impedance and showed a mass–transfer–boundary layer adjacent to the electrode contributing to the measured AC impedance. Tran et al. [37] demonstrated that homogeneous dissociation of acetic acid enhances cathodic reduction of hydronium ions.

A Gerischer type impedance has been used to fit many electrode processes, including solid oxide fuel cell systems [38, 39, 40], oxide electrode systems [41], mixed conducting solid electrolyte systems [42], systems with boundary conditions on a disordered boundary [43] and electrocatalytic systems influenced by the hydrogen evolution reaction [44]. Coupled electrochemical and enzymatic homogeneous reactions are also involved in sensors used to monitor glucose concentrations for management of diabetes [45, 46, 47].

2.4 Impedance Models for Continuous Glucose Monitors

Müller [48] discovered the enzyme notatin from Aspergillus niger and Penicillium glaucum in 1928 and named it glucose oxidase. The name notatin stuck and can still be referenced in medical journals [49, 50] but recent publications all refer to the enzyme as glucose oxidase. He showed that glucose was oxidized by glucose oxidase and created a product of gluconic acid. Franke and Lorenz [51] found that hydrogen peroxide was also a product of the oxidation of glucose by glucose oxidase as well as suggested that the enzyme glucose oxidase contained a flavoprotein. The glucose oxidase enzyme catalyzes the oxidation of D-glucose to D-glucono-δ-lactone which changes by a non-enzymatic reaction to gluconic acid with the presence of water [52]. In the 1960's a big scientific push

was made to understand the kinetics of glucose oxidase. Nakamura and Ogura [53] looked at the action of glucose oxidase, namely that the product D-glucono-δ-lactone can act as an inhibitor. Glucose was found to react much faster then other sugars in the presence of glucose oxidase [54]. Nakamura and Ogura and Bright and Gibson [55, 56] believed part of the oxidation of glucose by gluconic acid was partially reversible. An overview of Glucose Oxidase and the use of glucose oxidase in food science related areas was published by Banker [57].

The mechanism used in this dissertation was

$$\mathrm{G} + \mathrm{GOx} \underset{k_{\mathrm{b1}}}{\overset{k_{\mathrm{f1}}}{\rightleftarrows}} \mathrm{GOx2} - \mathrm{GA} \xrightarrow{k_{\mathrm{f2}}} \mathrm{GA} + \mathrm{GOx2} \tag{2–34}$$

$$\mathrm{GOx2} + \mathrm{O_2} \underset{k_{\mathrm{b3}}}{\overset{k_{\mathrm{f3}}}{\rightleftarrows}} \mathrm{GOx} - \mathrm{H_2O_2} \xrightarrow{k_{\mathrm{f4}}} \mathrm{GOx} + \mathrm{H_2O_2} \tag{2–35}$$

where G represents glucose, GOx is the oxidized form of glucose oxidase, GOx2-GA is the first enzyme complex that is created by glucose and GOx, GOx2 is the reduced form of glucose oxidase, GA is gluconic acid, O2 is oxygen, GOx-H2O2 is the second enzyme complex formed, and H2O2 is hydrogen peroxide. Enzymatic reactions, from reactant to product, is commonly thought of as an irreversible process. By explicitly considering the enzyme complex we can say that the reaction from reactant to product is irreversible but with a reversible first step.

With the revolution of coating electrodes with enzymes [46, 58, 59] it became possible to coat an electrode with glucose oxidase and then measure the product of H2O2 electrochemically and thus measure blood glucose levels. To further the understanding of an enzyme coated electrode the kinetics of immobilized glucose oxidase [60] and diffusion coefficients in hydrogels for oxygen, hydrogen peroxide and glucose [61] were studied by van Stroe-Biezen et al. . Temperature dependence of diffusion coefficients of oxygen

in water were studied by Han and Bartels [62]. Some studies look at a direct electrical connection to the glucose oxidase enzyme [47, 63].

Since 1974, the artificial pancreas has been acknowledged by the greater scientific community to be a possible way to manage type 1 diabetes [64]. In 2013 and 2008, case studies showed that a subcutaneous continuous glucose monitor sensors could be implanted for 14 days[65] and 28 days[66] in patients with type 1 diabetes. However, commercially available subcutaneous continuous glucose monitors are generally changed every seven days to avoid unreliable results. Some reasons for failure are sensor analysis failure [67], cell based metabolic barriers like macrophages [68],and red blood cell clots or "metabolic sinks" [69, 70].

The potential of impedance spectroscopy to monitor the state of health of subcutaneous continuous glucose monitor sensors has not been fully explored and now it is suggested that this is a plausible way to see if the CGM is still working. The status of implanted biological sensors has been studied using EIS [71, 72, 73].

CHAPTER 3
INFLUENCE OF FINITE SCHMIDT NUMBER ON THE IMPEDANCE OF DISK ELECTRODES

The rotating disk and submerged impinging jet electrodes are commonly used in electrochemical experiments. In electrochemical systems where there is no convective flow, temperature and concentration gradients can create a natural convection, which is hard to characterize. Characterizing the fluid flow in an electrochemical system has advantages, such as theory and mathematical models are easier to work with and simulations of the system are much easier then with natural convection. Simulations of a rotating disk and a submerged impinging jet use a convective-diffusion impedance dependent on the Schmidt number, $\mathrm{Sc} = \nu/D_\mathrm{i}$. When the Schmidt number is considered infinite the problem can be solved analytically. This chapter explores cases when the Schmidt number is equal to infinity as well as a finite value for both systems.

3.1 Electrochemical Mathematical Equations

The flux density of each species in an infinitely dilute electrochemical system is given by

$$\mathbf{N}_\mathrm{i} = -z_\mathrm{i} u_\mathrm{i} \mathrm{F} c_\mathrm{i} \nabla\Phi - D_\mathrm{i} \nabla c_\mathrm{i} + c_\mathrm{i}\mathbf{v} \tag{3–1}$$

and has units of mol/(cm^{2}s)[74]. The first term in equation (3–1) represents migration of the species i, which can occur if an electric field is present and if the species is charged. The second term represents the diffusion of the species. It accounts for deviations from the average velocity if diffusion is due to a concentration gradient, ∇c_i and D_i is the diffusion coefficient of species i. The last term, convection, is due to the bulk velocity, v. In equation (3–1), F is Faraday's constant with a value of 96,487 C/equiv, Φ is the electrostatic potential where the gradient is the negative electric field, and z_i represents the number of proton charges carried by an ion. The mobility of the species, u_i, is related to the diffusion coefficient using the Nernst-Einstein equation

$$D_\mathrm{i} = \mathrm{RT} u_\mathrm{i} \tag{3–2}$$

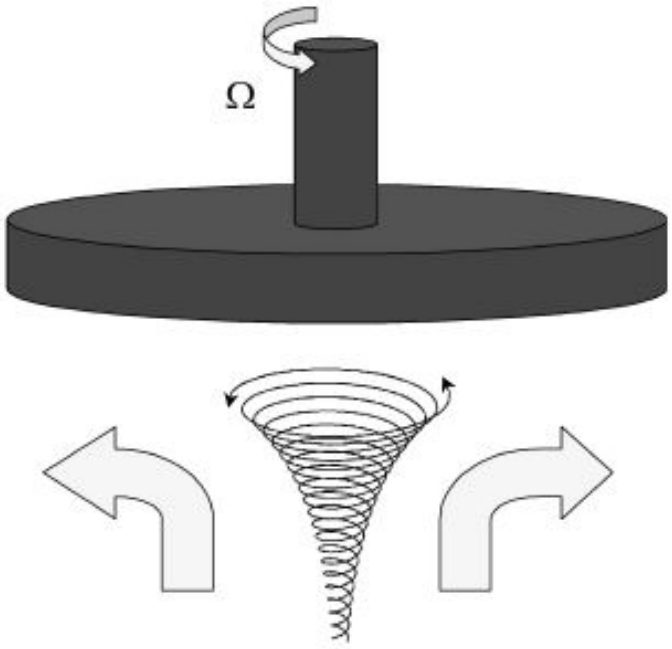


Figure 3-1. Rotating disk electrode flow pattern showing a three-dimensional flow trajectory leading to a net flow towards the disk and in the radial direction [3].

where R is the universal gas constant equal to 8.3143 J/mol K and T is temperature in Kelvin. Substitution of the mobility into equation (3–1) gives

$$\mathbf{N}_\mathrm{i} = D_\mathrm{i}\left(-z_\mathrm{i}\frac{\mathrm{F}}{\mathrm{RT}}c_\mathrm{i}\nabla\Phi - \nabla c_\mathrm{i}\right) + c_\mathrm{i}\mathbf{v} \tag{3–3}$$

A material balance is needed to obtain the mass flux at the interface

$$\frac{\partial c_\mathrm{i}}{\partial t} = -\nabla\cdot\mathbf{N}_\mathrm{i} + R_\mathrm{i} \tag{3–4}$$

where R_i is the homogeneous reaction. Each species in the system will have their own form of equation (3–4).

3.2 Rotating Disk Electrode

A schematic of the flow in a rotating disk electrode system is shown in Figure 3-1. The fluid is spiraled toward the electrode and a net velocity occurs perpendicular to the electrode and in the radial direction.

3.2.1 Fluid Velocity Profile for Rotating Disk Electrode

Following section (2.3.1) the fluid flow for a rotating disk electrode can be described by a velocity expansion in the axial direction, see equation (2–27). Without a homogeneous reaction occurring in an RDE system there is no need for a velocity expansion that goes further then that described by equation (2–27). This means the velocity profile for the system is described by the combination of equation (2–24), the axial velocity and equation (2–27), the series expansion for the axial direction for small values of ζ, to obtain

$$v_y = \sqrt{\nu\Omega}(-a\zeta^2 + \frac{1}{3}\zeta^3 + \frac{b}{6}\zeta^4 + ...) \tag{3–5}$$

3.2.2 Mathematical Development for Rotating Disk Electrode

Electrochemical impedance spectroscopy involves the perturbation of an electrochemical system with a small sinusoidal signal and then recording the output. All oscillating quantities, such as concentration and potential, can be written in the form

$$X = \bar{X} + \mathrm{Re}\left\{\tilde{X}\exp\left(\mathrm{j}\omega t\right)\right\} \tag{3–6}$$

where the over-bar represents the steady-state value, j is the imaginary number, ω is the angular frequency and the tilde represents a complex oscillating component.

The governing equation for mass transfer of a rotating disk in the absence of homogeneous reaction becomes

$$\frac{\partial c_{\mathrm{i}}}{\partial t} = \nabla \cdot \mathbf{N}_{\mathrm{i}} \tag{3–7}$$

The divergence operator of $\mathbf{N}_{\mathrm{i}}$ in cylindrical form is

$$\nabla \cdot \mathbf{N}_{\mathrm{i}} = \frac{1}{r}\frac{\partial(r\mathbf{N}_{\mathrm{r,i}})}{\partial r} + \frac{1}{r}\frac{\partial(\mathbf{N}_{\theta,\mathrm{i}})}{\partial \theta} + \frac{\partial(\mathbf{N}_{\mathrm{y,i}})}{\partial y} \tag{3–8}$$

where r is the radial component, θ is the angular component, and y is the axial component.

When applying equation (3–8) to (3–7) and assuming antisymmetric flow the governing equation becomes

$$\frac{\partial c_{\mathrm{i}}}{\partial t} = \frac{1}{r}\frac{\partial (r\mathbf{N}_{\mathrm{r,i}})}{\partial r} + \frac{\partial (\mathbf{N}_{\mathrm{y,i}})}{\partial y} \tag{3–9}$$

Flux in the y direction, excluding migration, is

$$\mathbf{N}_{\mathrm{y,i}} = -D_{\mathrm{i}}\frac{\partial c_{\mathrm{i}}}{\partial y} + c_{\mathrm{i}} v_{\mathrm{y}} \tag{3–10}$$

and flux in the r direction, excluding migration, is

$$\mathbf{N}_{\mathrm{r,i}} = c_{\mathrm{i}} v_{\mathrm{r}} \tag{3–11}$$

When substituting the flux expressions, equations (3–11)and (3–10), for the r and y directions respectively, into (3–9) the governing equation is now

$$\frac{\partial c_{\mathrm{i}}}{\partial t} = -\frac{1}{r}\frac{\partial}{\partial r}(r c_{\mathrm{i}} v_{\mathrm{r}}) - \frac{\partial}{\partial y}(-D_{\mathrm{i}}\frac{\partial c_{\mathrm{i}}}{\partial y} + c_{\mathrm{i}} v_{\mathrm{y}}) \tag{3–12}$$

After calculating the derivatives in (3–12) and inserting the first term of the velocity expansion for $\mathbf{v}_y$

$$v_{\mathrm{y}} = -ay\frac{\Omega^{3/2}}{\nu^{1/2}} \tag{3–13}$$

and $\mathbf{v}_r$

$$v_{\mathrm{r}} = ray\frac{\Omega^{3/2}}{\nu^{1/2}} \tag{3–14}$$

where $a = 0.5102326189$, Ω is the rotation speed and ν is the kinematic viscosity, the governing equation becomes

$$\frac{\partial c_{\mathrm{i}}}{\partial t} = \frac{-c_{\mathrm{i}}}{r}\frac{2ra\Omega^{3/2}}{\nu^{1/2}}y + c_{\mathrm{i}}\frac{2a\Omega^{3/2}}{\nu^{1/2}}y - \mathbf{v}_y\frac{\partial c_{\mathrm{i}}}{\partial y} + D_{\mathrm{i}}\frac{\partial^2 c_{\mathrm{i}}}{\partial y^2} \tag{3–15}$$

The first two terms in (3–15) cancel and the final governing equation for convective diffusion becomes

$$\frac{\partial c_{\mathrm{i}}}{\partial t} = -v_{\mathrm{y}}\frac{\partial c_{\mathrm{i}}}{\partial y} + D_{\mathrm{i}}\frac{\partial^2 c_{\mathrm{i}}}{\partial y^2} \tag{3–16}$$

If the expansions used more then the first term they would also cause cancelation and the final governing equation for convective diffusion remains the same.

In order to obtain impedance the oscillating concentrations of the reacting species in an electrochemical system is needed. When applying equation (3–6) to (3–16) we get an equation with separable steady-state and oscillating variables

$$\mathrm{j}\omega\tilde{c}_\mathrm{i}\mathrm{e}^{\mathrm{j}\omega t} + v_\mathrm{y}\frac{\mathrm{d}\bar{c}_\mathrm{i}}{\mathrm{d}y} + v_\mathrm{y}\frac{\mathrm{d}\tilde{c}_\mathrm{i}}{\mathrm{d}y}\mathrm{e}^{\mathrm{j}\omega t} - D_\mathrm{i}\frac{\mathrm{d}^2\bar{c}_\mathrm{i}}{\mathrm{d}t} - D_\mathrm{i}\frac{\mathrm{d}^2\tilde{c}_\mathrm{i}}{\mathrm{d}t}\mathrm{e}^{\mathrm{j}\omega t} = 0 \tag{3–17}$$

The equation for the steady-state variables is

$$v_\mathrm{y}\frac{\mathrm{d}\bar{c}_\mathrm{i}}{\mathrm{d}y} - D_\mathrm{i}\frac{\mathrm{d}^2\bar{c}_\mathrm{i}}{\mathrm{d}t} = 0 \tag{3–18}$$

and the equation for the oscillating variables is

$$\mathrm{j}\omega\tilde{c}_\mathrm{i}\mathrm{e}^{\mathrm{j}\omega t} + v_\mathrm{y}\frac{\mathrm{d}\tilde{c}_\mathrm{i}}{\mathrm{d}y}\mathrm{e}^{\mathrm{j}\omega t} - D_\mathrm{i}\frac{\mathrm{d}^2\tilde{c}_\mathrm{i}}{\mathrm{d}t}\mathrm{e}^{\mathrm{j}\omega t} = 0 \tag{3–19}$$

The exponential term $\mathrm{e}^{\mathrm{j}\omega t}$ in (3–19) can be canceled, thus eliminating the explicit dependence on time.

$$\mathrm{j}\omega\tilde{c}_\mathrm{i} + v_\mathrm{y}\frac{\mathrm{d}\tilde{c}_\mathrm{i}}{\mathrm{d}y} - D_\mathrm{i}\frac{\mathrm{d}^2\tilde{c}_\mathrm{i}}{\mathrm{d}t} = 0 \tag{3–20}$$

The convective-diffusion equation is made dimensionless by a dimensionless position

$$\xi = y/\delta_\mathrm{i} \tag{3–21}$$

where δ_i is defined as

$$\delta_\mathrm{i} = \left(\frac{3D_\mathrm{i}}{\alpha}\right)^{1/3} = \left(\frac{3}{a}\right)^{1/3}\frac{1}{Sc_\mathrm{i}^{1/3}}\sqrt{\frac{\nu}{\Omega}} \tag{3–22}$$

a dimensionless concentration $\theta_\mathrm{i} = \tilde{c}_\mathrm{i}/\tilde{c}_\mathrm{i}(0)$, and a dimensionless frequency given by

$$K_\mathrm{i} = \frac{\omega}{\Omega}\left(\frac{9\nu}{a^2 D_\mathrm{i}}\right)^{1/3} = \frac{\omega}{\Omega}\left(\frac{9}{a^2}\right)^{1/3}\mathrm{Sc}^{1/3} \tag{3–23}$$

After substitution, the dimensionless convective-diffusion equation is

$$\frac{\mathrm{d}^2\theta_\mathrm{i}}{\mathrm{d}\xi^2} + 3\xi^2\frac{\mathrm{d}\theta_\mathrm{i}}{\mathrm{d}\xi} - \mathrm{j}K_\mathrm{i}\theta_\mathrm{i} = 0 \tag{3–24}$$

Under the assumption that the axial velocity is given by the three-term expression valid close to the electrode surface, equation (3–5), and is made dimensionless by equation (3–21), equation (3–24) may be expressed as

$$\frac{\mathrm{d}^2\theta_\mathrm{i}}{\mathrm{d}\xi^2} + \left(3\xi^2 - \left(\frac{3}{a^4}\right)^{1/3}\frac{\xi^3}{\mathrm{Sc_i}^{1/3}} - \frac{b}{6}\left(\frac{3}{a}\right)^{5/3}\frac{\xi^4}{\mathrm{Sc_i}^{2/3}}\right)\frac{\mathrm{d}\theta_\mathrm{i}}{\mathrm{d}\xi} - \mathrm{j}K_\mathrm{i}\theta_\mathrm{i} = 0 \tag{3–25}$$

From equation (3–25), three coupled equations for a finite Schmidt number can be obtained using

$$\theta_\mathrm{i}(\xi, \mathrm{Sc_i}, K) = \theta_{\mathrm{i},0}(\xi, K) + \frac{\theta_{\mathrm{i},1}(\xi, K)}{\mathrm{Sc_i}^{1/3}} + \frac{\theta_{\mathrm{i},2}(\xi, K)}{\mathrm{Sc_i}^{2/3}} + \ldots \tag{3–26}$$

The first is equation, not dependent of the Schmidt number, is

$$\frac{\mathrm{d}^2\theta_{\mathrm{i},0}}{\mathrm{d}\xi^2} + 3\xi^2\frac{\mathrm{d}\theta_{\mathrm{i},0}}{\mathrm{d}\xi} - \mathrm{j}K_\mathrm{i}\theta_{\mathrm{i},0} = 0 \tag{3–27}$$

The other two equations, dependent on $\mathrm{Sc}^{-1/3}$ and $\mathrm{Sc}^{-2/3}$ are

$$\frac{\mathrm{d}^2\theta_{\mathrm{i},1}}{\mathrm{d}\xi^2} + 3\xi^2\frac{\mathrm{d}\theta_{\mathrm{i},1}}{\mathrm{d}\xi} - \mathrm{j}K_\mathrm{i}\theta_{\mathrm{i},1} = \left(\frac{3}{a^4}\right)^{1/3}\xi^3\frac{\mathrm{d}\theta_{\mathrm{i},0}}{\mathrm{d}\xi} \tag{3–28}$$

and

$$\frac{\mathrm{d}^2\theta_{\mathrm{i},2}}{\mathrm{d}\xi^2} + 3\xi^2\frac{\mathrm{d}\theta_{\mathrm{i},2}}{\mathrm{d}\xi} - \mathrm{j}K_\mathrm{i}\theta_{\mathrm{i},2} = \frac{b}{6}\left(\frac{3}{a}\right)^{5/3}\xi^4\frac{\mathrm{d}\theta_{\mathrm{i},0}}{\mathrm{d}\xi} + \left(\frac{3}{a^4}\right)^{1/3}\xi^3\frac{\mathrm{d}\theta_{\mathrm{i},1}}{\mathrm{d}\xi} \tag{3–29}$$

where terms order Sc^{-1} and greater are neglected. The boundary conditions are

$$\theta_{\mathrm{i},0} \to 0;\ \theta_{i,1} \to 0;\ \theta_{i,2} \to 0 \text{ as } \xi \to \infty \tag{3–30}$$

and

$$\theta_{\mathrm{i},0} = 1;\ \theta_{\mathrm{i},1} = 0;\ \theta_{\mathrm{i},2} = 0 \text{ as } \xi = 0 \tag{3–31}$$

The value of $\theta_{1,0}(0)$ was chosen arbitrarily because the governing equations for the impedance response are linear, even when the steady-state problem is non-linear.

As the Schmidt number approaches infinite, equation (3–27) describes the system. For finite Schmidt numbers, solutions of equations (3–27)-(3–29) are needed to obtain accurate values for diffusion impedance.

The diffusion impedance for a rotating disk electrode, for a finite Schmidt number, is given by

$$-\frac{1}{\theta'_{\mathrm{i}}(0)} = \frac{1}{\theta'_{\mathrm{i},0}(0) + \theta'_{\mathrm{i},1}(0)\mathrm{Sc_i}^{-1/3} + \theta'_{\mathrm{i},2}(0)\mathrm{Sc_i}^{-2/3}} \tag{3–32}$$

which can be expressed as

$$-\frac{1}{\theta'_{\mathrm{i}}(0)} = -\frac{1}{\theta'_{\mathrm{i},0}(0)} + \frac{\theta'_{\mathrm{i},1}(0)}{(\theta'_{\mathrm{i},0}(0))^2}\frac{1}{\mathrm{Sc}_{\mathrm{i}}^{1/3}} - \frac{1}{\theta'_{\mathrm{i},0}(0)}\left[\left(\frac{\theta'_{\mathrm{i},1}(0)}{\theta'_{\mathrm{i},0}(0)}\right)^2 - \frac{\theta'_{\mathrm{i},2}(0)}{\theta'_{\mathrm{i},0}(0)}\right]\frac{1}{\mathrm{Sc}_{\mathrm{i}}^{2/3}} \tag{3–33}$$

such that

$$Z_{(0)} = -\frac{1}{\theta'_{\mathrm{i},0}(0)} \tag{3–34}$$

$$Z_{(1)} = \frac{\theta'_{\mathrm{i},1}(0)}{(\theta'_{\mathrm{i},0}(0))^2} \tag{3–35}$$

and

$$Z_{(2)} = -\frac{1}{\theta'_{\mathrm{i},0}(0)}\left[\left(\frac{\theta'_{\mathrm{i},1}(0)}{\theta'_{\mathrm{i},0}(0)}\right)^2 - \frac{\theta'_{\mathrm{i},2}(0)}{\theta'_{\mathrm{i},0}(0)}\right] \tag{3–36}$$

The convective-diffusion impedance is obtained directly as a function of the Schmidt number from

$$-\frac{1}{\theta'_{\mathrm{i}}(0)} = Z_{(0)} + \frac{Z_{(1)}}{\mathrm{Sc}_{\mathrm{i}}^{1/3}} + \frac{Z_{(2)}}{\mathrm{Sc}_{\mathrm{i}}^{2/3}} \tag{3–37}$$

3.2.3 Numerical Methods for Rotating Disk Electrode

MatLab is a powerful simulation environment chosen to solve the complex, non-linear coupled differential equations of an electrochemical system. The algorithm BAND developed by Newman[74] in 1968 in Fortran, was converted into a MatLab code and is called in the programs developed to solve electrochemical differential equations. BAND is described in detail in Appendix A. The MatLab codes written to solve equation (3–25) are shown in Appendix B.

3.2.4 Convective-Diffusion Impedance for a Rotating Disk Electrode

The dimensionless oscillating concentration, $\theta_{\mathrm{i}} = \tilde{c}_{\mathrm{i}}/\tilde{c}_{\mathrm{i}}(0)$, is a complex quantity and results for $\theta_{\mathrm{i},0}$, $\theta_{\mathrm{i},1}$ and $\theta_{\mathrm{i},2}$ are presented in Figure 3-2. These results presented in Figure 3-2A were obtained from solving for equation (3–27). The results presented in Figure

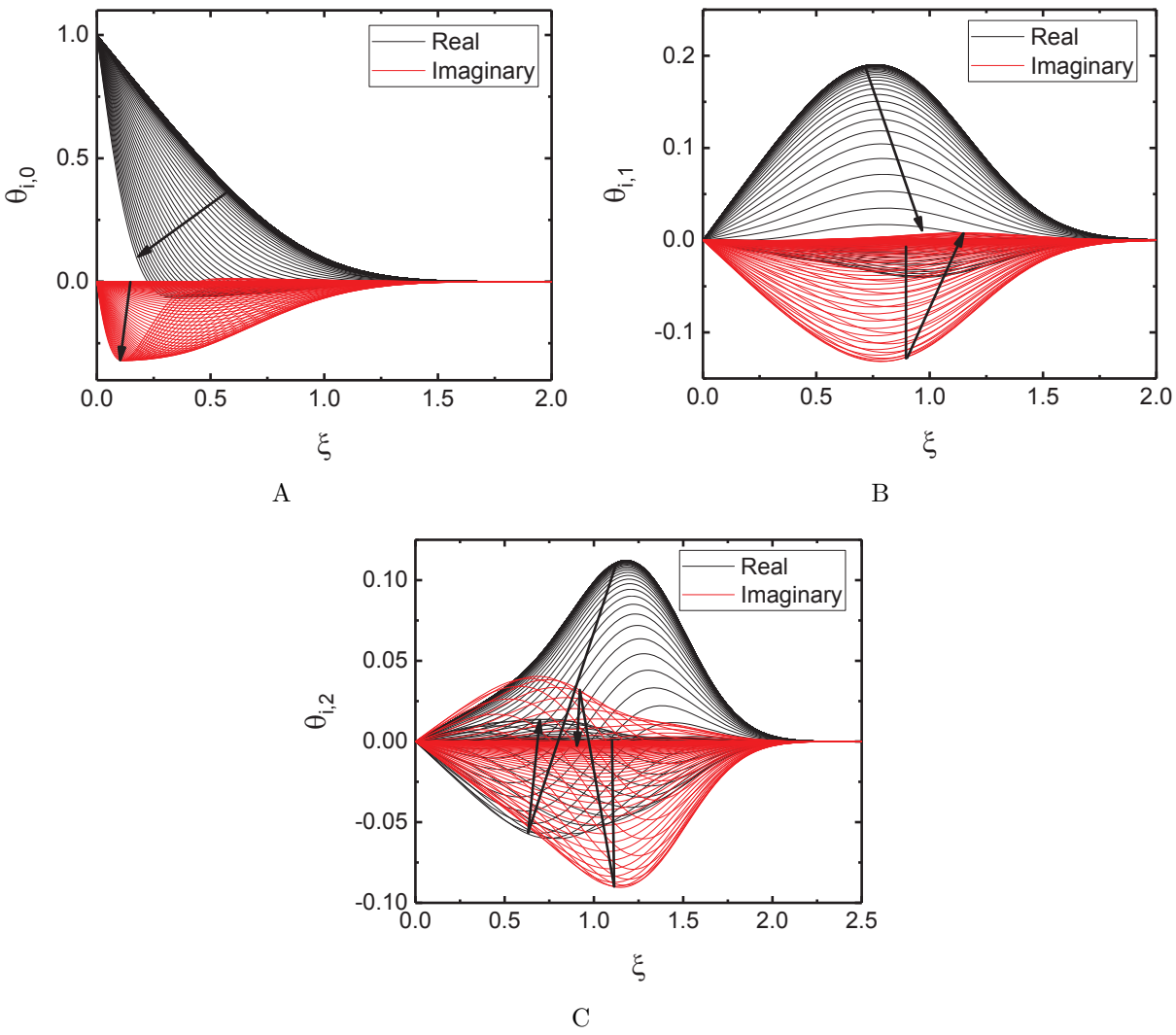


Figure 3-2. Oscillating dimensionless concentrations for a rotating disk electrode over a range of frequencies for both the real and imaginary components for a) $\theta_{i,0}$, b) $\theta_{i,1}$, and c) $\theta_{i,2}$. The values of K, arrow indicating increasing values of K, ranged from 10^{-2} to 10^{2} with 10 points per decade.

3-2B came from solving equation (3–28), however, these results depend on the solution of equation (3–27). The results presented in Figure 3-2C came from solving equation (3–29) and is dependent on the solutions for both equation (3–27) and (3–28). It is clear from decreasing magnitude of the oscillating concentration values in Figure 3-2A to 3-2C that the first term is the most important for determining the impedance and importance decreases with dependence on the Schmidt number.

The contribution to the impedance for each term is presented in Figure 3-3. These plots were created from equations (3–34) to (3–36). The Nyquist plot of all three Z_{i} values shows clearly that Z_0 resembles the hyperbolic tangent model used to describe diffusion through a film (no convective term). The characteristic frequency for Z_1 is about half that of Z_0 and the characteristic frequency for Z_2 is even smaller. This can be seen better when looking at the imaginary impedance as a function of frequency, presented in Figure 3-3C. The real impedance as a function of frequency as well as the Nyquist plot both show the magnitude of the dimensionless diffusion impedance is much larger for Z_0 then for Z_1 or Z_2.

A comparison of the impedance response obtained under the assumption of an infinite Schmidt number and a finite Schmidt number (Sc=1000 and Sc=100) is presented in Figure 3-4. The solutions generated were in agreement with Orazem and Tribollet[5]. The low frequency values show the greatest difference, see Figure 3-4A, between the assumption of using an infinite Schmidt number. The differences in low frequency values are also shown in Figrues 3-4B and 3-4C. The error at low frequency of the dimensionless diffusion impedance for a rotating disk electrode with different Schmidt numbers is shown in Table 3-1. The percent error was calculated using

$$\% \text{ error} = \left| -\frac{1}{\theta'_{\mathrm{i,r}}(0)}\left(\omega \to 0\right) - Z_{(0),\mathrm{r}}(\omega \to 0)\right| \Big/ \left(-\frac{1}{\theta'_{\mathrm{i,r}}(0)}\left(\omega \to 0\right)\right) \tag{3–38}$$

where $Z_{(0),\mathrm{r}}$ is the real component of equation (3–34). If the infinite Schmidt number analysis is used to calculate the convective-diffusion impedance and the Schmidt number in the system is on the order of 100, the low frequency part of the impedance will be inaccurate up to 6.62 %. With a Schmidt number of 1000 the error is still as great as 3.03 %. The dimensionless convective-diffusion impedance analytical expression (for an infinite Schmidt number) should not be used for small values of the Schmidt number especially when mass transfer plays a large role in the electrochemical system.

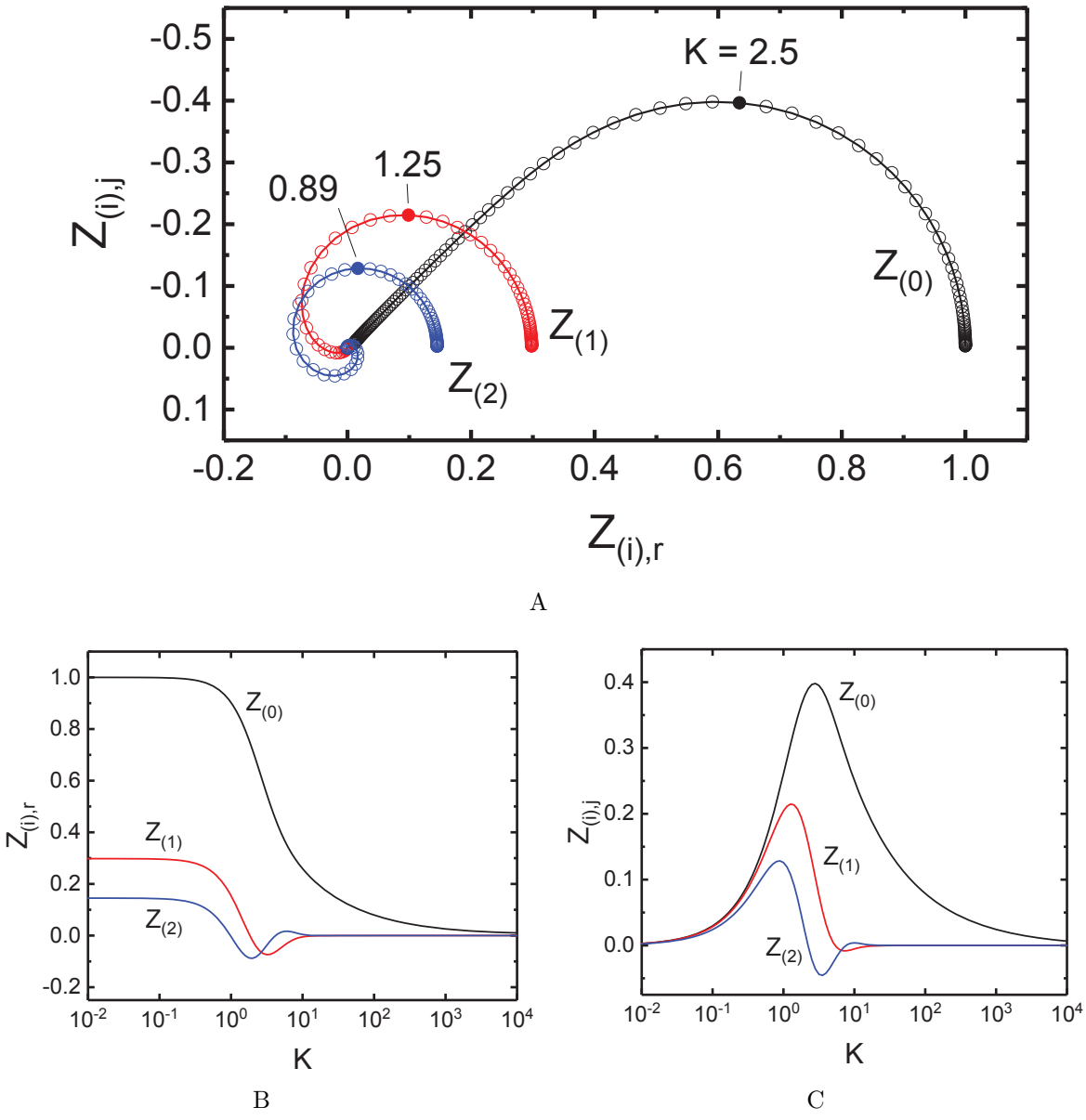


Figure 3-3. Contributions to the impedance from series expansion for finite Schmidt number, $Z_{(0)}$, $Z_{(1)}$, and $Z_{(2)}$ shown in a) Nyquist form, b) real part of the dimensionless diffusion impedance as a function of frequency, and c) imaginary part of the dimensionless diffusion impedance as a function of frequency. calculated from equations (3–34)-(3–36).

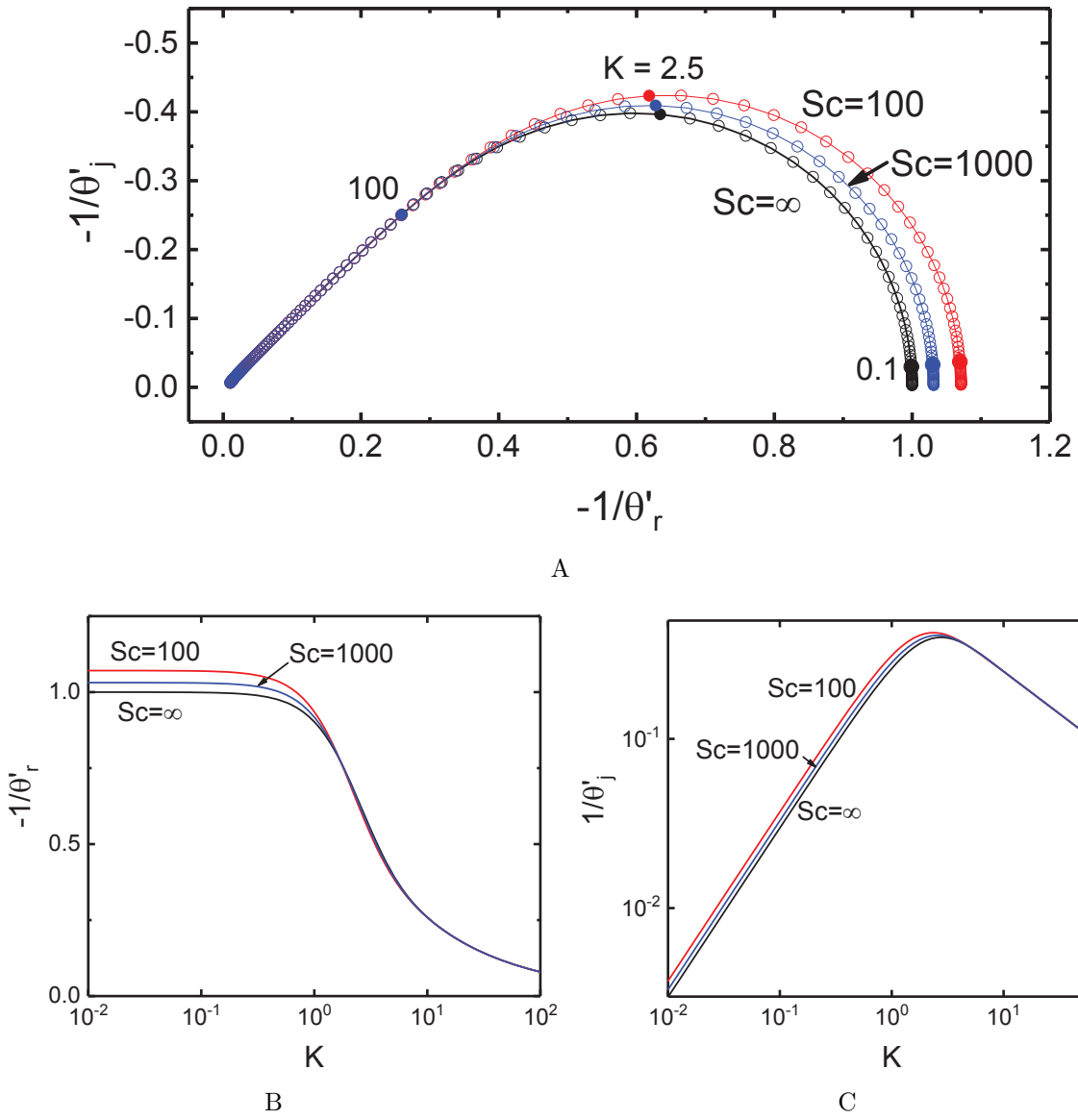


Figure 3-4. Dimensionless diffusion impedance obtained for a rotating disk under assumption of infinite Schmidt number and for a finite Schmidt number equal to 100 and 1000 shown in A) Nyquist form, B) real part of the dimensionless diffusion impedance as a function of frequency, and C) imaginary part of the dimensionless diffusion impedance as a function of frequency.

Table 3-1. Error at low frequency in convective diffusion impedance for a rotating disk electrode when compared to using infinite Schmidt number analysis

Sc	% Error
10000	1.40
1000	3.03
100	6.62
50	8.39

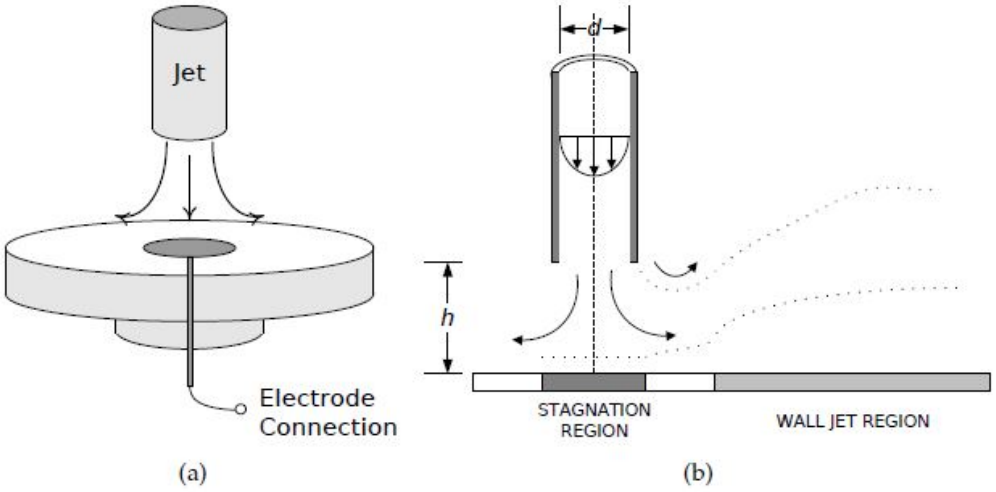


Figure 3-5. Submerged impinging jet flow diagram a) schematic illustration and b) identification of flow regimes [3].

3.3 Submerged Impinging Jet Electrode

The submerged impinging jet electrode, Figure 3-5, though less popular then the rotating disk electrode, is also very attractive to characterize electrochemical systems. With a stagnation region, shown in Figure 3-5B, the convective diffusion of fluid towards the electrode is uniform, similar to the rotating disk electrode. For an electrode that is entirely in the stagnation region, the mass-transfer rate is uniform. In contrast to the rotating disk, the electrode is stationary and is therefore suitable for in situ observation[3].

3.3.1 Velocity Expansion for a Submerged Impinging Jet

The fluid flow within the stagnation region of the electrode in an impinging jet cell is well-defined [75, 76, 77, 78, 79, 80]. The stagnation region is defined to be the region surrounding the stagnation point in which the axial velocity, given by

$$v_{\mathrm{y}} = -\sqrt{a_{\mathrm{IJ}}\nu}\phi(\eta) \tag{3–39}$$

is independent of the radial velocity

$$v_{\mathrm{r}} = \frac{a_{\mathrm{IJ}}}{2} r \frac{\phi(\eta)}{d\eta} \tag{3–40}$$

where a_{IJ} is a hydrodynamic constant that is dependent on geometry and fluid velocity, y and r are the axial and radial directions, ν is the kinematic viscosity and ϕ is a stream function given in terms of a dimensionless axial position

$$\eta = y\sqrt{\frac{a_{\mathrm{IJ}}}{\nu}} \tag{3–41}$$

as [75]

$$\phi(\eta) = 1.352\eta^2 - \frac{1}{3}\eta^3 + 7.2888 \times 10^{-3}\eta^6 + ... \tag{3–42}$$

The stagnation region extends to a radial distance that is approximately the size of the impinging jet nozzle[76].

3.3.2 Mathematical Development for Impinging Jet Electrode

Under the assumption that the electrode is uniformly accessible, the equation governing mass transfer in the frequency domain is given by

$$\frac{\mathrm{d}^2\theta_{\mathrm{i}}}{\mathrm{d}\xi^2} + \left(3\xi^2 - \left(\frac{3}{1.352^4}\right)^{1/3}\frac{\xi^3}{\mathrm{Sc}_{\mathrm{i}}^{1/3}}\right)\frac{\mathrm{d}\theta_{\mathrm{i}}}{\mathrm{d}\xi} - \mathrm{j}K_{\mathrm{i}}\theta_{\mathrm{i}} = 0 \tag{3–43}$$

where

$$K_{\mathrm{i}} = \frac{\omega}{a_{\mathrm{IJ}}}\left(\frac{9}{1.352^2}\right)^{1/3} Sc_{\mathrm{i}}^{1/3} \tag{3–44}$$

The axial direction is made dimensionless using equation (3–21). The dimensionless distance for the impinging jet is set by

$$\delta_{\mathrm{i}} = \left(\frac{3}{1.352}\right)^{1/3}\sqrt{\frac{\nu}{a_{\mathrm{IJ}}}}Sc_{\mathrm{i}}^{1/3} \tag{3–45}$$

and the oscillating concentration is made dimensionless in the same way as in the rotating disk electrode case previously described, $\theta_{\mathrm{i}} = \widetilde{c}_i/\widetilde{c}_i(0)$. As done for the rotating disk electrode, the impinging jet convective-diffusion equation can be expressed as a series expansion, see equation (3–26), where a solution to each oscillating concentration can be found from

$$\frac{\mathrm{d}^2\theta_{\mathrm{i},0}}{\mathrm{d}\xi^2} + 3\xi^2\frac{\mathrm{d}\theta_{\mathrm{i},0}}{\mathrm{d}\xi} - \mathrm{j}K_{\mathrm{i}}\theta_{\mathrm{i},0} = 0 \tag{3–46}$$

$$\frac{\mathrm{d}^2\theta_{\mathrm{i},1}}{\mathrm{d}\xi^2} + 3\xi^2\frac{\mathrm{d}\theta_{\mathrm{i},1}}{\mathrm{d}\xi} - \mathrm{j}K_{\mathrm{i}}\theta_{\mathrm{i},1} = \left(\frac{3}{1.352^4}\right)^{1/3}\xi^3\frac{\mathrm{d}\theta_{\mathrm{i},0}}{\mathrm{d}\xi} \tag{3–47}$$

and

$$\frac{\mathrm{d}^2\theta_{\mathrm{i},2}}{\mathrm{d}\xi^2} + 3\xi^2\frac{\mathrm{d}\theta_{\mathrm{i},2}}{\mathrm{d}\xi} - \mathrm{j}K_{\mathrm{i}}\theta_{\mathrm{i},2} = \left(\frac{3}{1.352^4}\right)^{1/3}\xi^3\frac{\mathrm{d}\theta_{\mathrm{i},1}}{\mathrm{d}\xi} \tag{3–48}$$

The individual contributions to impedance separated by influence of Schmidt number are

$$Z_{(0)} = -\frac{1}{\theta'_{\mathrm{i},0}(0)} \tag{3–49}$$

$$Z_{(1)} = \frac{\theta'_{\mathrm{i},1}(0)}{(\theta'_{\mathrm{i},0}(0))^2} \tag{3–50}$$

and

$$Z_{(2)} = -\frac{1}{\theta'_{\mathrm{i},0}(0)}\left[\left(\frac{\theta'_{\mathrm{i},1}(0)}{\theta'_{\mathrm{i},0}(0)}\right)^2 - \frac{\theta'_{\mathrm{i},2}(0)}{\theta'_{\mathrm{i},0}(0)}\right] \tag{3–51}$$

The convective-diffusion impedance is obtained directly as a function of the Schmidt number from

$$-\frac{1}{\theta'_{\mathrm{i}}(0)} = Z_{(0)} + \frac{Z_{(1)}}{\mathrm{Sc}_{\mathrm{i}}^{1/3}} + \frac{Z_{(2)}}{\mathrm{Sc}_{\mathrm{i}}^{2/3}} \tag{3–52}$$

3.3.3 Numerical Methods for Impinging Jet Electrode

Similarity to the rotating disk electrode, MatLab was the chosen simulation environment to solve the complex, non-linear coupled differential equations for an impinging jet electrode. The Matlab-converted-BAND algorithm is called in the programs developed to solve electrochemical differential equations. BAND is described in detail in Appendix A. The MatLab codes written to solve equation (3–43) are shown in Appendix C.

3.3.4 Convective-Diffusion Impedance for a Submerged Impinging Jet Electrode

As for the rotating disk electrode, the dimensionless oscillating concentration for the impinging jet electrode, $\theta_{\mathrm{i}} = \tilde{c}_{\mathrm{i}}/\tilde{c}_{\mathrm{i}}(0)$, is a complex quantity and results for $\theta_{\mathrm{i},0}$, $\theta_{\mathrm{i},1}$ and $\theta_{\mathrm{i},2}$ (from equations (3–46)-(3–48))are presented in Figure 3-6. The results presented in Figure 3-6A were obtained from solving for equation (3–46). The results presented

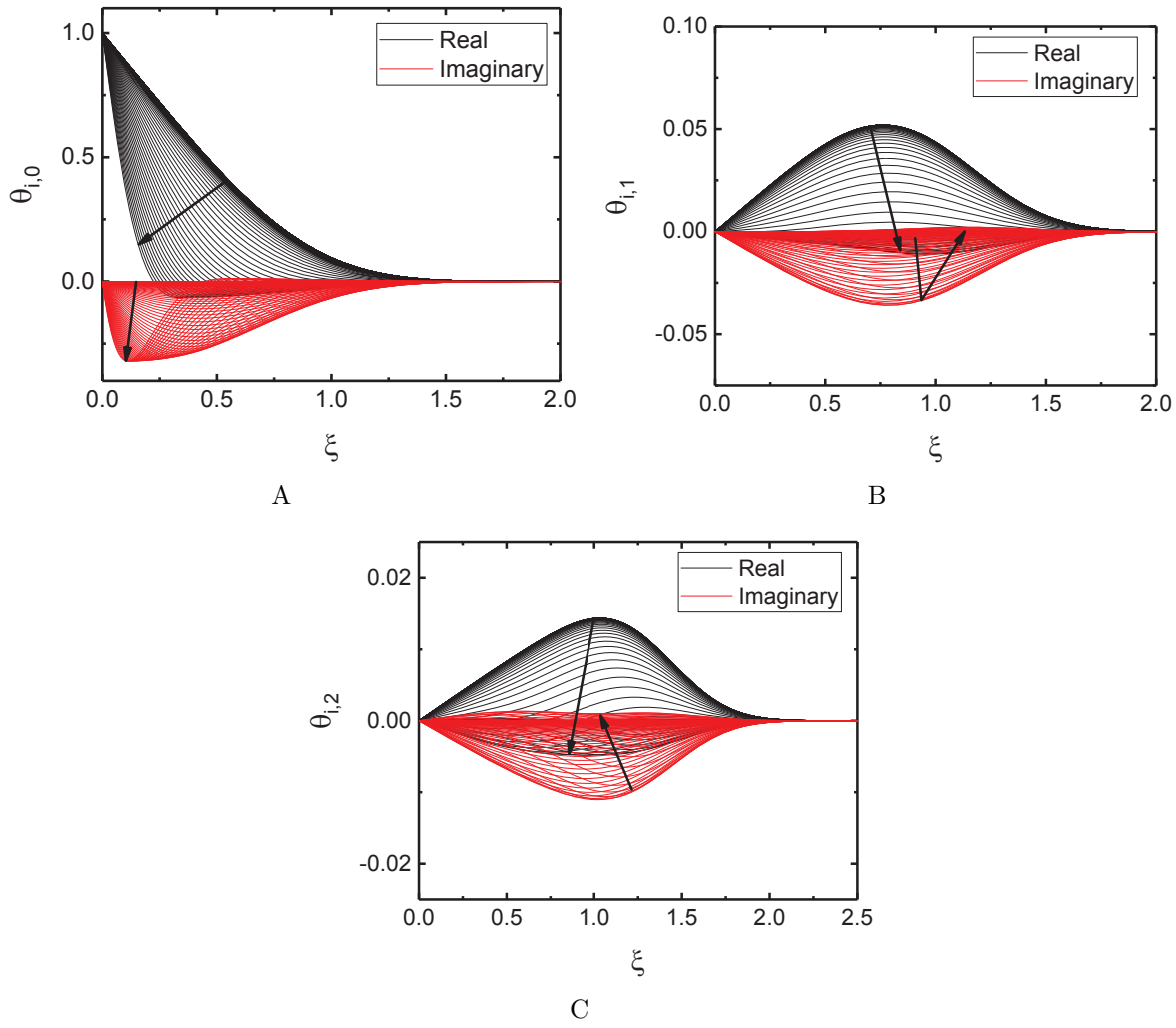


Figure 3-6. Oscillating dimensionless concentrations for a submerged impinging jet electrode over a range of frequencies for both the real and imaginary components for a) $\theta_{i,0}$, b) $\theta_{i,1}$, and c) $\theta_{i,2}$. The values of K, arrow indicating increasing values of K, ranged from 10^{-2} to 10^{2} with 10 points per decade.

in Figure 3-6B came from solving equation (3–47), however, these results depend on the solution of equation (3–46). The results presented in Figure 3-6C came from solving equation (3–48) and are dependent on the solutions for both equation (3–46) and (3–47). The oscillating concentration presented in Figure 3-6A is the same as for the rotating disk electrode, Figure 3-2A. The second and third oscillating concentration plots, Figures 3-6B and 3-6C, however, show an even smaller magnitude then the second and third oscillating

Table 3-2. Error at low frequency in convective diffusion impedance for a submerged impinging jet electrode when compared to using infinite Schmidt number analysis

Sc	% Error
10000	0.38
1000	0.83
100	1.82
50	2.31

concentrations for the rotating disk electrode. This implies that the second and third terms in the velocity expansion will have less impact on the impedance results.

The contribution to the impedance for each term is presented in Figure 3-7. These plots were created from equations (3–49) to (3–51). The Nyquist plot of all three Z_i values shows clearly that Z_0 resembles, but is not identical to, the hyperbolic tangent model used to describe diffusion through a film (no convective term). The characteristic frequency for Z_1 is about half that of Z_0 and the characteristic frequency for Z_2 is even smaller. This can be seen better when looking at the imaginary impedance as a function of frequency, presented in Figure 3-7C. The real impedance as a function of frequency as well as the Nyquist plot both show the magnitude of the dimensionless diffusion impedance is much larger for Z_0 then for Z_1 or Z_2. The impedances that depend on the Schmidt number, Z_1 or Z_2, are even smaller then for the rotating disk electrode system.

A comparison of the impedance response obtained under the assumption of an infinite Schmidt number and a finite Schmidt number (Sc=1000 and Sc=100) is presented in Figure 3-8. The low frequency values show the greatest difference, see Figure 3-8A, between the assumption of using an infinite Schmidt number. The differences in low frequency values are also shown in Figrues 3-8B and 3-8C. The differences in low frequency are less then when using a rotating disk electrode. The error at low frequency of the dimensionless diffusion impedance for a rotating disk electrode with different Schmidt numbers is shown in Table 3-2. The percent error was calculated using equation (3–38).

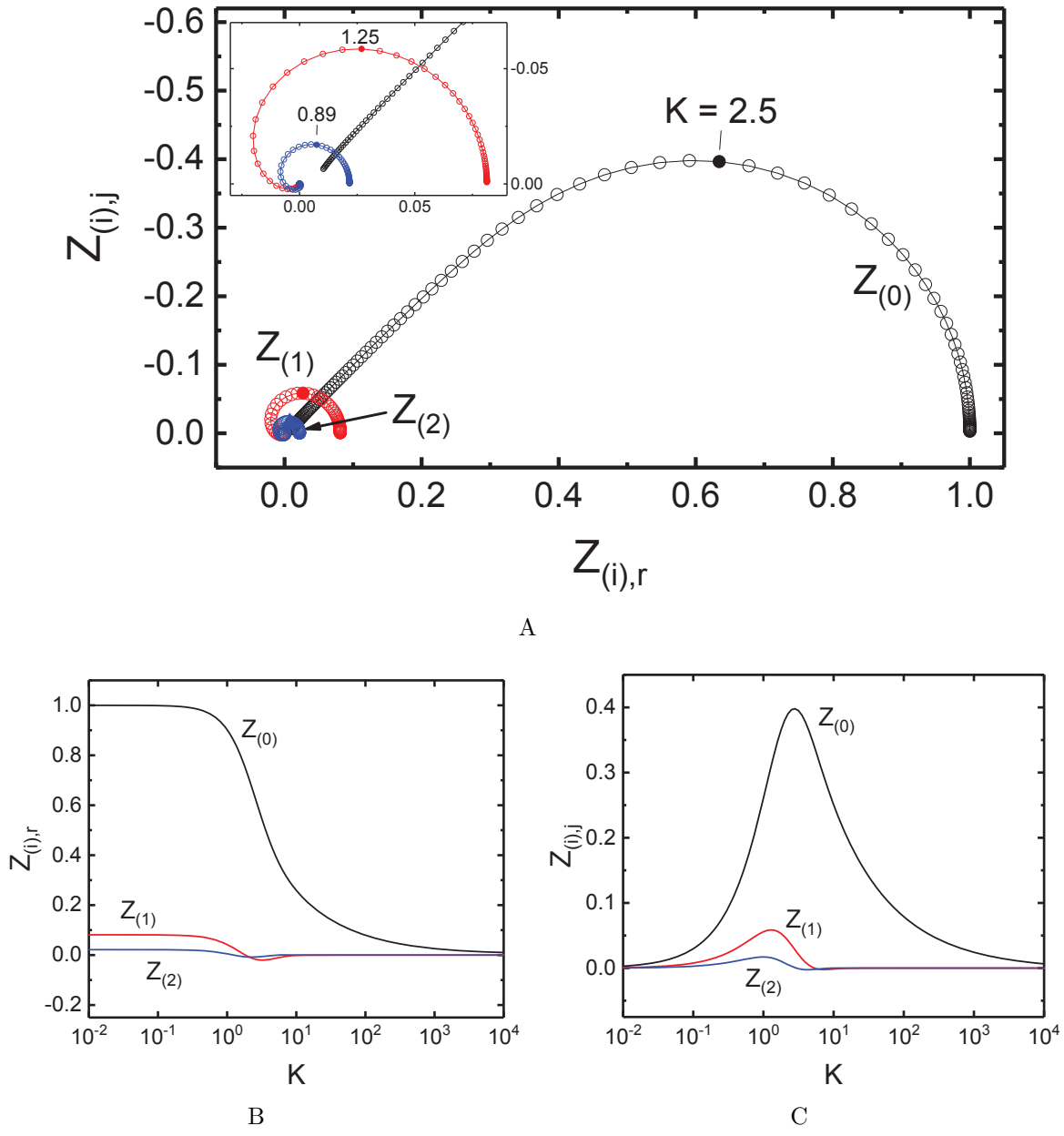


Figure 3-7. Contributions to the impedance from series expansion for finite Schmidt number, $Z_{(0)}$, $Z_{(1)}$, and $Z_{(2)}$ shown in a) Nyquist form, b) real part of the dimensionless diffusion impedance as a function of frequency, and c) imaginary part of the dimensionless diffusion impedance as a function of frequency. calculated from equations (3–49)-(3–51).

If the infinite Schmidt number analysis is used to calculate the convective-diffusion impedance and the Schmidt number in the system is on the order of 100, the low frequency part of the impedance will be inaccurate up to 1.82 %. With a Schmidt number of 1000 the error is 0.83 %. The dimensionless convective-diffusion impedance analytical expression (for an infinite Schmidt number) is will likely be within 2 % of the exact answer.

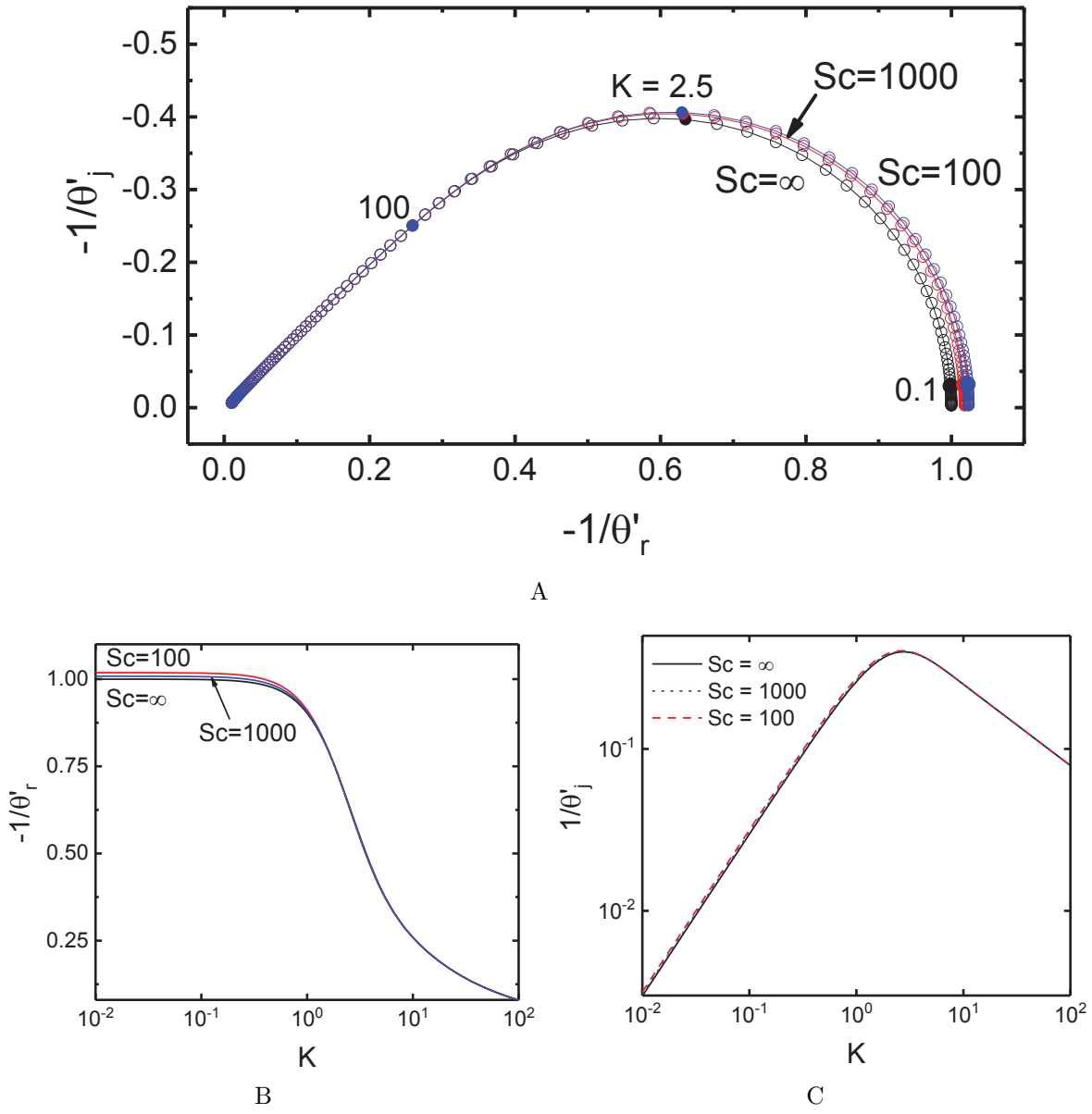


Figure 3-8. Dimensionless diffusion impedance obtained for an impinging jet electrode under the assumption of infinite Schmidt number and for a finite Schmidt number equal to 100 and 1000 shown in A) Nyquist form, B) real part of the dimensionless diffusion impedance as a function of frequency, and C) imaginary part of the dimensionless diffusion impedance as a function of frequency.

CHAPTER 4
CONVECTIVE-DIFFUSION IMPEDANCE WITH HOMOGENEOUS CHEMICAL REACTIONS

A mathematical model was developed for the impedance response associated with the coupled homogeneous chemical and heterogeneous electrochemical reactions. The model includes a homogeneous reaction in the electrolyte where species AB reacts reversibly to form A^- and B^+ and B^+ reacts electrochemically on a rotating disk electrode to produce B. The resulting convective diffusion impedance has two asymmetric capacitive loops, one associated with convective diffusion impedance the other with the homogeneous reaction. For an infinitely fast homogeneous reaction, the system is shown to behave as though AB is the electroactive species. A modified Gerischer impedance was found to provide a good fit to the simulated data.

4.1 Mathematical Development for Convective Diffusion and Homogeneous Reaction

A mathematical model is presented below for the impedance response associated with the coupling of homogeneous and heterogeneous electrochemical reactions.

4.1.1 Governing Equations

The flux density of species i in a dilute electrolyte and in absence of migration is expressed as

$$\mathbf{N}_\mathrm{i} = -D_\mathrm{i}\nabla c_\mathrm{i} + c_\mathrm{i}\mathbf{v} \tag{4–1}$$

For an axisymmetric rotating-disk electrode, the convective diffusion equation with homogeneous reaction is expressed in cylindrical coordinates as

$$\frac{\partial c_\mathrm{i}}{\partial t} + v_\mathrm{y}\frac{\partial c_\mathrm{i}}{\partial y} = D_\mathrm{i}\frac{\partial^2 c_\mathrm{i}}{\partial y^2} + R_\mathrm{i} \tag{4–2}$$

where R_i is the rate of production of species i by homogeneous reactions.

4.1.2 Homogeneous Reaction

A diagram of an electrochemical reaction coupled with a chemical reaction is shown in Figure 4-1. where the reactions may be expressed as

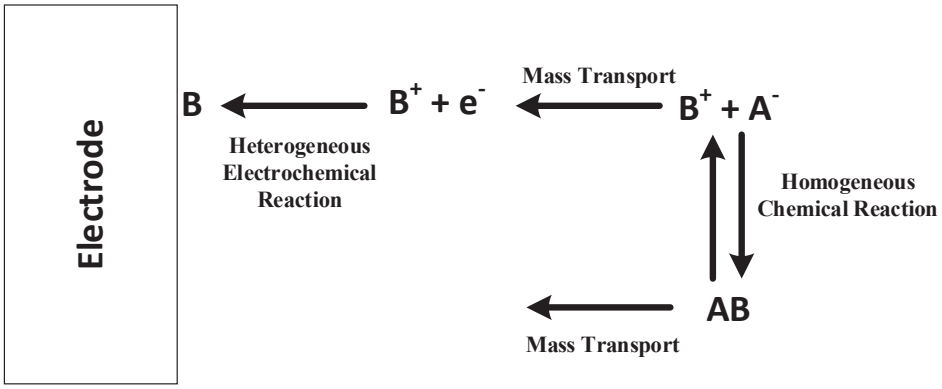


Figure 4-1. A diagram of an electrochemical reaction coupled by the influence of a homogeneous chemical reaction.

$$\mathrm{AB} \underset{k_\mathrm{b}}{\overset{k_\mathrm{f}}{\rightleftharpoons}} \mathrm{A^-} + \mathrm{B^+} \tag{4–3}$$

One of the ionic species involved in the homogeneous reaction, $\mathrm{B^+}$, was assumed to be electroactive and was consumed at the electrode. The electrochemical reaction was expressed as

$$\mathrm{B^+} + \mathrm{e^-} \rightarrow \mathrm{B} \tag{4–4}$$

The corresponding current, which is dependent on concentration and potential, was expressed as

$$i_{\mathrm{B^+}} = -K_{\mathrm{B^+}} c_{\mathrm{B^+}}(0) \exp\left(-b_{\mathrm{B^+}} V\right) \tag{4–5}$$

As the electroactive species is consumed, a concentration gradient of $\mathrm{B^+}$ must exist near the electrode surface.

Far from the electrode surface, the species AB, $\mathrm{A^-}$, and $\mathrm{B^+}$ are equilibrated, thus

$$K_\mathrm{eq} = \frac{k_\mathrm{f}}{k_\mathrm{b}} = \frac{c_{\mathrm{A^-}}(\infty) c_{\mathrm{B^+}}(\infty)}{c_\mathrm{AB}(\infty)} \tag{4–6}$$

The reaction term was expressed as

$$R_{\mathrm{A^-}} = R_{\mathrm{B^+}} = -R_\mathrm{AB} = k_\mathrm{f} c_\mathrm{AB}(y) - k_\mathrm{b} c_{\mathrm{A^-}}(y) c_{\mathrm{B^+}}(y) \tag{4–7}$$

Combination of equations (4–6) and (4–7) yields

$$R_{\mathrm{A^-}} = R_{\mathrm{B^+}} = -R_{\mathrm{AB}} = k_{\mathrm{b}}(K_{\mathrm{eq}}c_{\mathrm{AB}}(y) - c_{\mathrm{A^-}}(y)c_{\mathrm{B^+}}(y)) \tag{4–8}$$

where reaction (4–3) is not assumed to be equilibrated in a region near the electrode surface. In equation (4–8), k_{b} is the variable that can be independently explored, whereas, K_{eq} is constrained by assumed values of bulk concentrations. The form of equation (4–8) makes the steady-state problem non-linear.

4.1.3 Velocity Expression

The velocity profile used in equation (4–2) is created from equation (2–24). It includes the interpolation function, equation (2–31), with weighting parameter f, equation (2–32), to create a function that uses the velocity expansion for close to the electrode, equation (2–27), and far from the electrode, equation (2–30). The final velocity expression including all the details above is

$$\begin{aligned} v_y &= \left(1 - \frac{1}{1+\mathrm{e}^{-\alpha(y\sqrt{\Omega/\nu}-\zeta_0)}}\right)\sqrt{\nu\Omega}\left(-ay^2\frac{\Omega}{\nu} + \frac{1}{3}y^3\left(\frac{\Omega}{\nu}\right)^3 + \frac{b}{6}y^4\left(\frac{\Omega}{\nu}\right)\right) \\ &+ \left(\frac{1}{1+\mathrm{e}^{-\alpha(y\sqrt{\Omega/\nu}-\zeta_0)}}\right)\sqrt{\nu\Omega}\left\{-\alpha + \frac{2A}{\alpha}\exp\left(-\alpha y\sqrt{\frac{\Omega}{\nu}}\right)\right. \\ &- \left.\frac{A^2+B^2}{2\alpha^3}\exp\left(-2\alpha y\sqrt{\frac{\Omega}{\nu}}\right) - \frac{A(A^2+B^2)}{6\alpha^5}\exp\left(-3\alpha y\sqrt{\frac{\Omega}{\nu}}\right)\right\} \end{aligned} \tag{4–9}$$

The axial velocity is shown in figure 2-8C, where the solid line represents the velocity expression above.

4.1.4 Impedance with Homogeneous Chemical Reactions

The conservation equation for each species may be written as

$$\frac{\partial c_{\mathrm{AB}}}{\partial t} + v_{\mathrm{y}}\frac{\partial c_{\mathrm{AB}}}{\partial y} = D_{\mathrm{AB}}\frac{\partial^2 c_{\mathrm{AB}}}{\partial \mathrm{y}^2} - k_{\mathrm{b}}(K_{\mathrm{eq}}c_{\mathrm{AB}}(y) - c_{\mathrm{A^-}}(y)c_{\mathrm{B^+}}(y)) \tag{4–10}$$

for AB,

$$\frac{\partial c_{\mathrm{A^-}}}{\partial t} + v_{\mathrm{y}}\frac{\partial c_{\mathrm{A^-}}}{\partial y} = D_{\mathrm{A^-}}\frac{\partial^2 c_{\mathrm{A^-}}}{\partial \mathrm{y}^2} + k_{\mathrm{b}}(K_{\mathrm{eq}}c_{\mathrm{AB}}(y) - c_{\mathrm{A^-}}(y)c_{\mathrm{B^+}}(y)) \tag{4–11}$$

for A^-, and

$$\frac{\partial c_{B^+}}{\partial t} + v_y \frac{\partial c_{B^+}}{\partial y} = D_{B^+} \frac{\partial^2 c_{B^+}}{\partial y^2} + k_b (K_{eq} c_{AB}(y) - c_{A^-}(y) c_{B^+}(y)) \tag{4–12}$$

for B^+. The boundary conditions far from the electrode were

$$c_i \to c_i(\infty) \quad \text{for} \quad y \to \infty \tag{4–13}$$

and the boundary conditions at the electrode surface were

$$\left. \frac{\partial c_i}{\partial y} \right|_{y=0} = 0 \quad \text{for} \quad y = 0 \tag{4–14}$$

for the non–reacting species, and

$$F D_{B^+} \left. \frac{\partial c_{B^+}}{\partial y} \right|_{y=0} = i_{B^+} \quad \text{for} \quad y = 0 \tag{4–15}$$

for the reacting species B^+. The concentrations of each species were represented in terms of steady-state and oscillating terms as

$$c_i = \overline{c}_i + \mathrm{Re}\left\{ \widetilde{c}_i \exp(j\omega t) \right\} \tag{4–16}$$

The resulting equations governing the steady-state are

$$v_y \frac{\partial \overline{c}_{AB}}{\partial y} = D_{AB} \frac{\partial^2 \overline{c}_{AB}}{\partial y^2} - \overline{R}_{AB} \tag{4–17}$$

$$v_y \frac{\partial \overline{c}_{A^-}}{\partial y} = D_{A^-} \frac{\partial^2 \overline{c}_{A^-}}{\partial y^2} + \overline{R}_{A^-} \tag{4–18}$$

and

$$v_y \frac{\partial \overline{c}_{B^+}}{\partial y} = D_{B^+} \frac{\partial^2 \overline{c}_{B^+}}{\partial y^2} + \overline{R}_{B^+} \tag{4–19}$$

where

$$\overline{R}_{A^-} = \overline{R}_{B^+} = -\overline{R}_{AB} = k_b (K_{eq} \overline{c}_{AB}(y) - \overline{c}_{A^-}(y) \overline{c}_{B^+}(y)) \tag{4–20}$$

The equations governing the sinusoidal steady state are

$$\frac{\partial \widetilde{c}_{\mathrm{AB}}}{\partial t} + v_{\mathrm{y}} \frac{\partial \widetilde{c}_{\mathrm{AB}}}{\partial y} = D_{\mathrm{AB}} \frac{\partial^2 \widetilde{c}_{\mathrm{AB}}}{\partial \mathrm{y}^2} - \widetilde{R}_{\mathrm{AB}} \tag{4–21}$$

$$\frac{\partial \widetilde{c}_{\mathrm{A^-}}}{\partial t} + v_{\mathrm{y}} \frac{\partial \widetilde{c}_{\mathrm{A^-}}}{\partial y} = D_{\mathrm{A^-}} \frac{\partial^2 \widetilde{c}_{\mathrm{A^-}}}{\partial \mathrm{y}^2} + \widetilde{R}_{\mathrm{A^-}} \tag{4–22}$$

and

$$\frac{\partial \widetilde{c}_{\mathrm{B^+}}}{\partial t} + v_{\mathrm{y}} \frac{\partial \widetilde{c}_{\mathrm{B^+}}}{\partial y} = D_{\mathrm{B^+}} \frac{\partial^2 \widetilde{c}_{\mathrm{B^+}}}{\partial \mathrm{y}^2} + \widetilde{R}_{\mathrm{B^+}} \tag{4–23}$$

where

$$\begin{aligned} \widetilde{R}_{\mathrm{A^-}} &= \widetilde{R}_{\mathrm{B^+}} = -\widetilde{R}_{\mathrm{AB}} \\ &= k_{\mathrm{f}} \widetilde{c}_{\mathrm{AB}}(y) - k_{\mathrm{b}} \overline{c}_{\mathrm{A^-}}(y) \widetilde{c}_{\mathrm{B^+}}(y) - k_{\mathrm{b}} \widetilde{c}_{\mathrm{A^-}}(y) \overline{c}_{\mathrm{B^+}}(y) \end{aligned} \tag{4–24}$$

and terms with $\widetilde{c}^2$ have been neglected. Equations (4–21)-(4–24) are coupled and linear but are dependent on the steady-state solution. The boundary conditions for the oscillating concentrations were

$$\widetilde{c}_{\mathrm{i}} = 0 \quad \text{for} \quad y \to \infty \tag{4–25}$$

for each species

$$\left. \frac{\partial \widetilde{c}_{\mathrm{i}}}{\partial \mathrm{y}} \right|_{y=0} = 0 \quad \text{for} \quad y = 0 \tag{4–26}$$

for the AB and $\mathrm{A^-}$, and

$$\widetilde{c}_{\mathrm{B^+}}(0) = 1 \quad \text{for} \quad y = 0 \tag{4–27}$$

for the reacting species $\mathrm{B^+}$. The value of $\widetilde{c}_{\mathrm{B^+}}(0)$ was chosen arbitrarily because the governing equations for the impedance response are linear, even when the steady-state problem is non-linear.

4.1.4.1 Diffusion impedance

The oscillating current associated with $\mathrm{B^+}$, was expressed as

$$\widetilde{i}_{\mathrm{B^+}} = \left(\frac{\partial i_{\mathrm{B^+}}}{\partial V} \right)_{c_{\mathrm{B^+}}(0)} \widetilde{V} + \left(\frac{\partial i_{\mathrm{B^+}}}{\partial c_{\mathrm{B^+}}(0)} \right)_{V} \widetilde{c}_{\mathrm{B^+}}(0) \tag{4–28}$$

The flux expression for B^+ yields a second equation for the oscillating current density as

$$\widetilde{i}_{B^+} = \mathrm{F} D_{B^+} \left. \frac{\mathrm{d}\widetilde{c}_{B^+}}{\mathrm{d}y} \right|_{y=0} \tag{4–29}$$

Equation (4–28) was divided by equation (4–29), yielding

$$1 = \left(\frac{\partial i_{B^+}}{\partial V} \right)_{c_{B^+}(0)} \frac{\widetilde{V}}{\widetilde{i}_{B^+}} + \left(\frac{\partial i_{B^+}}{\partial c_{B^+}(0)} \right)_V \frac{\widetilde{c}_{B^+}(0)}{\mathrm{F} D_{B^+} \left. \frac{\mathrm{d}\widetilde{c}_{B^+}}{\mathrm{d}y} \right|_{y=0}} \tag{4–30}$$

Thus, the impedance was expressed as

$$Z_{\mathrm{F},B^+} = R_{\mathrm{t},B^+} + Z_{\mathrm{D},B^+} \tag{4–31}$$

where, from the respective derivatives of equation (4–5),

$$R_{\mathrm{t},B^+} = \frac{1}{K_{B^+} b_{B^+} \overline{c}_{B^+}(0) \exp\left(-b_{B^+} \overline{V}\right)} \tag{4–32}$$

and

$$Z_{\mathrm{D},B^+} = \frac{R_{\mathrm{t},B^+} K_{B^+} \exp\left(-b_{B^+} \overline{V}\right)}{\mathrm{F} D_{B^+}} \left(- \frac{\widetilde{c}_{B^+}(0)}{\left. \frac{\mathrm{d}\widetilde{c}_{B^+}}{\mathrm{d}y} \right|_{y=0}} \right) \tag{4–33}$$

The concentration distributions required to assess the diffusion impedance, equation (4–33), were obtained for each frequency from the numerical solution of equations (4–21)-(4–24).

The dimensionless diffusion impedance is

$$-\frac{1}{\theta'_{B^+}} = \frac{1}{\delta_{\mathrm{N},B^+}} \left(- \frac{\widetilde{c}_{B^+}(0)}{\left. \frac{\mathrm{d}\widetilde{c}_{B^+}}{\mathrm{d}y} \right|_{y=0}} \right) \tag{4–34}$$

where

$$\delta_{\mathrm{N},B^+} = \Gamma(4/3) \left(\frac{3}{\mathrm{a}} \right)^{1/3} \frac{1}{\mathrm{Sc}_{B^+}^{1/3}} \sqrt{\frac{\nu}{\Omega}} \tag{4–35}$$

is the Nernst diffusion layer thickness written in terms of the Schmidt number, $\mathrm{Sc}_{B^+} = \nu / D_{B^+}$.

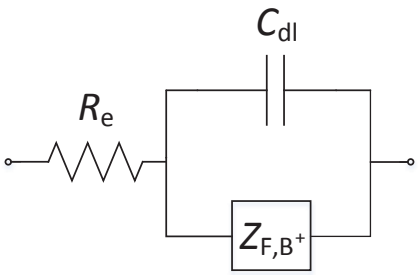


Figure 4-2. Electrical circuit representation of the overall electrode impedance associated with the faradaic impedance given as equation (4–31).

4.1.4.2 Overall impedance

The overall impedance was expressed by the circuit shown in Figure 4-2. An ohmic resistance is in series with the parallel contribution of the faradaic impedance, equation (4–31), and double-layer capacitance, i.e., ,

$$Z = R_e + \frac{Z_{F,B^+}}{1 + j\omega Z_{F,B^+} C_{dl}} \tag{4–36}$$

For a range of potentials, the faradaic impedance and double-layer capacitance are of the same magnitude and add another loop to the overall impedance. This means that three characteristic frequencies can be visible in the overall impedance.

4.1.4.3 Fast homogeneous reaction

In the limit of an infinitely fast homogeneous reaction, reactions (4–3) and (4–4) may be expressed as

$$AB + e^- \rightarrow A^- + B \tag{4–37}$$

and a dimensionless diffusion impedance can now be calculated by taking into account the oscillating concentration of species AB

$$-\frac{1}{\theta'_{B^+}} = \frac{1}{\delta_{N,B^+}} \left(-\frac{\widetilde{c}_{AB}(0)}{\left.\frac{d\widetilde{c}_{AB}}{dy}\right|_{y=0}} \right) \tag{4–38}$$

The electroactive species for the system is AB when the homogeneous reaction is infinitely fast.

4.1.4.4 Gerischer impedance

Under the assumptions that the diffusion coefficients for AB and B^+ are equal, that convection may be ignored, and that the concentration of A^- is sufficiently large to be considered constant, Gerischer developed an analytic expression for the diffusion impedance associated with a heterogeneous reaction influenced by a homogeneous reaction. This derivation makes use of the Nernst stagnant diffusion layer, equation (4–35). The concentration of A^- was considered invariant, and the rate of production of species AB and B^+ by reaction (4–3) was expressed as

$$R_{B^+} = -R_{AB} = k_f c_{AB}(y) - k_b c_{B^+}(y) \tag{4–39}$$

The conservation equations for species AB and B^+ may be expressed as

$$\frac{\partial c_{AB}}{\partial t} = D\frac{\partial^2 c_{AB}}{\partial y^2} - k_f c_{AB} + k_b c_{B^+} \tag{4–40}$$

and

$$\frac{\partial c_{B^+}}{\partial t} = D\frac{\partial^2 c_{B^+}}{\partial y^2} + k_f c_{AB} - k_b c_{B^+} \tag{4–41}$$

respectively, where

$$D = D_{AB} = D_{B^+} \tag{4–42}$$

The sum of equations (4–40) and (4–41) yields

$$\frac{\partial}{\partial t}\left(c_{AB} + c_{B^+}\right) = D\frac{\partial^2}{\partial y^2}\left(c_{AB} + c_{B^+}\right) \tag{4–43}$$

After algebraic manipulation, the difference between equations (4–40) and (4–41) may be expressed as

$$\frac{\partial}{\partial t}\left(c_{AB} - \frac{c_{B^+}}{K_{eq}}\right) = D\frac{\partial^2}{\partial y^2}\left(c_{AB} - \frac{c_{B^+}}{K_{eq}}\right) - k\left(c_{AB} - \frac{c_{B^+}}{K_{eq}}\right) \tag{4–44}$$

where $k = k_\mathrm{f} + k_\mathrm{b}$ and $K_\mathrm{eq} = k_\mathrm{f}/k_\mathrm{b}$. As equations (4–43) and (4–44) are linear, the solution for the convective-diffusion impedance does not require a solution for the steady-state. Equations (4–43) and (4–44) in the frequency domain were solved for the boundary conditions

$$\widetilde{c}_\mathrm{AB}(\delta) = \widetilde{c}_\mathrm{B^+}(\delta) = 0 \tag{4–45}$$

$$\widetilde{c}_\mathrm{B^+}(0) = 1 \tag{4–46}$$

and

$$\left.\frac{\mathrm{d}\widetilde{c}_\mathrm{AB}}{\mathrm{d}y}\right|_{y=0} = 0 \tag{4–47}$$

The dimensionless diffusion impedance may be expressed as

$$-\frac{1}{\theta'_\mathrm{B^+}} = -\frac{1}{\delta}\frac{\widetilde{c}_\mathrm{B^+}(0)}{\left.\dfrac{\mathrm{d}\widetilde{c}_\mathrm{B^+}}{\mathrm{d}y}\right|_{y=0}} \tag{4–48}$$

$$= \frac{1}{K_\mathrm{eq}+1}\frac{\tanh\sqrt{(\mathrm{j}\omega + k)\dfrac{\delta^2}{D}}}{\sqrt{(\mathrm{j}\omega + k)\dfrac{\delta^2}{D}}} + \frac{K_\mathrm{eq}}{K_\mathrm{eq}+1}\frac{\tanh\sqrt{\mathrm{j}\omega\dfrac{\delta^2}{D}}}{\sqrt{\mathrm{j}\omega\dfrac{\delta^2}{D}}} \tag{4–49}$$

or

$$-\frac{1}{\theta'_\mathrm{B^+}} = \frac{1}{K_\mathrm{eq}+1}\frac{\tanh\sqrt{\mathrm{j}K + k_\mathrm{dim}}}{\sqrt{\mathrm{j}K + k_\mathrm{dim}}} + \frac{K_\mathrm{eq}}{K_\mathrm{eq}+1}\frac{\tanh\sqrt{\mathrm{j}K}}{\sqrt{\mathrm{j}K}} \tag{4–50}$$

where $K = \omega\delta^2/D$ and $k_\mathrm{dim} = k\delta^2/D$.

The first term on the right-hand side of equation (4–50) corresponds to a weighted Gerischer impedance. The second term on the right-hand side of equation (4–50) corresponds to a weighted diffusion impedance through a film or in an electrolyte where there is no convection.

The convective-diffusion impedance with homogeneous reaction can be modeled by an expression using a function similar to equation (4–50), a weighted Gerischer impedance in series with a weighted convective-diffusion impedance.

$$-\frac{1}{\theta\prime_\mathrm{B^+}} = \frac{1}{K_\mathrm{eq}+1}\frac{\tanh\sqrt{\mathrm{j}K + k_\mathrm{dim}}}{\sqrt{\mathrm{j}K + k_\mathrm{dim}}} + \frac{1}{K_\mathrm{eq}+1}\frac{1}{\theta\prime_\mathrm{CD}} \tag{4–51}$$

where $1/\theta\iota_{CD}$ can be obtained using $K = \omega\delta_N^2/D_{B^+}$ and a look up table generated from the solution of the convective-diffusion impedance without homogeneous reaction.

4.1.5 Numerical Methods

The coupled nonlinear differential equations were solved first under the steady-state condition. Then the oscillating condition was solved with the steady-state results as an input. All the equations were linearized, formulated in finite difference form and solved numerically using Newman's BAND method coupled with Newton–Raphson iteration [74].

For input values shown in Tables 4-1 and 4-2, the steady-state concentrations of AB, A^-, and B^+ as well as the value of the homogeneous reaction rate were obtained. These steady-state values along with the original input values were used to obtain the oscillating values on concentration of the reacting species B^+. For a spectrum of dimensionless frequencies, from $K = 1 \times 10^{-5}$ to 1×10^5 with 20 points per decade, the oscillating concentrations of the reacting species obtained near the electrode were imputed into equation (4–34) to obtain a dimensionless diffusion impedance and equation (4–33) to obtain a diffusion impedance.

4.1.5.1 Accuracy of numerical methods

To obtain a solution that is the most accurate, the finite difference errors and round-off errors need to be minimized. Near the electrode surface the flux of the reacting species is changing so dramatically it is necessary to have a smaller mesh to obtain an accurate answer. Far from the electrode the concentrations are not changing and therefore the same small mesh would result in unnecessary roundoff errors. To mitigate errors we have solved the system using two mesh sizes. A small mesh size, HH, is used near the electrode and a large mesh size, H, is used far away from the electrode.

With the use of two dissimilar mesh sizes, it is necessary that a method of coupling be developed to allow the computation transition from one region to another while retaining the accuracy of the finite difference calculation. Such transition is performed by a coupling subroutine at an interface designated at j = KJ. A visual representation of two mesh size

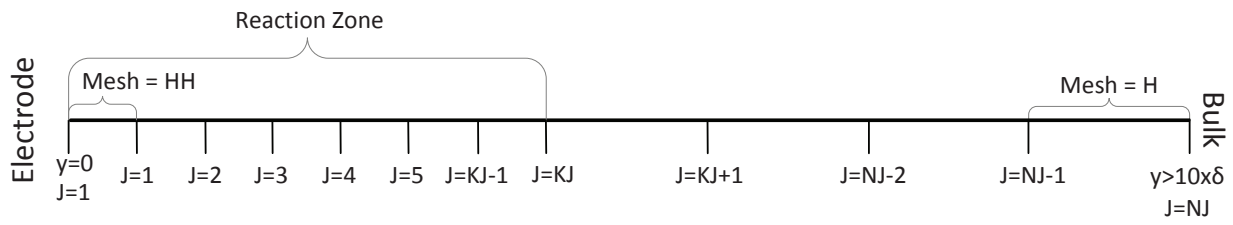


Figure 4-3. One-dimensional schematic representation showing two dissimilar mesh sizes. HH is the smaller mesh size and H is the larger mesh size. J=KJ is the interface region.

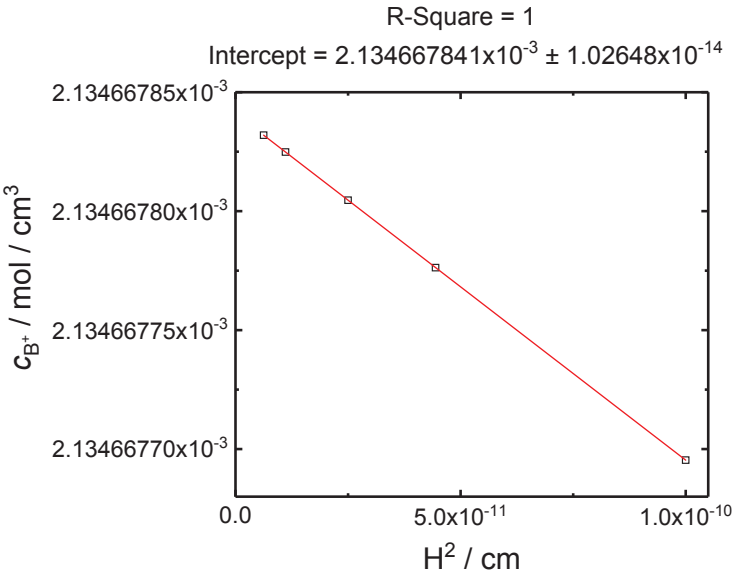


Figure 4-4. Graphical evidence of H^2 accuracy in finite difference simulation. The concentration of B^+ is shown vs different squared mesh values.

regions is shown in Figure 4-3. The small number of nodes, with NJ = 12, is exaggeratedly small to show the differences of mesh regions, the actual code uses NJ = 12,000 nodes.

A graph showing the accuracy of the simulations, which is on the order of the mesh size squared, is shown in Figure 4-4. The concentration of B^+ is shown with respect to different squared mesh values. The simulations for different mesh sizes, ranging from 3000 points to 12000 points, are accurate to seven orders of magnitude. The R^2 value of the linear fit is 1.0 and the intercept is value is calculated as $2.134667841 \times 10^{-3} \pm 1.0 \times 10^{-14}$.

4.1.5.2 Coupling domains with different mesh size

The different mesh sizes presented in Figure 4-3 need to be addressed differently then the bulk equations (4–17) to (4–19). The flux at the coupler point, j = KJ, should be equal when approach from the smaller mesh or the larger mesh

$$N_{\mathrm{i}}|_{\mathrm{KJ-1\rightarrow KJ}} = N_{\mathrm{i}}|_{\mathrm{KJ+1\rightarrow KJ}} \tag{4–52}$$

To accomplish this, the material balance equation was written for $KJ-1/4$ and $KC+1/4$

$$\left.\frac{dc_{\mathrm{i}}}{dt}\right|_{\mathrm{KJ-1/4}} = -\nabla \cdot N_{\mathrm{i}}|_{\mathrm{KJ-1/4}} + R_{\mathrm{i}}|_{\mathrm{KJ-1/4}} \tag{4–53}$$

and

$$\left.\frac{dc_{\mathrm{i}}}{dt}\right|_{\mathrm{KJ+1/4}} = -\nabla \cdot N_{\mathrm{i}}|_{\mathrm{KJ+1/4}} + R_{\mathrm{i}}|_{\mathrm{KJ+1/4}} \tag{4–54}$$

The derivative of flux written in cylindrical coordinates, assuming axisymmetric flow, is

$$\nabla \cdot \mathbf{N}_{\mathrm{i}} = \frac{1}{r}\frac{\partial(r\mathbf{N}_{\mathrm{r,i}})}{\partial r} + \frac{\partial(\mathbf{N}_{\mathrm{y,i}})}{\partial y} \tag{4–55}$$

re-writing the material balance equation in a difference of fluxes gives

$$\left.\frac{dc_{\mathrm{i}}}{dt}\right|_{\mathrm{KJ-1/4}} = -\left(\frac{1}{r}\frac{\partial(r\mathbf{N}_{\mathrm{r,i}})}{\partial r}\right)\Bigg|_{\mathrm{KJ-1/4}} - \left[\frac{N_{\mathrm{i}}|_{\mathrm{KJ}} - N_{\mathrm{i}}|_{\mathrm{KJ-1/2}}}{\mathrm{HH}/2}\right] + R_{\mathrm{i}}|_{\mathrm{KJ-1/4}} \tag{4–56}$$

and

$$\left.\frac{dc_{\mathrm{i}}}{dt}\right|_{\mathrm{KJ+1/4}} = -\left(\frac{1}{r}\frac{\partial(r\mathbf{N}_{\mathrm{r,i}})}{\partial r}\right)\Bigg|_{\mathrm{KJ+1/4}} - \left[\frac{N_{\mathrm{i}}|_{\mathrm{KJ+1/2}} - N_{\mathrm{i}}|_{\mathrm{KJ}}}{\mathrm{H}/2}\right] + R_{\mathrm{i}}|_{\mathrm{KJ+1/4}} \tag{4–57}$$

Rearranging equations (4–56) and (4–57) for $N_{\mathrm{i}}|_{\mathrm{KJ}}$ and then setting them equal to each other provides the necessary relationship for the coupler, when $j = KJ$.

4.2 Impedance for Convective-Diffusion and Homogeneous Reaction

The polarization curve corresponding to a system with an electrochemical reaction influenced by a homogeneous reaction, with the given parameters in Tables 4-1 and 4-2, is presented in Figure 4-5. The heterogeneous reaction rate increases as the potential becomes more negative, reaching a mass-transfer-limited plateau for potentials smaller

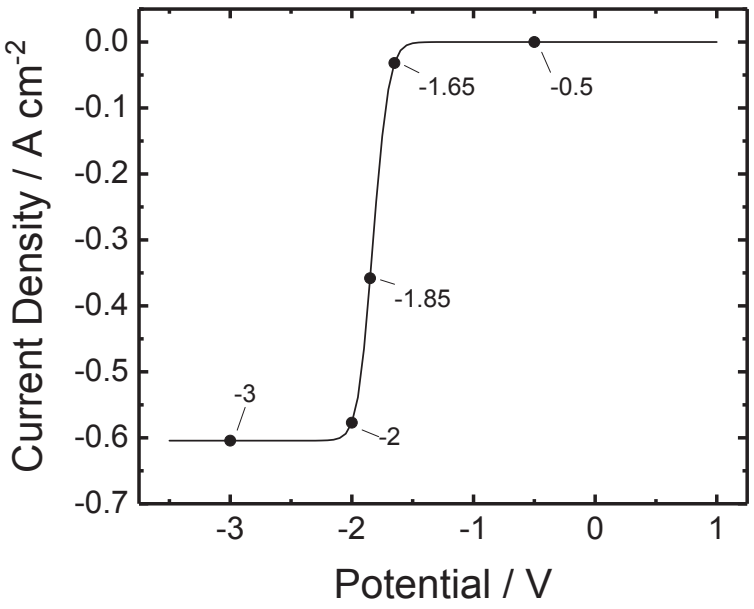


Figure 4-5. Polarization curve calculated for system parameters presented in tables 4-1 and 4-2. Labeled potential values correspond to steady-state concentration profiles presented in Figure 4-6.

Table 4-1. Species and associated parameter values for the system

Species	$c_{\mathrm{i}}(\infty)$, mol/cm^3	z_{i}	D_{i}, cm^2/s
AB	0.01	0	1.684×10^{-5}
A^-	0.0001	-1	1.957×10^{-5}
B^+	0.0001	1	1.902×10^{-5}

than -2 V. The large magnitude of the mass-transfer-limited current density may be attributed to the production of B^+ by the homogeneous reaction.

Labeled potential values in Figure 4-5 correspond to steady-state concentration profiles and reaction profile presented in Figure 4-6. Concentrations were scaled by the corresponding values at $y \to \infty$ to emphasize the relative changes in values. The reaction

Table 4-2. System and kinetic parameter values for the system

Parameter	Value	Units
Disk rotation rate, Ω	2,000	rpm
Kinematic viscosity, ν	0.01	cm^2/s
Homogeneous equilibrium constant, K_{eq}	10^{-6}	mol/cm^3
Homogeneous rate constant, k_{b},	10^7	cm^3/mols
Heterogeneous rate constant, K_{B^+}	2×10^{-12}	A/cm^2
Heterogeneous constant, b_{B^+}	19.9	V^{-1}

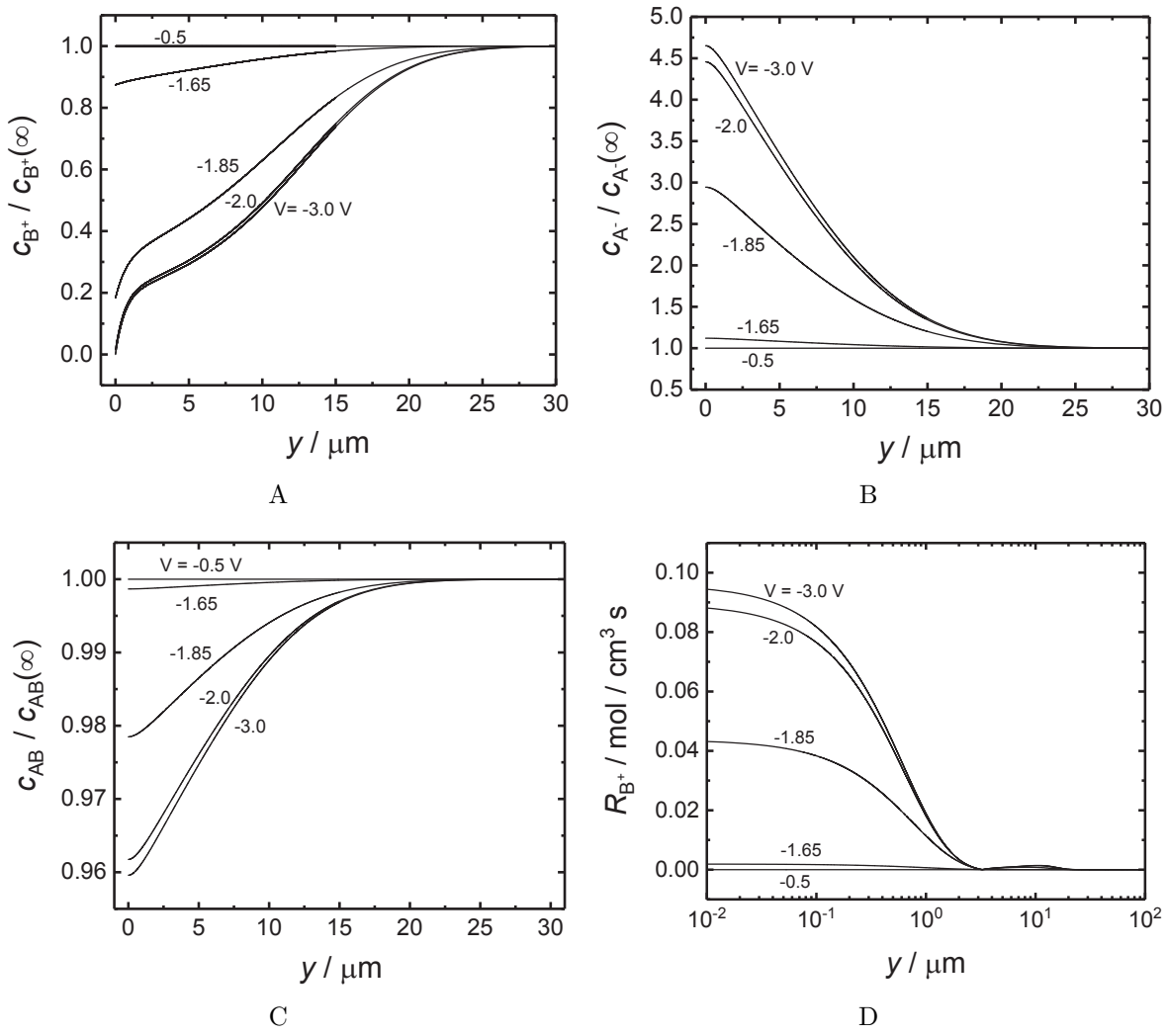


Figure 4-6. Calculated steady-state concentration distributions and homogeneous reaction corresponding to system parameters presented in Tables 4-1 and 4-2; a) B^+ b) A^- c) AB c) R_{B+}

was plotted against a logarithmic distance away from the electrode to emphasize the shape of the reaction profile. The influence of the homogeneous production of B^+ on the polarization curve can be seen in Figure 4-7A.

The convective-diffusion impedance corresponding to the steady-state results presented in Figure 4-6 is presented in Figure 4-8 with applied potential as a parameter.

The overall impedance can be calculated from the simulated diffusion impedance using a circuit model, Figure 4-2. An overall impedance with an ohmic resistance of 10

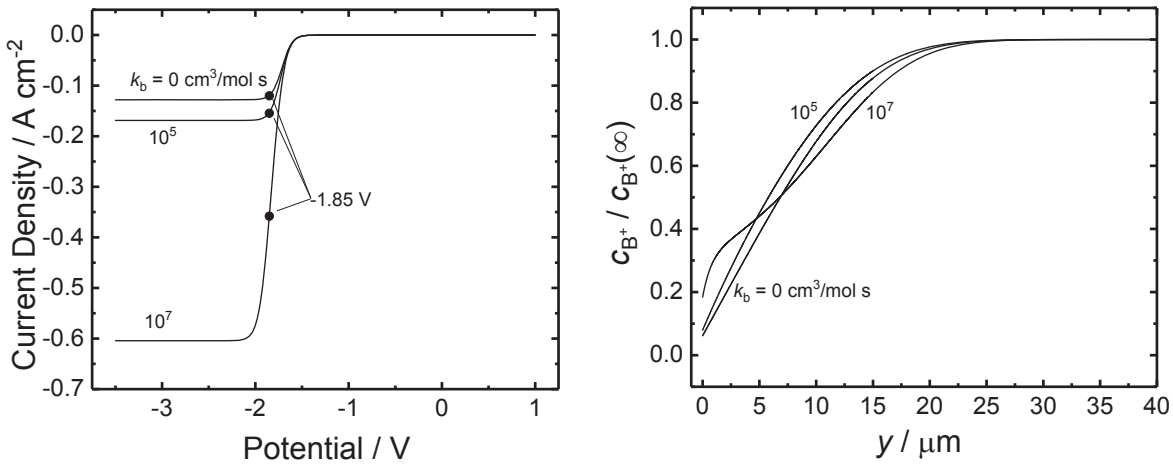


Figure 4-7. Calculations corresponding to system parameters presented in Tables 4-1 and 4-2 with homogeneous rate constant k_b as a parameter: a) polarization curve b) concentration distribution for B^+ at a potential of $V = -1.85$ V.

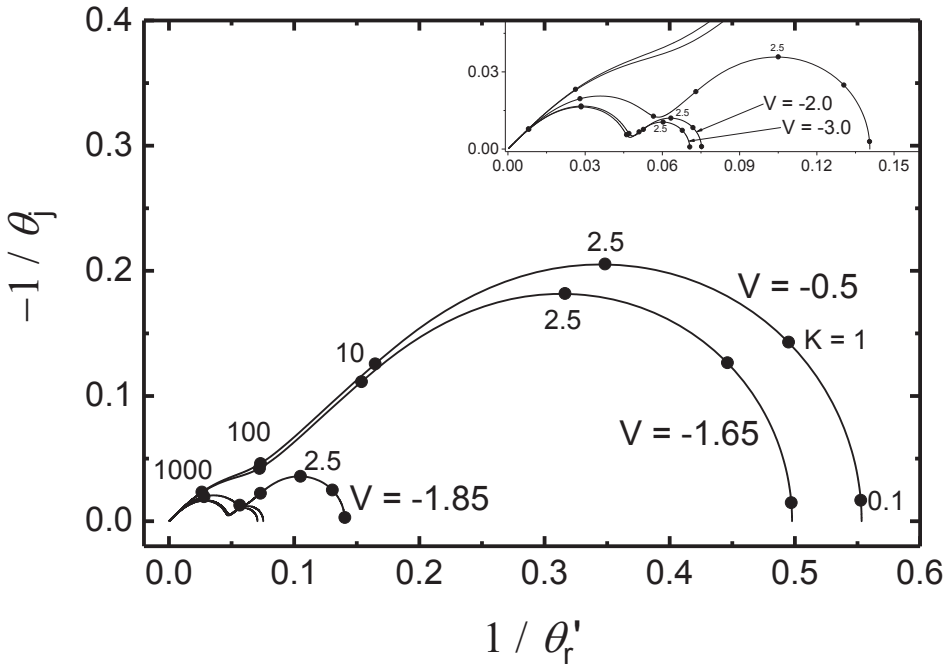


Figure 4-8. Dimensionless convective-diffusion impedance for the system presented in Figure 4-6 with applied potential as a parameter.

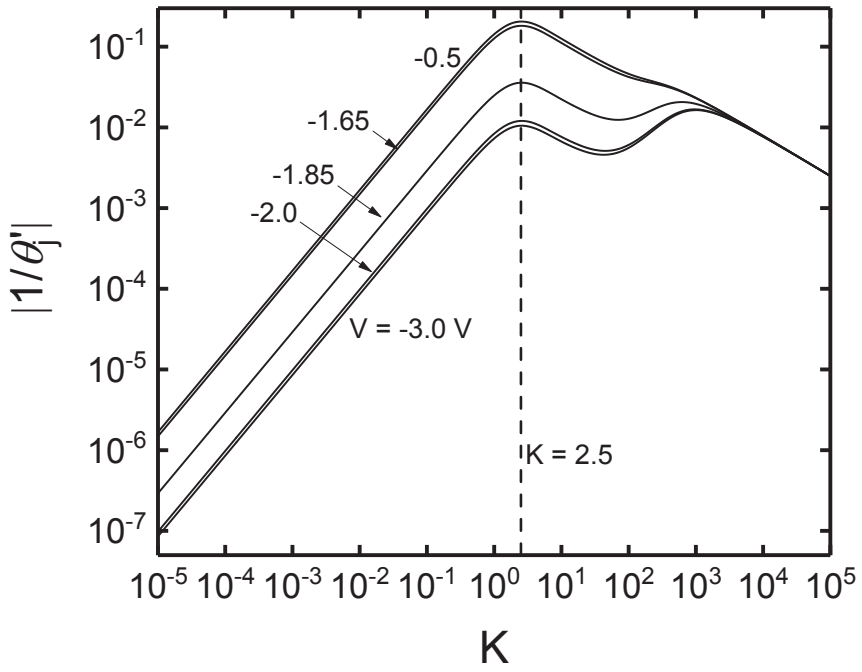


Figure 4-9. The imaginary part of the dimensionless convective-diffusion impedance for the system presented in Figure 4-6 as a function of dimensionless frequency with applied potential as a parameter.

Ωcm^2 in series with a double-layer capacitance of 20 $\mu F/$cm^2 in parallel with the diffusion impedance in Figure 4-8 is shown in Figure 4-10.

If the homogeneous reaction is infinitely fast we obtain the dimensionless diffusion impedance using the oscillating concentration of species c_{AB}, using equation (4–37). In this case, only one loop is visible in the convective-diffusion impedance plots corresponding to the diffusion of AB to the electrode surface, shown in Figure 4-11.

4.3 Discussion for Convective Diffusion and Homogeneous Reaction

The concentration of AB, shown in Figure 4-6C, decreases to a value that is 96 percent of the bulk value as potential approaches -3 V. In contrast, the concentration of A^- shown in Figure 4-6B reaches a value that is almost 5 times its bulk value at the mass-transfer-limited current density. The normalized concentration distribution of B^+ is presented in Figure 4-6A. The concentration of B^+ at the electrode surface approaches a value of zero as the mass–transfer–limited current density is approached. The sharp profile

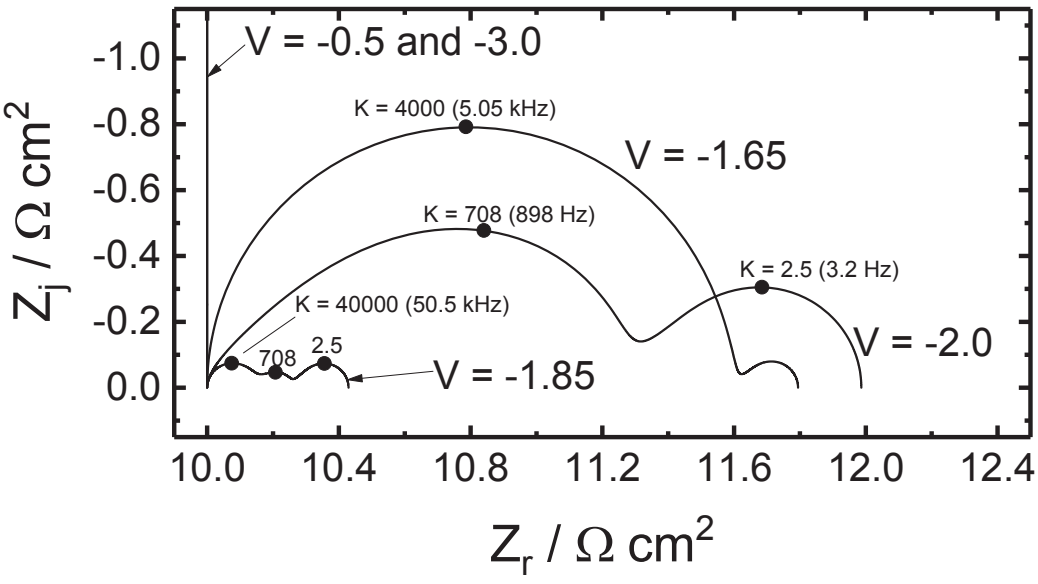


Figure 4-10. The overall impedance using the diffusion impedance from Figure 4-8 with ohmic resistance of 10 Ωcm^2 and a double-layer capacitance of 20 μF/cm^2.

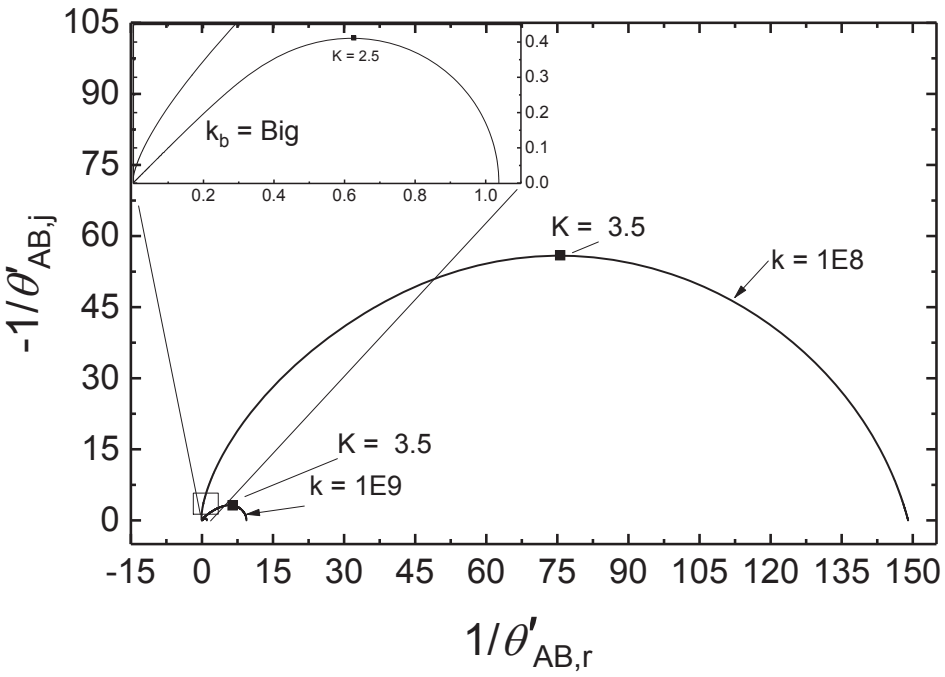


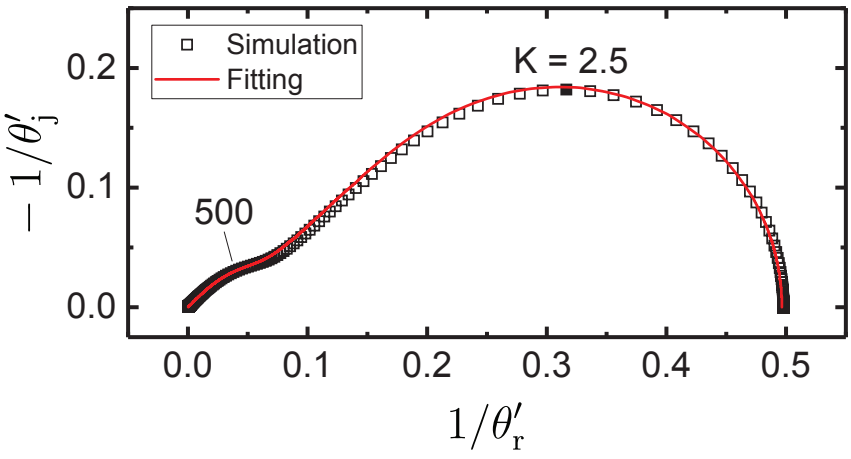
Figure 4-11. Dimensionless convective-diffusion impedance where the oscillating concentration of AB were taken into account in equation (4–38) and large values of the homogeneous reaction rate constant were used.

that appears close to the electrode surface in Figure 4-6A is consistent with a large current density.

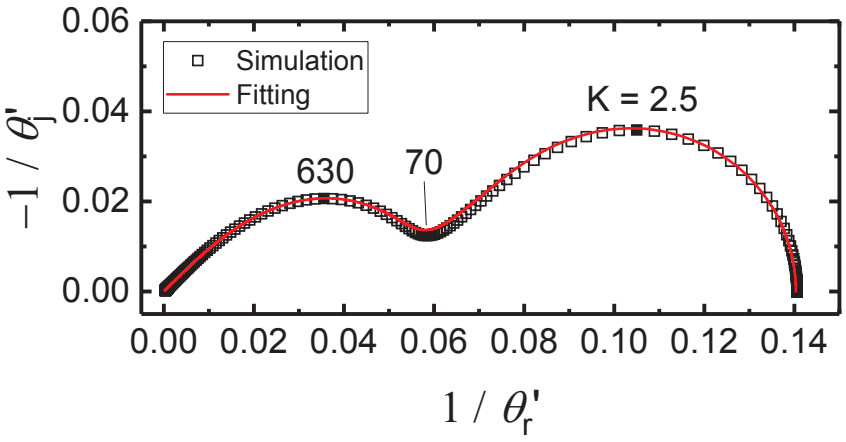
The influence of the homogeneous reaction can be seen in Figure 4-7A, where a homogeneous rate constant of $k_b = 10^7 \text{cm}^3/\text{mols}$ yields a mass-transfer-limited current density that is 4.7 times larger than that in the absence of homogeneous reactions. The concentration of B^+ is presented in Figure 4-7B as a function of position. In the absence of homogeneous reaction, the concentration profile resembles a traditional convective-diffusion impedance. The slope at the electrode-electrolyte interface becomes larger as the homogeneous rate constant increases. For $k_b = 10^7 \text{cm}^3/\text{mols}$ the steeper slope at the electrode corresponds to a smaller slope at intermediate values of y.

As compared to the dimensionless diffusion impedance without homogeneous reactions, for finite Schmidt numbers and in the absence of homogeneous reactions, the diffusion impedance presented in Figure 4-8 is smaller in magnitude and decreases as the rate of the heterogeneous consumption of B^+ increases. Also, two asymmetric capacitive loops are seen in Figure 4-8 as compared to a single loop in traditional convective-diffusion impedance. The low-frequency loop has a characteristic frequency $K = 2.5$, which is in agreement with the characteristic frequency associated with diffusion in the absence of homogeneous reactions. The high-frequency loop can then be associated with the homogeneous reaction. The characteristic frequency for the high-frequency loop is on the order of $K = 1000$, suggesting that the characteristic dimension for the reaction is much smaller than the Nernst diffusion-layer thickness.

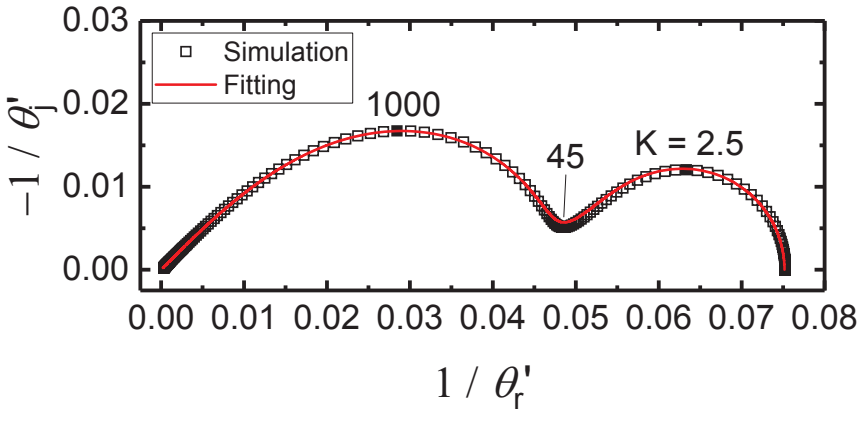
The overall impedance for -1.65 V shows a capacitive loop and a smaller diffusion loop. The overall impedance of -2.0 V shows a faradic impedance taking over and the two loops correspond to a homogeneous reaction loop and a tradition diffusion loop. A very interesting thing happens at about half the mass-transfer-limiting potential, -1.85 V. At this point there were similar contributions from both the capacitive and the faradaic impedances and there were three loops in the overall impedance. The first loop is



A



B



C

Figure 4-12. Simulated dimensionless convective-diffusion impedance and regression fitting using equation (4–51) for $k_b = 10^7$ and a) $V = -1.65$V b) $V = -1.85$V c) $V = -2.0$V.

Table 4-3. Fitting parameters found from regression using equation (4–51) and Figure 4-12.

Potential	k_{dim}, dimensionless	K_{eq}, dimensionless
-1.6V	246 ± 6.3	$0.859 \pm 1.9 \times 10^{-3}$
-1.85V	334 ± 2.2	$9.96 \times 10^{-2} \pm 2.4 \times 10^{-4}$
-2.0V	467 ± 2.6	$3.12 \times 10^{-2} \pm 1.6 \times 10^{-4}$

attributed to the capacitance, the second loop to the homogeneous reaction and the third loop to convective-diffusion impedance.

In the case of an infinitely fast homogeneous reaction, Figure 4-11, we see one loop in the convective-diffusion impedance. As the homogeneous reaction gets larger the shape of the distorted semi-circle becomes a perfect dimensionless convective-diffusion impedance. The low-frequency limit goes to 1.0392 and the high-frequency limit goes to the origin at at 45 degree angle.

A Levenberg–Marquardt regression was used in Origin Lab 2017 to regress equation (4–51) and show a fit to the data shown in Figure 4-8. The diffusion impedances fitted were $V = -1.65$V in Figure 4-12A, $V = -1.85$V in Figure 4-12B, and $V = -2.0$V in Figure 4-12C. The fitting parameters are listed in Table 4-3.

Showing the impedances in Figure 4-12 in a bode plot can aid in seeing how good the model fits, Figure

Using the definition of $k_{\text{dim}} = k\delta^2/D$ and a following Levich [21] we can obtain a relationship between the reaction thickness, δ_{r}, and the Nernst diffusion layer thickness

$$\delta_{\text{r}} = \delta_{\text{N,B}^+}\sqrt{\frac{1}{k_{\text{dim}}}} \tag{4–58}$$

The reaction thicknesses obtained from equation (4–58) are shown in Figure 4-14. The reaction distribution was scaled by the corresponding values at $y \to 0$ to emphasize the relative changes in values. As the heterogeneous reaction becomes bigger the reaction thickness decreases.

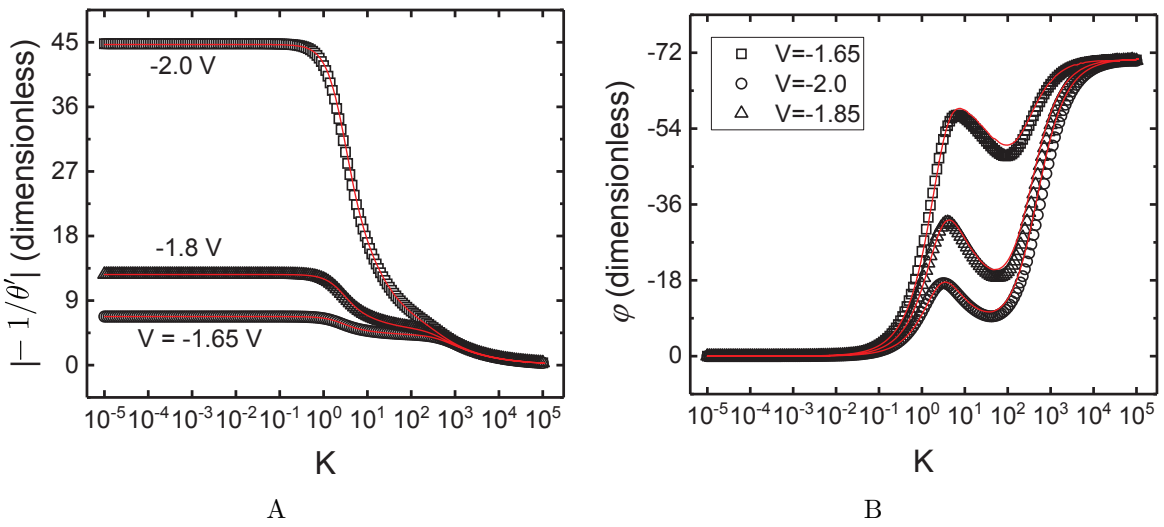


Figure 4-13. The simulated dimensionless convective-diffusion impedance and regression fitting using equation (4–51) for $k_b = 10^7$ for three potentials shown in a) Magnitude and b) Phase.

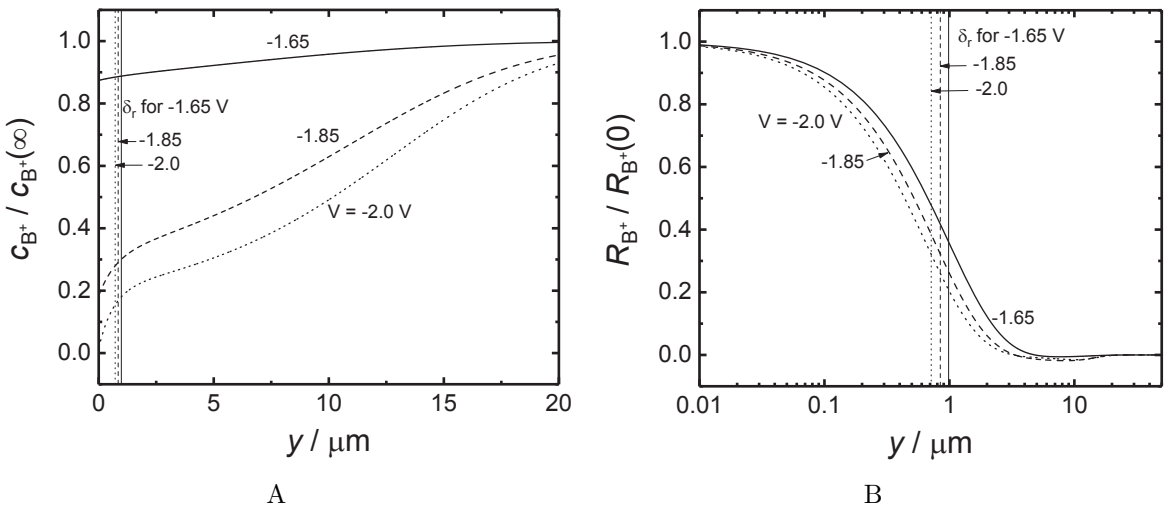


Figure 4-14. Calculated reaction thickness shown over a) concentration distribution for B^+ b) reaction distribution. for potentials that were fitted in Figure 4-12.

CHAPTER 5
IMPEDANCE RESPONSE FOR CONTINUOUS GLUCOSE MONITORS

A mathematical model was developed for the impedance response of a glucose oxidase enzyme-based electrochemical biosensor. A schematic of the glucose sensor inserted into the interstitial fluid is presented in Figure 5-1. The model accounts for a glucose limiting membrane GLM, which controls the amount of glucose participating in the enzymatic reaction. Between the GLM and the electrode is the enzyme glucose oxidase layer (GOX). The glucose oxidase was assumed to be immobilized within a thin film adjacent to the electrode. In the glucose oxidase layer, a process of enzymatic catalysis transforms the glucose into hydrogen peroxide, which can be detected electrochemically, presented in Figure 5-2.

This system was considered to be a special case of the coupled homogeneous and heterogeneous reactions addressed by Levich [21]. The model development required two steps. The nonlinear coupled differential equations governing this system were solved under the assumption of a steady state. The steady-state concentrations resulting from the steady-state simulation were used in the solution of the linearized set of differential equations describing the sinusoidal steady state. The enzymatic catalysis was treated in terms of two homogeneous reactions, one consuming the glucose oxidase and forming gluconic acid, and the other regenerating the glucose oxidase and forming the peroxide. Witt et al. [81] came up with a representation of the GOX reaction. The reaction of D-glucono-δ-lactone to gluconic acid is considered to be extremely fast and therefore is not represented in this model.

5.1 Mathematical Development for the Continuous Glucose Monitor

A mathematical model is presented below for the impedance response associated with a CGM including the coupling of the enzymatic and electrochemical reactions. The glucose oxidase layer (GOX) over the electrode was considered to be 7 μm thick and the

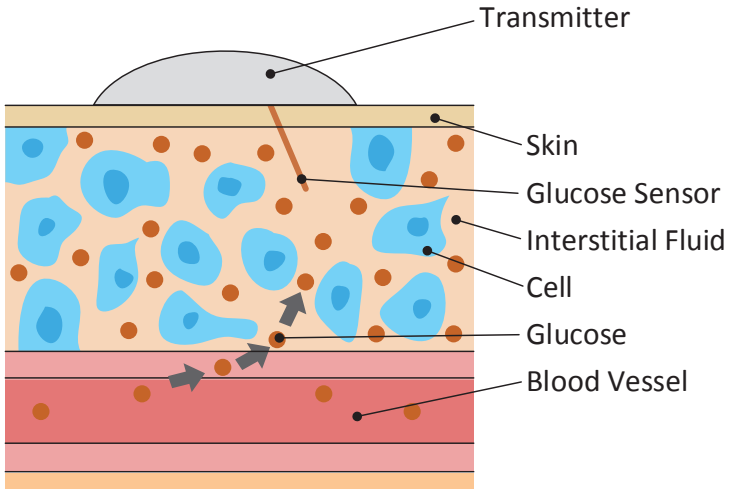


Figure 5-1. Schematic showing glucose sensor inserted under the skin into the interstitial fluid. Glucose diffuses through the blood vessels into the interstitial fluid where it can be measured with the glucose sensor.

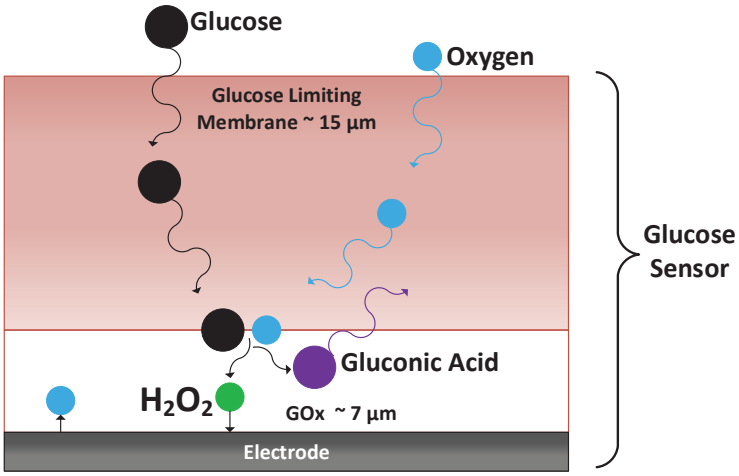


Figure 5-2. Layers in the Glucose Sensor, the glucose limiting membrane (GLM) limits the amount of glucose from the interstitial fluid to diffuse to the glucose oxidase layer (GOX). The glucose reacts with the GOX layer with oxygen to form gluconic acid and hydrogen peroxide. The hydrogen peroxide reacts electrochemically and the current of the electrochemical reaction is proportional to the concentration of glucose.

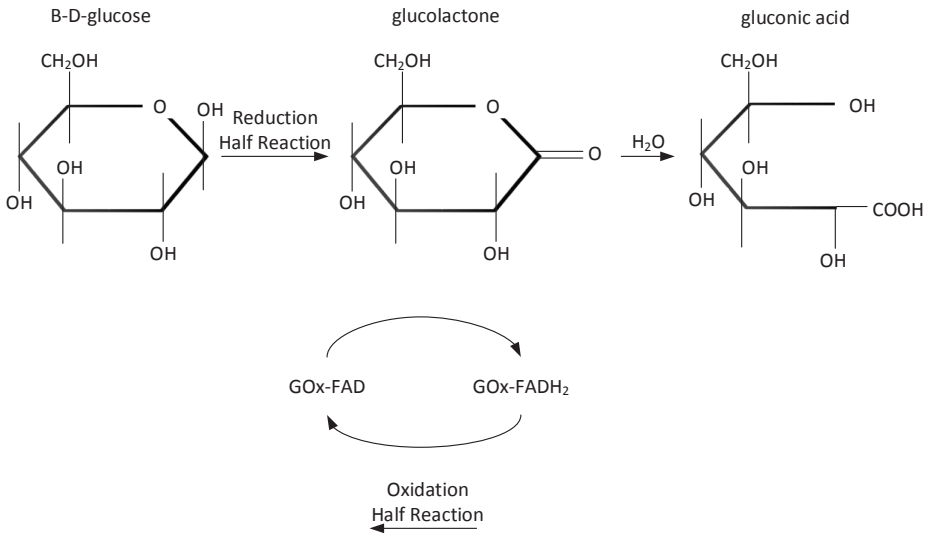


Figure 5-3. Representation of the glucose oxidase reaction. The product of the first enzymatic reaction is D-glucono-δ-lactone which combines chemically with water to form gluconic acid. The product of the second enzymatic reaction is hydrogen peroxide. Recreated from Witt et al. [81]. The model used in this dissertation assumes an enzyme complex is formed and the formation step is reversible, and the reaction of the enzyme complex to the products is irreversible.

glucose limiting membrane layer (GLM) on top of the enzyme layer was considered to be 15 μm thick.

5.1.1 Governing Equations for CGM System

The flux density of species i in a dilute electrolyte and in absence of migration and convection was expressed as

$$\mathbf{N}_\mathrm{i} = -D_\mathrm{i}\nabla c_\mathrm{i} \tag{5–1}$$

For a system where the homogeneous reaction occurs in a hydrogel the convective diffusion equation with enzymatic reaction was expressed as

$$\frac{\partial c_\mathrm{i}}{\partial t} = D_\mathrm{i}\frac{\partial^2 c_\mathrm{i}}{\partial y^2} + R_\mathrm{i} \tag{5–2}$$

where R_i is the rate of production of species i by homogeneous reactions. The overall enzymatic (homogeneous) reaction was

$$\mathrm{G} + \mathrm{GOx} \underset{k_{b1}}{\overset{k_{f1}}{\rightleftarrows}} \mathrm{GOx2} - \mathrm{GA} \xrightarrow{k_{f2}} \mathrm{GA} + \mathrm{GOx2} \tag{5–3}$$

which represents the enzymatic catalysis reaction and

$$\mathrm{GOx2} + \mathrm{O_2} \underset{k_{b3}}{\overset{k_{f3}}{\rightleftarrows}} \mathrm{GOx} - \mathrm{H2O2} \xrightarrow{k_{f4}} \mathrm{GOx} + \mathrm{H2O2} \tag{5–4}$$

which reporesents the mediation/regeneration reaction. The species are defined in Table 5-1.

5.1.2 Homogeneous Enzymatic Reactions for CGM System

The enzymatic homogeneous reactions, from equations (5–3) and (5–4), were expressed as individual reactions as

$$\mathrm{G} + \mathrm{GOx} \underset{k_{b1}}{\overset{k_{f1}}{\rightleftarrows}} \mathrm{GOx2} - \mathrm{GA} \tag{5–5}$$

$$\mathrm{GOx2} - \mathrm{GA} \xrightarrow{k_{f2}} \mathrm{GA} + \mathrm{GOx2} \tag{5–6}$$

$$\mathrm{GOx2} + \mathrm{O_2} \underset{k_{b3}}{\overset{k_{f3}}{\rightleftarrows}} \mathrm{GOx} - \mathrm{H2O2} \tag{5–7}$$

and

$$\mathrm{GOx} - \mathrm{H2O2} \xrightarrow{k_{f4}} \mathrm{GOx} + \mathrm{H2O2} \tag{5–8}$$

These equations represent the enzymatic reaction of the glucose oxidase enzyme turning glucose into hydrogen peroxide. One mole of glucose and one molecule of oxygen create one mole of gluconic acid as a byproduct and one mole of hydrogen peroxide as the desired

Table 5-1. Species and associated parameter values for the system

Species	$c_i(\infty)$, mol/cm^3	D_i, cm^2/s	Abbreviation
Glucose ($C_6H_{12}O_6$)	5.55075×10^{-6}	7.2×10^{-6}	G
Glucose Oxidase (GOx-FAD)	0.5×10^{-3}	0	GOx
Gluconic Acid ($C_6H_{12}O_7$)	1.0×10^{-20}	7.2×10^{-6}	GA
Enzyme Complex ($GOx - FADH_2 - GA$)	0.5×10^{-3}	0	GOx2-GA
Glucose Oxidase ($GOx - FADH_2$)	0.5×10^{-3}	1.0×10^{-20}	GOx2
Oxygen (O_2)	3.125×10^{-9}	2.46×10^{-5}	O2
Enzyme Complex2 ($GOx - FAD - H_2O_2$)	1.902×0.5^{-3}	0	GOx-H2O2
Hydrogen Peroxide (H_2O_2)	0.5×10^{-3}	0	H2O2

product. Each species and their bulk values, diffusion coefficients, and an abbreviation used in the mathematical workup are shown in Table 5-1.

Hydrogen peroxide is electroactive and was consumed at the electrode. The electrochemical reaction was expressed as

$$H_2O_2 \rightarrow 2H^+ + O_2 + 2e^- \tag{5–9}$$

Oxygen was a product of the electrochemical reaction, which helps fuel the enzymatic reaction. The corresponding current, which is dependent on concentration and potential, was expressed as

$$i_{H_2O_2} = K_{H_2O_2} c_{H_2O_2}(0) \exp\left(b_{H_2O_2} V\right) \tag{5–10}$$

As the electroactive species is consumed, a concentration gradient of H_2O_2 must exist near the electrode surface.

The four reaction terms corresponding to reactions (5–5)-(5–8) were expressed as

$$R_1 = k_{f1} c_G(y) c_{GOx}(y) - k_{b1} c_{GOx2-GA}(y) \tag{5–11}$$

$$R_2 = k_{f2} c_{GOx2-GA}(y) \tag{5–12}$$

$$R_3 = k_{f3} c_{O2}(y) c_{GOx2}(y) - k_{b3} c_{GOx-H2O2}(y) \tag{5–13}$$

$$R_4 = k_{f4} c_{GOx-H2O2}(y) \tag{5–14}$$

The conservation equation for each species may be written as

$$\frac{\partial c_{\mathrm{G}}}{\partial t} = D_{\mathrm{G}} \frac{\partial^2 c_{\mathrm{G}}}{\partial y^2} - R_1 \tag{5–15}$$

$$\frac{\partial c_{\mathrm{GOx}}}{\partial t} = -R_1 + R_4 \tag{5–16}$$

$$\frac{\partial c_{\mathrm{GOx2-GA}}}{\partial t} = R_1 - R_2 \tag{5–17}$$

$$\frac{\partial c_{\mathrm{GA}}}{\partial t} = D_{\mathrm{GA}} \frac{\partial^2 c_{\mathrm{GA}}}{\partial y^2} + R_2 \tag{5–18}$$

$$\frac{\partial c_{\mathrm{O2}}}{\partial t} = D_{\mathrm{O2}} \frac{\partial^2 c_{\mathrm{O2}}}{\partial y^2} - R_3 \tag{5–19}$$

$$\frac{\partial c_{\mathrm{GOx2}}}{\partial t} = -R_3 + R_2 \tag{5–20}$$

$$\frac{\partial c_{\mathrm{GOx-H2O2}}}{\partial t} = R_3 - R_4 \tag{5–21}$$

$$\frac{\partial c_{\mathrm{H2O2}}}{\partial t} = D_{\mathrm{H2O2}} \frac{\partial^2 c_{\mathrm{H2O2}}}{\partial y^2} + R_4 \tag{5–22}$$

Equations (5–11)-(5–14), representing the reactions occurring in the system, along with equations (5–15)-(5–22), representing the concentrations of the various species are solved for the steady-state and frequency-domain variables. This solution follows that of section 4.1.2. The boundary conditions far from the electrode were

$$c_{\mathrm{i}} \to c_{\mathrm{i}}(\infty) \quad \text{for} \quad y \to \infty \tag{5–23}$$

and the boundary conditions at the electrode surface were

$$\left.\frac{\partial c_{\mathrm{i}}}{\partial y}\right|_{y=0} = 0 \quad \text{for} \quad y = 0 \tag{5–24}$$

for the non-reacting species, and

$$FD_{\mathrm{B+}}\left.\frac{\partial c_{\mathrm{H2O2}}}{\partial y}\right|_{y=0} = i_{\mathrm{H2O2}} \quad \text{for} \quad y = 0 \tag{5–25}$$

for the reacting species, hydrogen peroxide.

Splitting the concentration variables using equation (4–16) we can separate out the steady-state. The steady-state equations representing the non-enzymatic species were

$$\frac{\partial \overline{c}_{\mathrm{G}}}{\partial t} = D_{\mathrm{G}}\frac{\partial^2 \overline{c}_{\mathrm{G}}}{\partial y^2} - \overline{R}_1 \tag{5–26}$$

$$\frac{\partial \overline{c}_{\mathrm{GA}}}{\partial t} = D_{\mathrm{GA}}\frac{\partial^2 \overline{c}_{\mathrm{GA}}}{\partial y^2} + \overline{R}_2 \tag{5–27}$$

$$\frac{\partial \overline{c}_{\mathrm{O2}}}{\partial t} = D_{\mathrm{O2}}\frac{\partial^2 \overline{c}_{\mathrm{O2}}}{\partial y^2} - \overline{R}_3 \tag{5–28}$$

and

$$\frac{\partial \overline{c}_{\mathrm{H2O2}}}{\partial t} = D_{\mathrm{H2O2}}\frac{\partial^2 \overline{c}_{\mathrm{H2O2}}}{\partial y^2} + \overline{R}_4 \tag{5–29}$$

and the steady-state equations representing the enzymatic species and enzymatic complexes were

$$\frac{\partial \overline{c}_{\mathrm{GOx}}}{\partial t} = -\overline{R}_1 + \overline{R}_4 \tag{5–30}$$

$$\frac{\partial \overline{c}_{\mathrm{GOx2-GA}}}{\partial t} = \overline{R}_1 - \overline{R}_2 \tag{5–31}$$

$$\frac{\partial \overline{c}_{\mathrm{GOx2}}}{\partial t} = -\overline{R}_3 + \overline{R}_2 \tag{5–32}$$

and

$$\frac{\partial \overline{c}_{\mathrm{GOx-H2O2}}}{\partial t} = \overline{R}_3 - \overline{R}_4 \tag{5–33}$$

and the steady-state reaction equations were

$$\overline{R}_1 = k_{\mathrm{f1}}\overline{c}_{\mathrm{G}}(y)\overline{c}_{\mathrm{GOx}}(y) - k_{\mathrm{b1}}\overline{c}_{\mathrm{GOx2-GA}}(y) \tag{5–34}$$

$$\overline{R}_2 = k_{\mathrm{f2}}\overline{c}_{\mathrm{GOx2-GA}}(y) \tag{5–35}$$

$$\overline{R}_3 = k_{\mathrm{f3}}\overline{c}_{\mathrm{O2}}(y)\overline{c}_{\mathrm{GOx2}}(y) - k_{\mathrm{b3}}\overline{c}_{\mathrm{GOx-H2O2}}(y) \tag{5–36}$$

$$\overline{R}_4 = k_{\mathrm{f4}}\overline{c}_{\mathrm{GOx-H2O2}}(y) \tag{5–37}$$

All the derivatives with respect to time are equal to zero. And the reaction terms are equal to zero in the GLM layer. To satisfy BAND and the number of equations and variables, equations (5–30) - (5–33) are rewritten, and one became a mass balance equation, with the assumption that the total enzyme species concentration is constant.

$$0 = -\overline{R}_1 + \overline{R}_4 \tag{5–38}$$

$$\begin{aligned} 0 = {} & \overline{c}_{\mathrm{GOx}}(\infty) + \overline{c}_{\mathrm{GOx2-GA}}(\infty) + \overline{c}_{\mathrm{GOx2}}(\infty) + \overline{c}_{\mathrm{GOx-H2O2}}(\infty) \\ & -\overline{c}_{\mathrm{GOx}}(y) - \overline{c}_{\mathrm{GOx2-GA}}(y) - \overline{c}_{\mathrm{GOx2}}(y) - \overline{c}_{\mathrm{GOx-H2O2}} \end{aligned} \tag{5–39}$$

$$0 = \overline{R}_1 - \overline{R}_2 \tag{5–40}$$

and

$$0 = \overline{R}_3 - \overline{R}_4 \tag{5–41}$$

The form of equations (5–34) and (5–36) made the coupled steady-state differential equations non-linear. The steady-state equations were solved in BAND in a similar fashion as in Chapter 4. The oscillating equations were

$$\frac{\partial \widetilde{c}_{\mathrm{G}}}{\partial t} = D_{\mathrm{G}}\frac{\partial^2 \widetilde{c}_{\mathrm{G}}}{\partial \mathrm{y}^2} - \widetilde{R}_1 \tag{5–42}$$

$$\frac{\partial \widetilde{c}_{\mathrm{GA}}}{\partial t} = D_{\mathrm{GA}}\frac{\partial^2 \widetilde{c}_{\mathrm{GA}}}{\partial \mathrm{y}^2} + \widetilde{R}_2 \tag{5–43}$$

$$\frac{\partial \widetilde{c}_{\mathrm{O2}}}{\partial t} = D_{\mathrm{O2}}\frac{\partial^2 \widetilde{c}_{\mathrm{O2}}}{\partial \mathrm{y}^2} - \widetilde{R}_3 \tag{5–44}$$

and

$$\frac{\partial \widetilde{c}_{\mathrm{H2O2}}}{\partial t} = D_{\mathrm{H2O2}}\frac{\partial^2 \widetilde{c}_{\mathrm{H2O2}}}{\partial \mathrm{y}^2} - \widetilde{R}_4 \tag{5–45}$$

for the non-enzymatic concentrations and

$$\frac{\partial \widetilde{c}_{\mathrm{GOx}}}{\partial t} = -\widetilde{R}_1 + \widetilde{R}_4 \tag{5–46}$$

$$\frac{\partial \widetilde{c}_{\mathrm{GOx2-GA}}}{\partial t} = \widetilde{R}_1 - \widetilde{R}_2 \tag{5–47}$$

$$\frac{\partial \widetilde{c}_{\mathrm{GOx2}}}{\partial t} = -\widetilde{R}_3 + \widetilde{R}_2 \tag{5–48}$$

and

$$\frac{\partial \widetilde{c}_{\mathrm{GOx-H2O2}}}{\partial t} = \widetilde{R}_3 - \widetilde{R}_4 \tag{5–49}$$

for the enzymatic species and enzyme complexes. And the equations for the oscillating reaction terms were

$$\widetilde{R}_1 = k_{\mathrm{f1}}\widetilde{c}_{\mathrm{G}}(y)\overline{c}_{\mathrm{GOx}}(y) + k_{\mathrm{f1}}\overline{c}_{\mathrm{G}}(y)\widetilde{c}_{\mathrm{GOx}}(y) - k_{\mathrm{b1}}\widetilde{c}_{\mathrm{GOx2-GA}}(y) \tag{5–50}$$

$$\widetilde{R}_2 = k_{\mathrm{f2}}\widetilde{c}_{\mathrm{GOx2-GA}}(y) \tag{5–51}$$

$$\widetilde{R}_3 = k_{\mathrm{f3}}\widetilde{c}_{\mathrm{O2}}(y)\overline{c}_{\mathrm{GOx2}}(y) + k_{\mathrm{f3}}\overline{c}_{\mathrm{O2}}(y)\widetilde{c}_{\mathrm{GOx2}}(y)k_{\mathrm{b3}}\widetilde{c}_{\mathrm{GOx-H2O2}}(y) \tag{5–52}$$

$$\widetilde{R}_4 = k_{\mathrm{f4}}\widetilde{c}_{\mathrm{GOx-H2O2}}(y) \tag{5–53}$$

and terms with $\widetilde{c}^2$ were neglected. Equations (5–42) - (5–53) are coupled and linear but are dependent on the steady-state solution that shows up in equations (5–50) and (5–52). The boundary conditions for the oscillating concentrations were

$$\widetilde{c}_{\mathrm{i}} = 0 \quad \text{for} \quad y \to \infty \tag{5–54}$$

for each species

$$\left.\frac{\partial \widetilde{c}_{\mathrm{i}}}{\partial y}\right|_{y=0} = 0 \quad \text{for} \quad y = 0 \tag{5–55}$$

for the all the species except for hydrogen peroxide and oxygen. The boundary condition for the reacting species (hydrogen peroxide) was

$$\widetilde{c}_{\mathrm{H2O2}}(0) = 1 \quad \text{for} \quad y = 0 \tag{5–56}$$

The value of $\widetilde{c}_{\mathrm{H2O2}}(0)$ was chosen arbitrarily because the governing equations for the impedance response are linear, even when the steady-state problem is non-linear. The boundary condition for $\widetilde{c}_{\mathrm{O2}}$, also technically a reacting species, cannot be the same as for hydrogen peroxide because the problem would be overspecified. Following reaction stoichiometry,

$$-\nabla \cdot N_{\mathrm{H2O2}}|_{\mathrm{J}=1} = \nabla \cdot N_{\mathrm{O2}}|_{\mathrm{J}=1} \tag{5–57}$$

as shown in equation (5–9). In a manner similar to that developed in section 4.1.5.2, a material balance at the quarter mode for each species was taken as

$$\left.\frac{dc_{\mathrm{H2O2}}}{dt}\right|_{\mathrm{J}+1/4} = -\nabla \cdot N_{\mathrm{H2O2}}|_{\mathrm{J}+1/4} + R_4|_{\mathrm{J}+1/4} \tag{5–58}$$

and

$$\left.\frac{dc_{\mathrm{O2}}}{dt}\right|_{\mathrm{J}+1/4} = -\nabla \cdot N_{\mathrm{O2}}|_{\mathrm{J}+1/4} - R_3|_{\mathrm{J}+1/4} \tag{5–59}$$

re-writing the material balance equation in a difference of fluxes gave

$$\left.\frac{dc_{\mathrm{H2O2}}}{dt}\right|_{\mathrm{J}+1/4} = -\left[\frac{N_{\mathrm{H2O2}}|_{\mathrm{J}+1/2} - N_{\mathrm{H2O2}}|_{\mathrm{J}}}{\mathrm{HHH}/2}\right] + R_4|_{\mathrm{J}+1/4} \tag{5–60}$$

and

$$\left.\frac{dc_{\mathrm{O2}}}{dt}\right|_{\mathrm{J}+1/4} = -\left[\frac{N_{\mathrm{O2}}|_{\mathrm{J}+1/2} - N_{\mathrm{O2}}|_{\mathrm{J}}}{\mathrm{HHH}/2}\right] - R_3|_{\mathrm{J}+1/4} \tag{5–61}$$

See section 5.1.3 for clarification on the mesh size, HHH. With the relationship between the flux of hydrogen peroxide and oxygen at the electrode surface, equation (5–57), and rearrangement of equations (5–60) and (5–60), a final expression for the oscillating boundary condition of O2 is

$$\begin{aligned} 0 = -\frac{\mathrm{HHH}}{2}\left.\frac{dc_{\mathrm{H2O2}}}{dt}\right|_{\mathrm{J}+1/4} - \frac{\mathrm{HHH}}{2}\left.\frac{dc_{\mathrm{O2}}}{dt}\right|_{\mathrm{J}+1/4} + N_{\mathrm{H2O2}}|_{\mathrm{J}+1/2} + N_{\mathrm{O2}}|_{\mathrm{J}+1/2} \\ + \frac{\mathrm{HHH}}{2}R_4|_{\mathrm{J}+1/4} - \frac{\mathrm{HHH}}{2}R_3|_{\mathrm{J}+1/4} \end{aligned} \tag{5–62}$$

5.1.2.1 Diffusion impedance for CGM system

The oscillating current associated with B^+, was expressed as

$$\widetilde{i}_{\mathrm{H2O2}} = \left(\frac{\partial i_{\mathrm{H2O2}}}{\partial V}\right)_{c_{\mathrm{H2O2}}(0)} \widetilde{V} + \left(\frac{\partial i_{\mathrm{H2O2}}}{\partial c_{\mathrm{H2O2}}(0)}\right)_{V} \widetilde{c}_{\mathrm{H2O2}}(0) \tag{5–63}$$

The flux expression for H2O2 yields a second equation for the oscillating current density as

$$\widetilde{i}_{\mathrm{H2O2}} = \mathrm{F} D_{\mathrm{H2O2}} \left.\frac{\mathrm{d}\widetilde{c}_{\mathrm{H2O2}}}{\mathrm{d}y}\right|_{y=0} \tag{5–64}$$

Equation (5–63) was divided by equation (5–64), yielding

$$1 = \left(\frac{\partial i_{\mathrm{H2O2}}}{\partial V}\right)_{c_{\mathrm{H2O2}}(0)} \frac{\widetilde{V}}{\widetilde{i}_{\mathrm{H2O2}}} + \left(\frac{\partial i_{\mathrm{H2O2}}}{\partial c_{\mathrm{H2O2}}(0)}\right)_{V} \frac{\widetilde{c}_{\mathrm{H2O2}}(0)}{\mathrm{F} D_{\mathrm{H2O2}} \left.\frac{\mathrm{d}\widetilde{c}_{\mathrm{H2O2}}}{\mathrm{d}y}\right|_{y=0}} \tag{5–65}$$

Thus, the impedance was expressed as

$$Z_{\mathrm{F,H2O2}} = R_{\mathrm{t,H2O2}} + Z_{\mathrm{D,H2O2}} \tag{5–66}$$

where, from the respective derivatives of equation (5–10),

$$R_{\mathrm{t,H2O2}} = \frac{1}{K_{\mathrm{H2O2}} b_{\mathrm{H2O2}} \bar{c}_{\mathrm{H2O2}}(0) \exp\left(-b_{\mathrm{H2O2}} \overline{V}\right)} \tag{5–67}$$

and

$$Z_{\mathrm{D,H2O2}} = \frac{R_{\mathrm{t,H2O2}} K_{\mathrm{H2O2}} \exp\left(-b_{\mathrm{H2O2}} \overline{V}\right)}{\mathrm{F} D_{\mathrm{H2O2}}} \left(- \frac{\widetilde{c}_{\mathrm{H2O2}}(0)}{\left.\frac{\mathrm{d}\widetilde{c}_{\mathrm{H2O2}}}{\mathrm{d}y}\right|_{y=0}} \right) \tag{5–68}$$

The concentration distributions required to assess the diffusion impedance, equation (5–68), were obtained for each frequency from the numerical solution of equations (5–42) - (5–53). The dimensionless diffusion impedance is

$$\frac{1}{\theta'_{\mathrm{H2O2}}} = \frac{1}{\delta_{\mathrm{GOx}}} \left(- \frac{\widetilde{c}_{\mathrm{H2O2}}(0)}{\left.\frac{\mathrm{d}\widetilde{c}_{\mathrm{H2O2}}}{\mathrm{d}y}\right|_{y=0}} \right) \tag{5–69}$$

where δ_{GOx} is the thickness of the glucose oxidase layer.

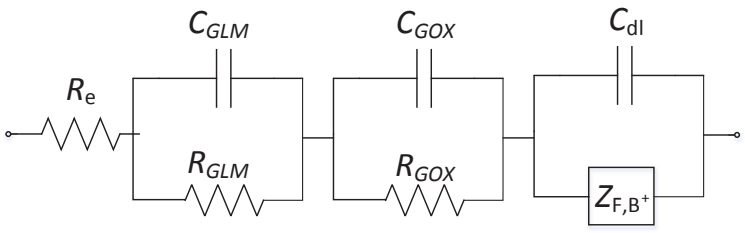


Figure 5-4. Circuit diagram of how the overall impedance will be modeled with the simulated faradaic impedance, equation (4–31).

5.1.2.2 Overall impedance for CGM system

The overall impedance was expressed by a circuit model shown in Figure 5-4. An ohmic resistance is in series with RC circuits for the GOX and GLM layers and the parallel contribution of the faradaic impedance, equation (5–68), and double-layer capacitance. To approximate the influence of the GOX and GLM layers the capacitance for each layer is expressed as

$$C_{\mathrm{GOX}} = \frac{\varepsilon_{\mathrm{GOX}}\varepsilon_o}{\delta_{\mathrm{GOX}}} \tag{5–70}$$

and

$$C_{\mathrm{GLM}} = \frac{\varepsilon_{\mathrm{GLM}}\varepsilon_o}{\delta_{\mathrm{GLM}}} \tag{5–71}$$

where ε is the dielectric constant of the medium and is approximated as 20 for the GOX and GLM. The distance for the GOX layer and the GLM layer, δ_{GOX} and δ_{GLM}, are 7 μm and 15 μm respectively. The permittivity of a vacuum, ε_o, is 8.8542×10^{-14} F/cm. The capacitance for the GOX and GLM are 0.002 and 0.001 μF/cm^2, respectively. The resistance for each layer is represented by

$$R_{\mathrm{GOX}} = \delta_{\mathrm{GOX}}\rho_{\mathrm{GOX}} \tag{5–72}$$

and

$$R_{\mathrm{GLM}} = \delta_{\mathrm{GLM}}\rho_{\mathrm{GLM}} \tag{5–73}$$

where ρ is the resistivity which is the inverse of conductivity. Conductivity was approximated as being the same as a phosphate buffered saline (PBS) solution, 0.01 Ω^{-1}cm^{-1}.

The resistance for each layer is 0.058 Ωcm^2 for GOX and 0.125 Ωcm^2 for GLM. The characteristic frequency is approximated by

$$f_c = \frac{1}{2\pi R_{\text{layer}} C_{\text{layer}}} \tag{5–74}$$

The characteristic frequency for both systems, after inputting the values of capacitance and resistance calculated above, is estimated as greater then 1 GHz, which is outside the typical frequency range for impedance spectroscopy. This means Figure 5-4 can be represented mathematically by equation (5–75). The faradaic impedance, $Z_{\text{F,H2O2}}$ comes from equation (5–66).

$$Z = R_{\text{e}} + R_{\text{GOx}} + R_{\text{GLM}} + \frac{Z_{\text{F,H2O2}}}{1 + \text{j}\omega Z_{\text{F,B}^+} C_{\text{dl}}} \tag{5–75}$$

5.1.3 Numerical Methods for CGM

The coupled non-linear differential equations were solved following the numerical approach described in Section 4.1.5. BAND is described in more detail in Appendix A. The second derivatives were discretized by (A–2) and first derivatives by (A–3).

For input values shown in Tables 5-1 and 5-2 the steady-state concentrations of the non enzymatic and enzymatic concentrations as well as the value of the homogeneous reactions were obtained. These steady-state values along with the original input values were used to obtain the oscillating values on concentration of the reacting species H2O2. For a spectrum of dimensionless frequencies, from $K = 1E-5$ to $1E5$ with 20 points per decade, the oscillating concentrations of the reacting species obtained near the electrode were imputed into equation (5–69) to obtain a dimensionless diffusion impedance and equation (5–68) to obtain a diffusion impedance.

To obtain a solution that is the most accurate, the finite difference errors and round-off errors need to be minimized. Near the electrode surface the flux of the reacting species is changing so dramatically it is necessary to have a smaller mesh to obtain an accurate answer. Far from the electrode the concentrations are not changing as much and

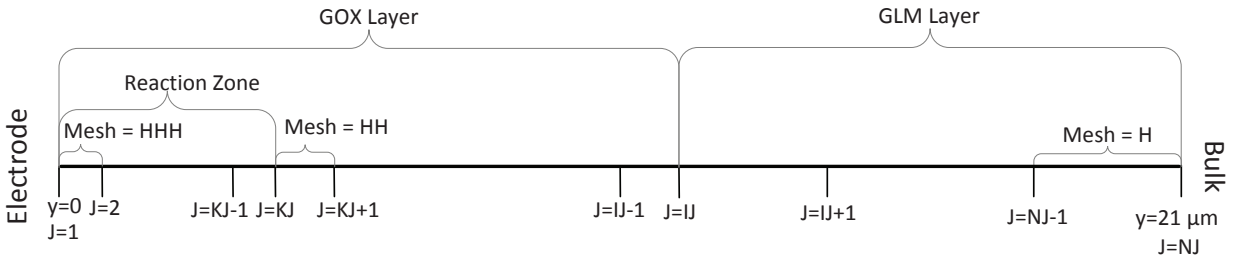


Figure 5-5. One dimensional schematic showing three dissimilar mesh sizes. HHH is the smallest mesh size and HH is the medium mesh size, and both are in the GOX layer. J=KJ is the interface in the GOX layer. H is the largest mesh size and is used throughout the GLM layer. J=IJ is the interface between the GOX and GLM layers.

therefore the same small mesh would result in unnecessary roundoff errors. Even further from the electrode, in the GLM region, the only influence on concentration is diffusion, so an even larger mesh size is necessary in this region. To mitigate errors we have solved the system using three mesh sizes. A small mesh size, HHH, is used near the electrode and a large mesh size, HH, is used far away from the electrode but still in the GOX region. The largest mesh size, H, is used for in the GLM region where no reactions are occurring.

With the use of multiple dissimilar mesh sizes, it is necessary that a method of coupling be developed to allow the computation transition from one region to another while retaining the accuracy of the finite difference calculation. Such transition is performed by a coupling subroutine at an interface designated at j = KJ in the GOX region and j = IJ at the GOX-GLM interface. A visual representation of the three mesh size regions is shown in Figure 5-5.

5.1.3.1 Couplers in the CGM

The coupler in the GOX region of the code is coupling two different mesh sizes with the same governing equations. The flux on one side of the coupler at j = KJ, is equal to the flux on the other side of the coupler. The divergence of flux of each side is considered.

$$\nabla \cdot N_{\mathrm{i}}|_{\mathrm{KJ}-1/4} = \frac{N_{\mathrm{i}}|_{\mathrm{KJ}-1/2} - N_{\mathrm{i}}|_{\mathrm{KJ}}}{2 * \mathrm{HHH}} + R_{\mathrm{i}}|_{\mathrm{KJ}-1/4} = 0 \tag{5–76}$$

Table 5-2. System parameter values for the continuous glucose monitor simulations

Parameter	Value	Units
Distance of the Reaction Layer, Y1	0.0004	cm
Distance of the Inner Layer, Y2	0.0003	cm
Distance of the GLM Layer, Y3	0.0015	cm
Porosity factor GOX, σ_{GOX}	0.8	dimensionless
Porosity factor GLM small molecules, $\sigma_{\mathrm{GLM}} - \mathrm{small}$	0.42	dimensionless
Porosity factor GLM large molecules, $\sigma_{\mathrm{GLM}} - \mathrm{large}$	0.169	dimensionless
Solubility coefficient H2O2, α_{H2O2}	0.32	dimensionless
Solubility coefficient O2, α_{O2}	0.11	dimensionless
Solubility coefficient Glucose, α_{G}	0.0176	dimensionless
Heterogeneous Rate Constant, K_{H2O2}	1	$\mathrm{A/cm^2}$
Heterogeneous Constant, b_{H2O2}	37.42	$\mathrm{V^{-1}}$

and

$$\nabla \cdot N_{\mathrm{i}}|_{\mathrm{KJ+1/4}} = \frac{N_{\mathrm{i}}|_{\mathrm{KJ}} - N_{\mathrm{i}}|_{\mathrm{KJ+1/2}}}{2 * \mathrm{HH}} + R_{\mathrm{i}}|_{\mathrm{KJ+1/4}} = 0 \tag{5–77}$$

Re-writing equations (5–76) and (5–77) for $N_{\mathrm{i}}|_{\mathrm{KJ}}$ and setting equal we obtain

$$D_{\mathrm{i}}\frac{c_{\mathrm{i}}(\mathrm{KJ}+1) - c_{\mathrm{i}}(\mathrm{KJ})}{\mathrm{HH}} - D_{\mathrm{i}}\frac{c_{\mathrm{i}}(\mathrm{KJ}) - c_{\mathrm{i}}(\mathrm{KJ}-1)}{\mathrm{HHH}} + R_{\mathrm{i}}|_{\mathrm{KJ-1/4}} - R_{\mathrm{i}}|_{\mathrm{KJ+1/4}} = 0 \tag{5–78}$$

to represent the j = KJ coupler node.

The second coupler, representing j = IJ, is very similar to the first one. Because there are no enzymes in the GLM, there is no reaction term in the GLM. The equation representing the coupler between the GOX region and GLM region was

$$D_{\mathrm{i}}\frac{c_{\mathrm{i}}(\mathrm{IJ}+1) - c_{\mathrm{i}}(\mathrm{IJ})}{\mathrm{H}} - D_{\mathrm{i}}\frac{c_{\mathrm{i}}(\mathrm{IJ}) - c_{\mathrm{i}}(\mathrm{IJ}-1)}{\mathrm{HH}} + R_{\mathrm{i}}|_{\mathrm{IJ-1/4}} = 0 \tag{5–79}$$

5.2 CGM Results and Discussion

The steady-state and impedance results from the mathematical model for a continuous glucose monitor are presented. Some parameters were kept constant for all simulations and these parameter are presented in Table 5-2. The variable names, bulk value concentrations and diffusion coefficients were presented earlier in the chapter in Table 5-1. The first set of results are for systems where the changing parameters are the homogeneous (or enzymatic) reaction rates. The different systems looked at, titled System

Table 5-3. Kinetic parameter values for system 1

Parameter	Value	Units
Homogeneous Rate Constant 1, k_{f1},	10^6	$cm^3/mol/s$
Homogeneous Equilibrium Constant 1, K_{eq1}	10^{-2}	mol/cm^3
Homogeneous Rate Constant 2, k_{f2},	10^2	$cm^3/mol/s$
Homogeneous Rate Constant 3, k_{f3},	10^6	$cm^3/mol/s$
Homogeneous Equilibrium Constant 3, K_{eq3}	10^{-2}	mol/cm^3
Homogeneous Rate Constant 3, k_{f4},	10^2	$cm^3/mol/s$

Table 5-4. Kinetic parameter values for system 2

Parameter	Value	Units
Homogeneous Rate Constant 1, k_{f1},	10^5	$cm^3/mol/s$
Homogeneous Equilibrium Constant 1, K_{eq1}	10^2	mol/cm^3
Homogeneous Rate Constant 2, k_{f2},	10^3	$cm^3/mol/s$
Homogeneous Rate Constant 3, k_{f3},	10^5	$cm^3/mol/s$
Homogeneous Equilibrium Constant 3, K_{eq3}	10^2	mol/cm^3
Homogeneous Rate Constant 3, k_{f4},	10^3	$cm^3/mol/s$

1, System 2 and System 3 all have varying kinetic parameters, presented in Tables 5-3-5-5. The second set of results show the affects of bulk oxygen concentration on the steady-state and impedance results. The oxygen concentration varied from $5 \times 10^{-9} mol/cm^3$, the value given in Table 5-1, to an order of magnitude larger, $5 \times 10^{-8} mol/cm^3$, and an order of magnitude smaller, $5 \times 10^{-10} mol/cm^3$.

5.2.1 Homogeneous Reaction Rate Influence on the CGM

The polarization curve for a system described in Table 5-2 using different homogeneous reaction rates is shown in Figure 5-6. The three curves correspond to different homogeneous reaction rates and are in Tables 5-3, 5-4, and 5-5. The inset graph is a zoomed in view to show the behavior of System 1. As the overall homogeneous reaction increases, from

Table 5-5. Kinetic parameter values for system 3

Parameter	Value	Units
Homogeneous Rate Constant 1, k_{f1},	10^8	$cm^3/mol/s$
Homogeneous Equilibrium Constant 1, K_{eq1}	1	mol/cm^3
Homogeneous Rate Constant 2, k_{f2},	10^7	$cm^3/mol/s$
Homogeneous Rate Constant 3, k_{f3},	10^8	$cm^3/mol/s$
Homogeneous Equilibrium Constant 3, K_{eq3}	1	mol/cm^3
Homogeneous Rate Constant 3, k_{f4},	10^7	$cm^3/mol/s$

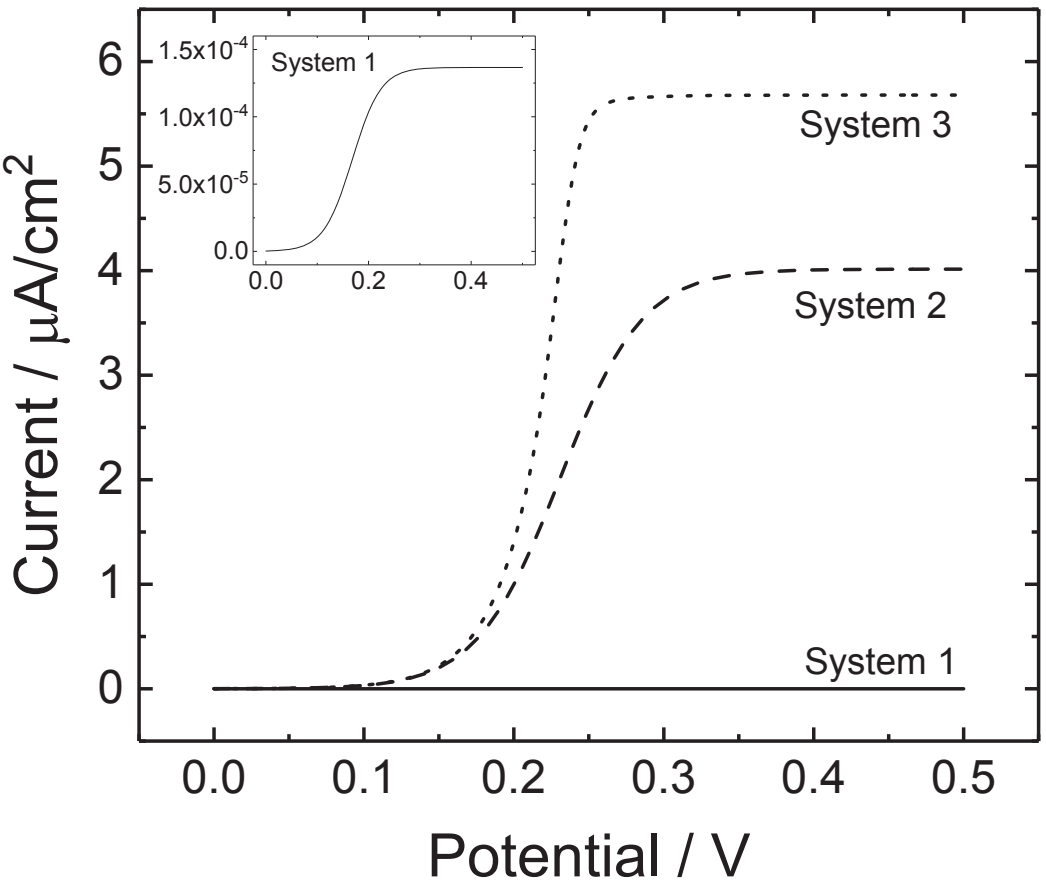


Figure 5-6. Polarization curve calculated from data in Table 5-2. The different homogenous reaction rates are in Tables 5-3, 5-4, and 5-5.

System 1 to System 3, the current response is also greater. The different hydrogel layers on top of the electrode complicate the system. Species do not have the same mobility in a hydrogel as they would in a more traditional electrolyte. The porosity factors in Table 5-2 account for a general porosity factor in the GOX layer and two different porosity factors in the GLM layer, one for the small molecules, oxygen and hydrogen peroxide and one for large molecules, glucose and gluconic acid. The large molecules have a much more hindered mobility through the GLM them small molecules and that is reflected in the much smaller porosity factor. The porosity effects the system anywhere a diffusion coefficient is called. The diffusion coefficient is multiplied by the appropriate porosity factor to the 1.5 power. Because the molecules will be flowing in the bulk in an aqueous electrolyte and then into a hydrogel, a partition coefficient (or

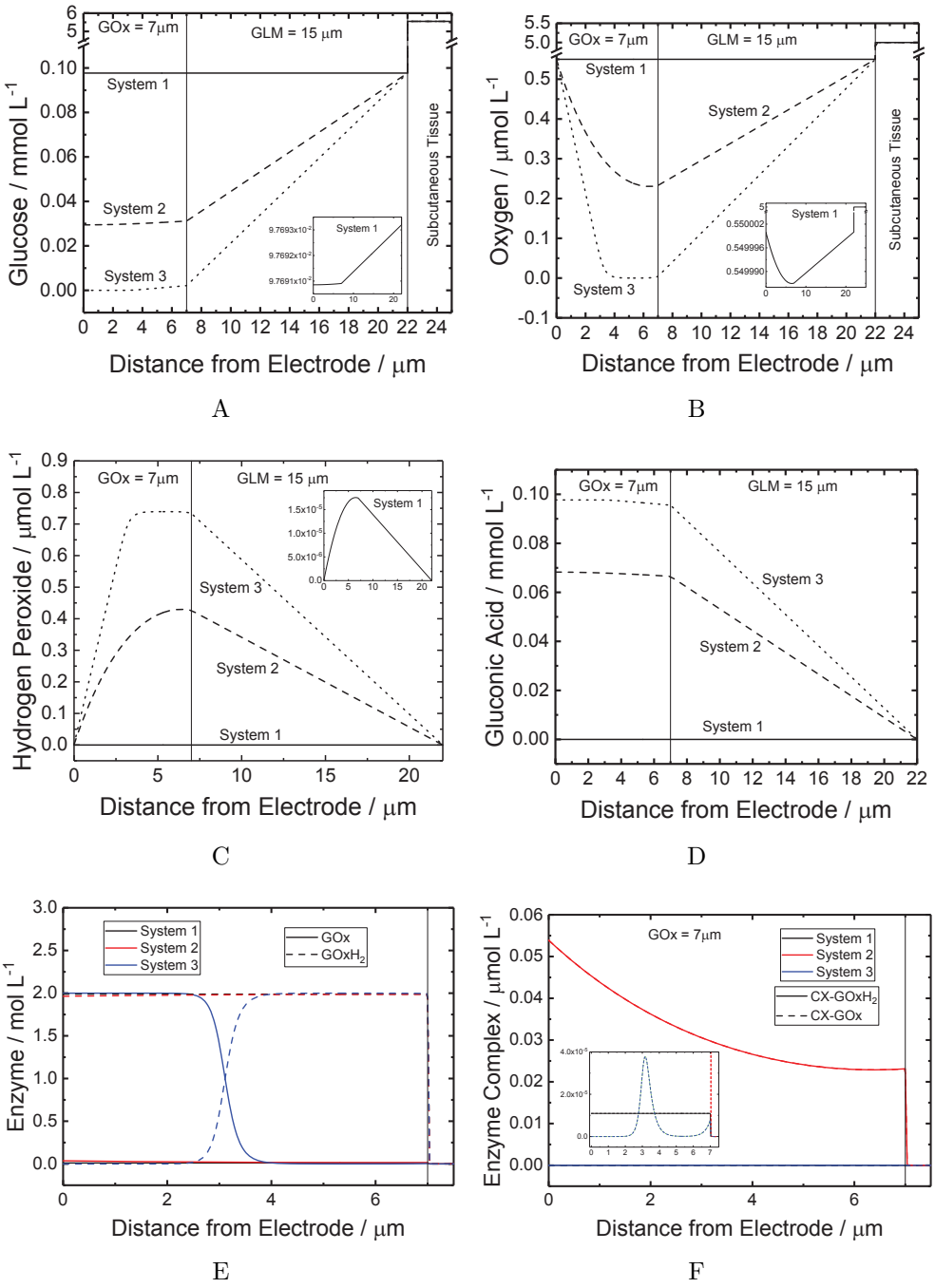


Figure 5-7. Calculated steady-state concentration distributions corresponding to system parameters presented in Tables 5-1 and 5-2; a) Glucose b) Oxygen c) Hydrogen Peroxide d) Gluconic Acid e) Glucose Oxidase Enzyme in reduced and oxidized forms f) Glucose Oxidase Enzyme complex formed in both enzymatic reactions with the different systems being described by different homogeneous reactions presented in Tables 5-3, 5-4, and 5-5.

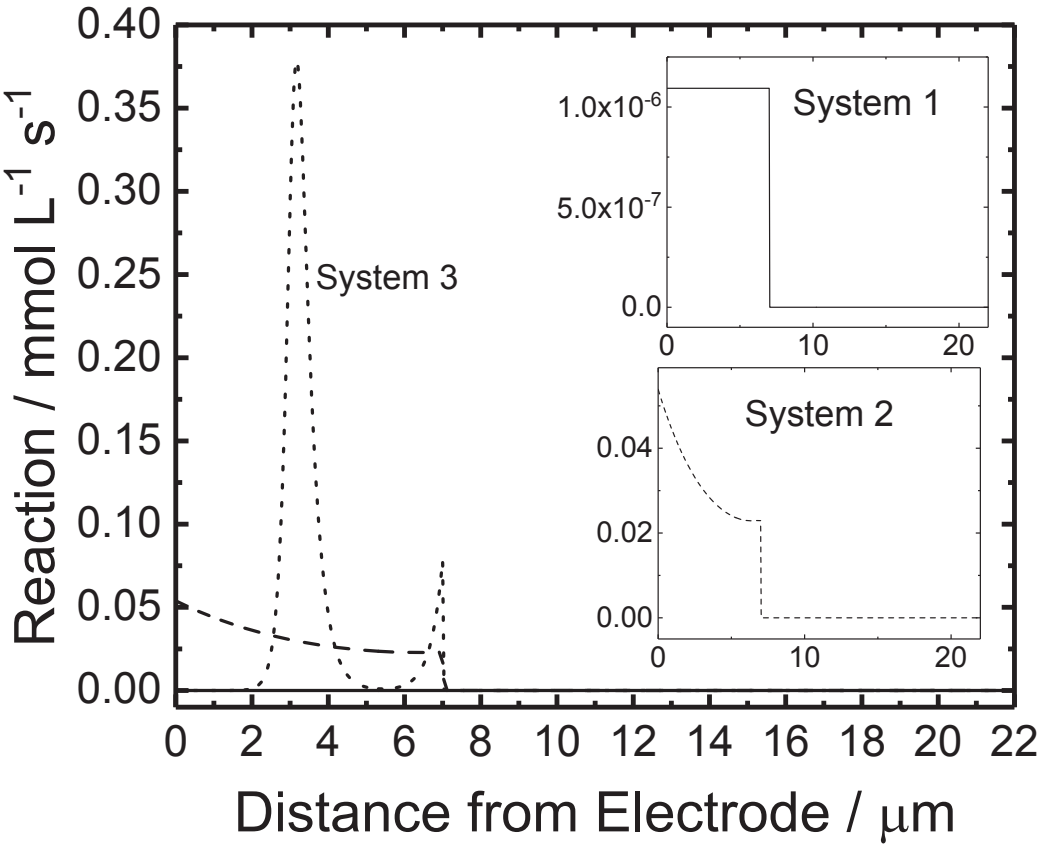


Figure 5-8. Reaction profile calculated for different homogeneous reaction rates from data in Table 5-2 and corresponding with the steady-state concentrations in Figure 5-7. The different homogenous reaction rates are in Tables 5-3, 5-4, and 5-5.

solubility factor) is multiplied by the bulk value to show the change of concentration. The steady-state results corresponding to the polarization curve, Figure 5-6 are displayed in Figure 5-7. The glucose concentration, Figure 5-7A, decreases dramatically as soon as it diffuses into the glucose limiting membrane due to the solubility coefficient and reacts in the GOX layer. With a large enough homogeneous reaction rate, System 3, the glucose concentration is completely consumed as it gets closer to the electrode surface. The oxygen concentration, Figure 5-7B, is also effected by a solubility coefficient so the concentration in the subcutaneous fluid is significantly more then is right inside the GLM. Similar to the glucose concentration for the larger homogeneous reaction rate, System 3, the oxygen concentration is complectly consumed for this larger homogeneous reaction

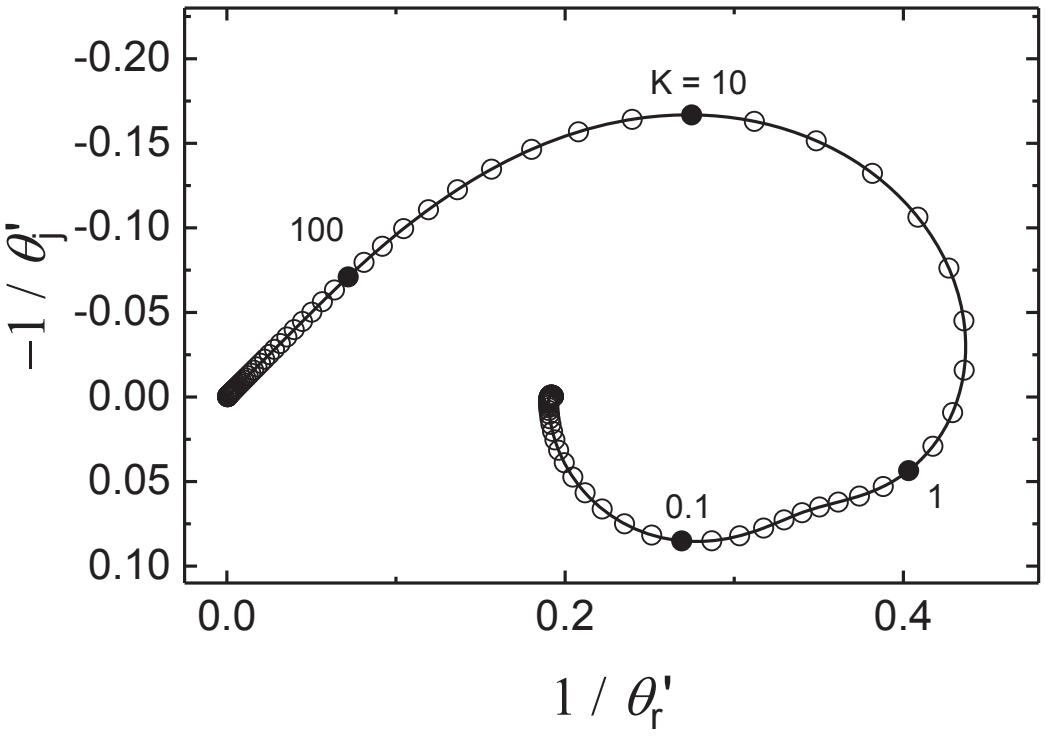


Figure 5-9. Calculated dimensionless diffusion impedances for the CGM with different homogeneous reaction rates corresponding to system parameters presented in Tables 5-1 and 5-2 for System 1 described in Table 5-3.

rate. The oxygen and hydrogen peroxide are allowed to flow through the GLM easier then the larger molecules.

Oxygen is also created in the electrochemical reaction at the electrode surface which is shown for all three systems. The hydrogen peroxide, Figure 5-7C, is produced in the GOX layer and reacts on the electrode surface, however, some of the hydrogen peroxide diffuses into the GLM. Gluconic acid, Figure 5-7D, is also formed in the enzymatic reaction but does not react in the electrochemical reaction and therefore has a much larger concentration, and diffuses into the body through the GLM. The concentration profile of the glucose oxidase enzyme in the oxidized form (solid lines) and the reduced form (dashed lines) are presented in Figure 5-7E.

The enzyme complex concentration profiles are presented in Figure 5-7F. The overall concentration of the enzyme in the reduced and oxidized form as well as the concentration

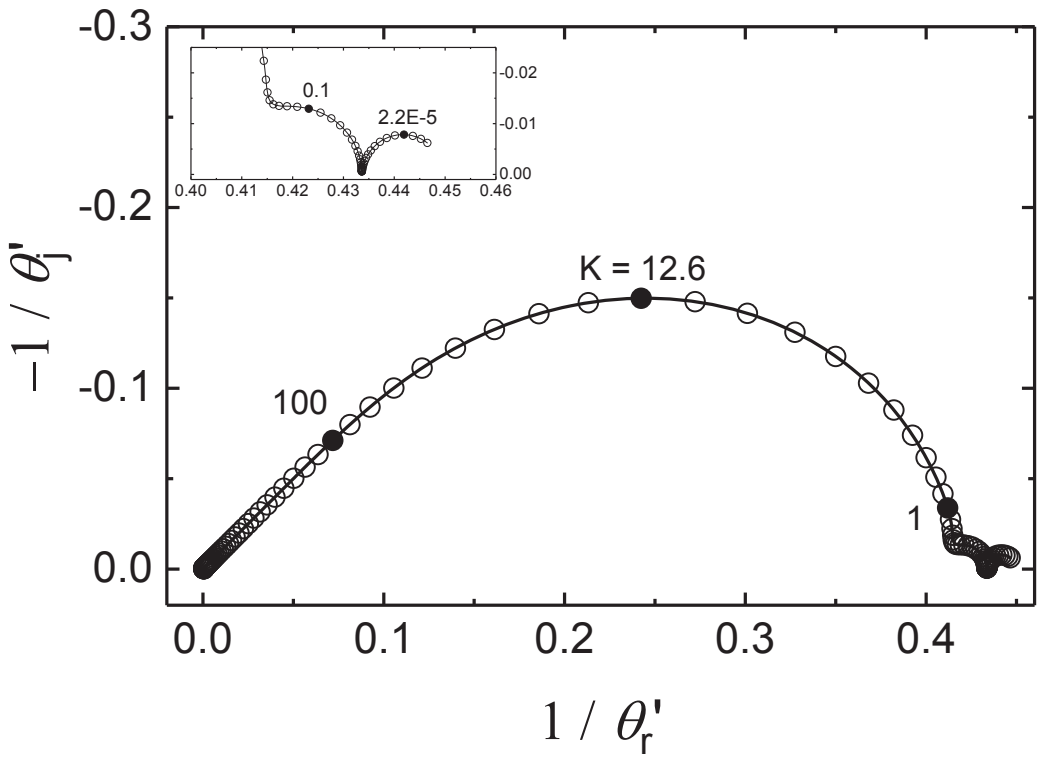


Figure 5-10. Calculated dimensionless diffusion impedances for the CGM with different homogeneous reaction rates corresponding to system parameters presented in Tables 5-1 and 5-2 for System 2 described in Table 5-4.

of the enzyme complexes at any distance from the electrode is a constant. The enzyme complex concentration profiles follow the same shape as the reaction profile, Figure 5-8. The reaction profiles show different behaviors. For System 3, the reaction spikes in the center of the GOX layer. The reaction profile for System 2 is largest near the electrode surface. System 1 shows a smaller reaction value then the other two systems, and the profile is flat throughout the GOX region. All the systems show no reaction occurring in the GLM. There is also no concentration of the enzyme in either of its forms or either enzyme complex in the GLM.

The diffusion impedance for Systems 1-3 are presented in Figures 5-9-5-11.

The diffusion impedance is not dramatically different for any of the systems. System 1, presented in Figure 5-9, is the only diffusion impedance with an inductive feature. The first loop resembles a Gerischer impedance and has a characteristic frequency of 10. The

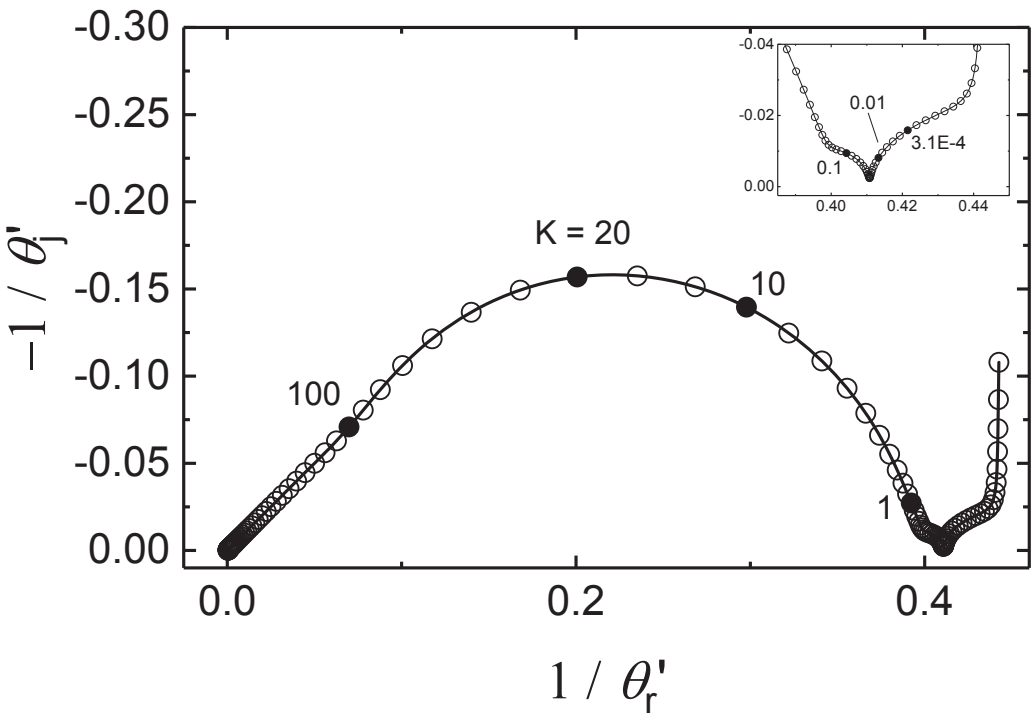


Figure 5-11. Calculated dimensionless diffusion impedances for the CGM with different homogeneous reaction rates corresponding to system parameters presented in Tables 5-1 and 5-2 for System 3 described in Table 5-5.

inductive loop has a characteristic frequency of 0.1. For system 2, shown in Figure 5-10, the impedance looks like a Gerischer impedance at high frequency and at low frequency has a tail with two more time constants. The first time constant is at a dimensionless frequency of 12.6. The second time constant could be attributed to convective-diffusion impedance. The third time constant occurs at very low frequency, a dimensionless frequency of 2.2×10^{-5}. The diffusion impedance for System 3 is presented in Figure 5-11 which strongly resembles the shape of System 2. The first loop looks like a traditional convective-diffusion impedance and characteristic frequency is 20. The low frequency loops have a characteristic frequency of 0.1 and 3.1×10^{-4}.

5.2.2 Influence of Oxygen Concentration on the CGM

Other properties besides the homogeneous reaction rate can be explored. The effect of oxygen concentration in the bulk (subcutaneous fluid) was studied for the

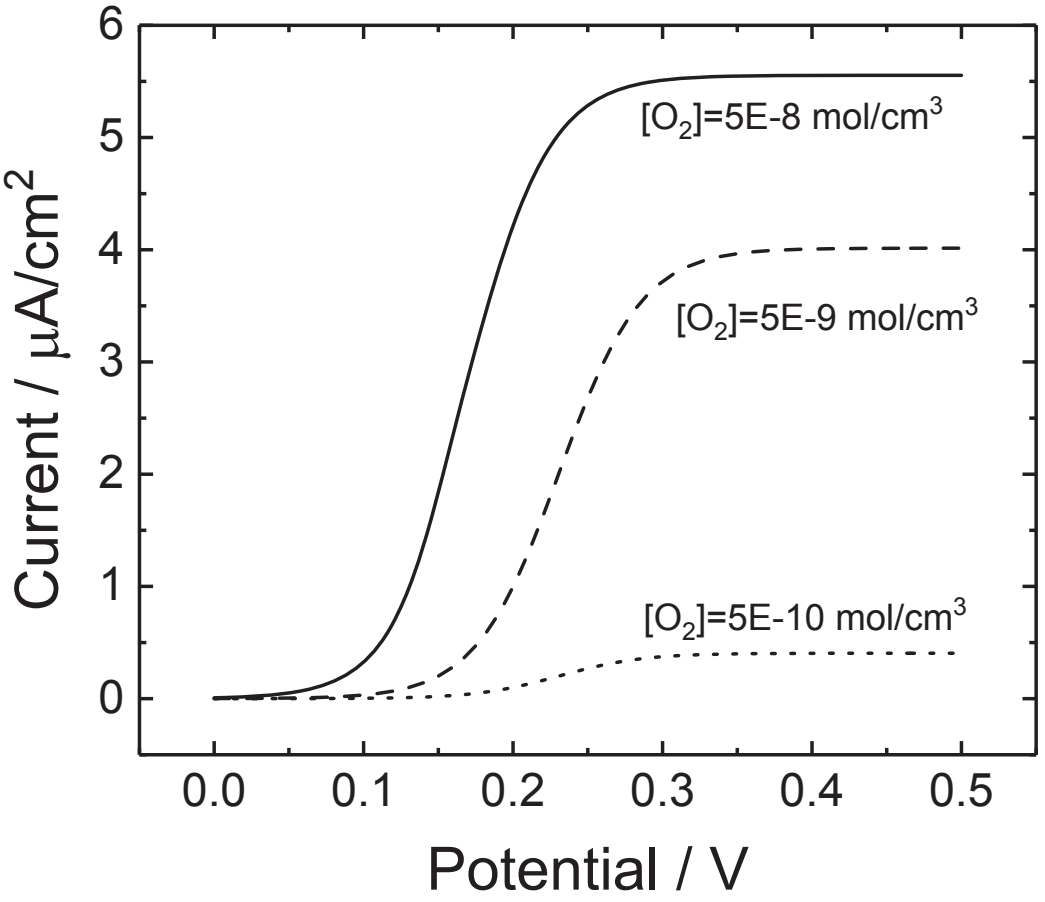


Figure 5-12. Polarization curve calculated from data in Table 5-2 and corresponding with the steady-state concentrations in Figure 5-13. The homogenous reaction rates are in Table 5-4. The oxygen concentration varies from 5E-8, 5E-9 and 5E-10 mol/cm^3.

homogeneous kinetic properties of System 2, Table 5-4. The oxygen concentration in the bulk varied from a standard value of 5×10^{-9}mol/cm^3, used for the homogeneous reaction rate study, to a value an order of magnitude larger and smaller, 5×10^{-8}mol/cm^3 and 5×10^{-10}mol/cm^3, respectively. The polarization curve, see Figure 5-12, shows that more oxygen in the bulk corresponds to a higher current and less to a smaller current. The middle curve, corresponding to the standard value of 5×10^{-9}mol/cm^3 and System 2 kinetic parameters, Table 5-4, is the same data plotted in the above section, section 5.2.1. The steady-state results corresponding to the polarization curve, Figure 5-12 are displayed in Figure 5-13. The glucose concentration profiles, Figure 5-13A show that having a

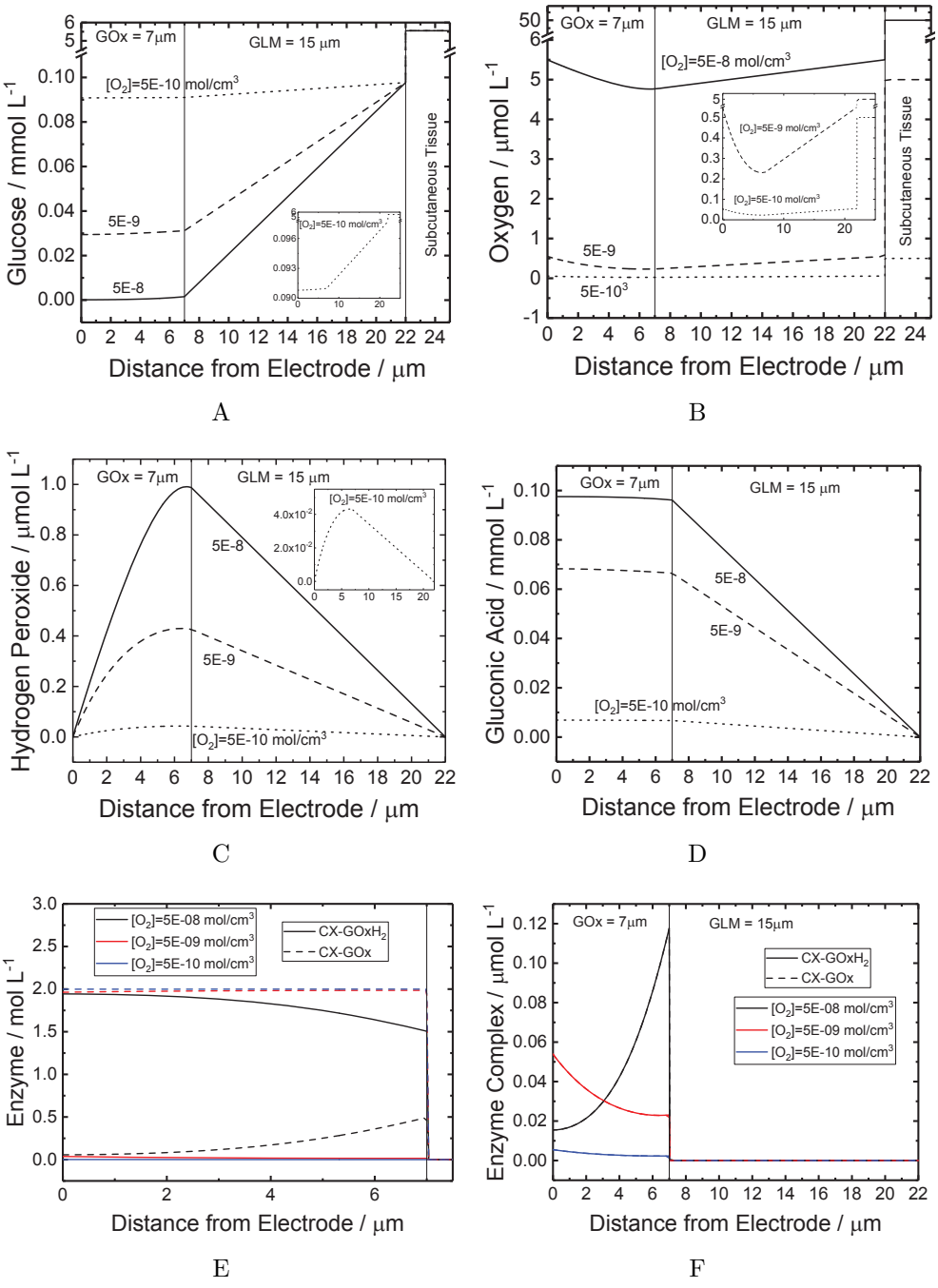


Figure 5-13. Calculated steady-state concentration distributions corresponding to system parameters presented in Tables 5-1, 5-2 and ; a) Glucose b) Oxygen c) Hydrogen Peroxide d) Gluconic Acid e) Glucose Oxidase Enzyme in reduced and oxidized forms f) Glucose Oxidase Enzyme complex formed in both enzymatic reactions with the different concentrations of oxygen

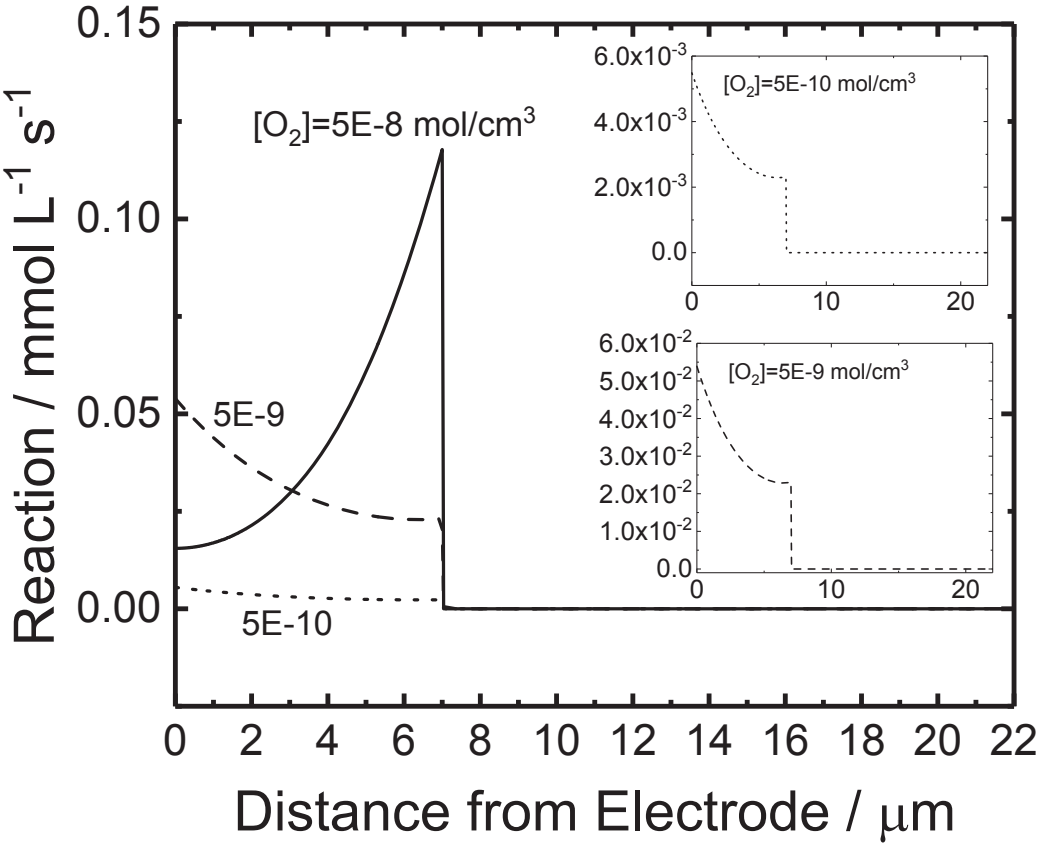


Figure 5-14. Reaction profile calculated from data in Table 5-2 and corresponding with the steady-state concentrations in Figure 5-13. The homogenous reaction rates are in Tables 5-4.

higher concentration of oxygen allows more glucose to be consumed in the glucose oxidase layer. The oxygen concentration profiles, Figure 5-13B, have the same shape regardless of the bulk oxygen concentration. The glucose and oxygen, as previously shown in Figure 5-7, have a dramatic concentration difference inside the CGM and outside the CGM in the interstitial fluid due to the partition coefficients. The hydrogen peroxide profiles show that the hydrogen peroxide is consumed at the electrode and are also the same shape regardless of the oxygen concentration, see Figure 5-13C. Gluconic acid, Figure 5-13D, a byproduct in the enzymatic reaction, is produced more with a larger bulk oxygen concentration. The enzyme concentration profile, both the reduced and oxidized forms, are presented in Figure 5-13E and the enzyme complex concentration profiles are presented in Figure 5-13F. The enzyme complex concentration profile has the same shape as the reaction profile,

see Figure 5-14. When the oxygen concentration is smaller the reaction profile shows a spike at the electrode surface. As the oxygen concentration in the bulk is increased to 5×10^{-8}mol/cm^3, the spike in the reaction profile is at the GOX–GLM interface.

The dimensionless diffusion impedances for the different oxygen concentrations are shown in Figures 5-15-5-17. The diffusion impedance for the largest concentration of oxygen, Figure 5-15, shows one time constant at 0.56 and is a much larger impedance then the other bulk oxygen concentrations. For the impedances for both the oxygen concentration of 5×10^{-9}mol/cm^3 and 5×10^{-10}mol/cm^3, in Figures 5-16 and 5-17, the characteristic frequency is at 12.5. The data for all the impedances with different bulk oxygen concentrations are presented in Figure 5-18, and it is clear that the lower oxygen concentrations overlap in their diffusion impedance at high frequencies and are slightly different, but visually very similar, at low frequencies. This comparison also makes it more clear that the impedance for the high bulk concentration of oxygen is much bigger then the other two impedances.

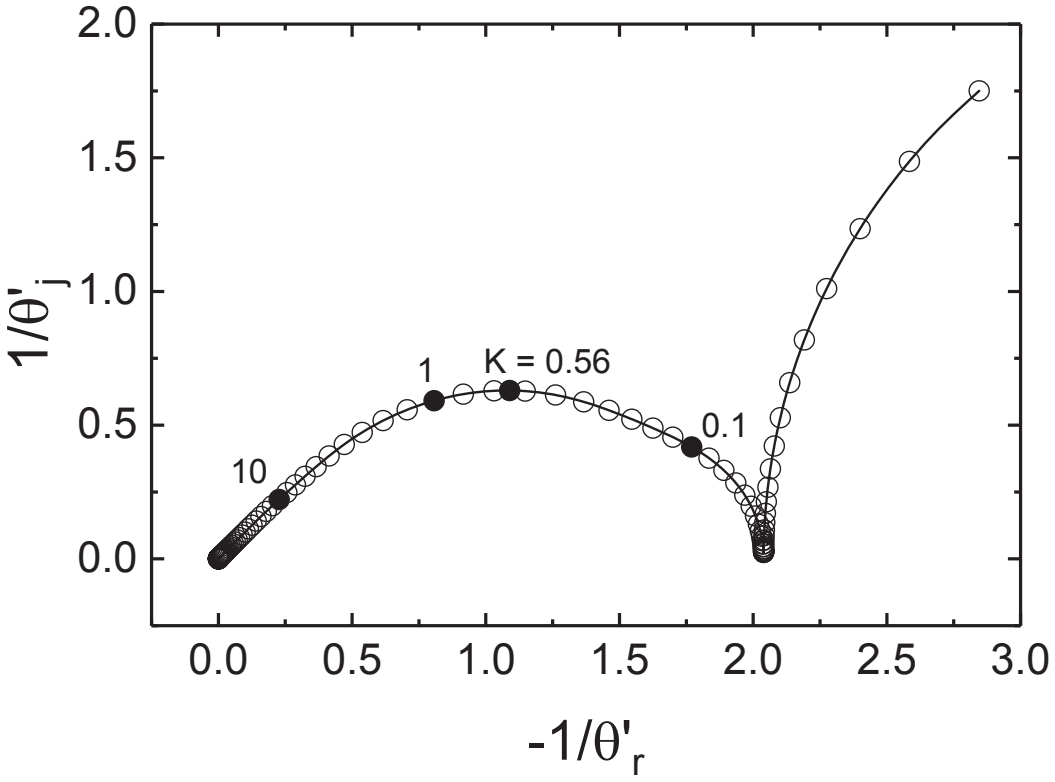


Figure 5-15. Calculated dimensionless diffusion impedance for the CGM with oxygen concentration of $5 \times 10^{-8} mol/cm^3$ corresponding to system parameters presented in Tables 5-1 and 5-2 and kinetic parameters shown in Table 5-4.

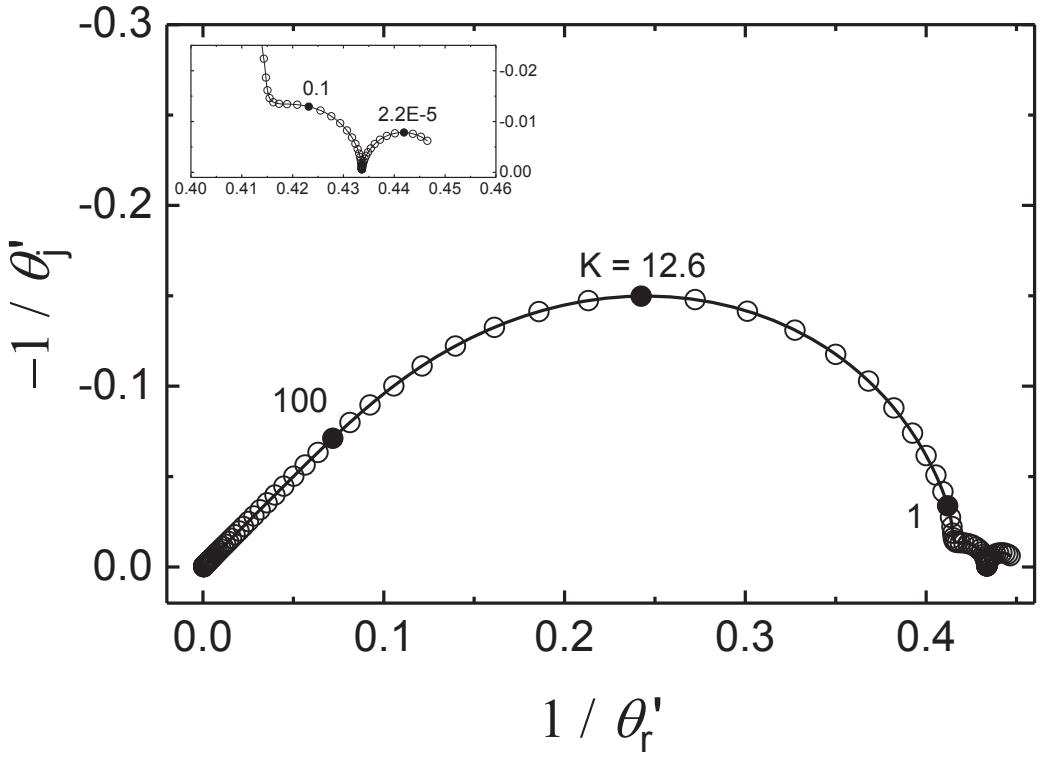


Figure 5-16. Calculated dimensionless diffusion impedance for the CGM with oxygen concentration of $5 \times 10^{-9} mol/cm^3$ corresponding to system parameters presented in Tables 5-1 and 5-2 and kinetic parameters shown in Table 5-4.

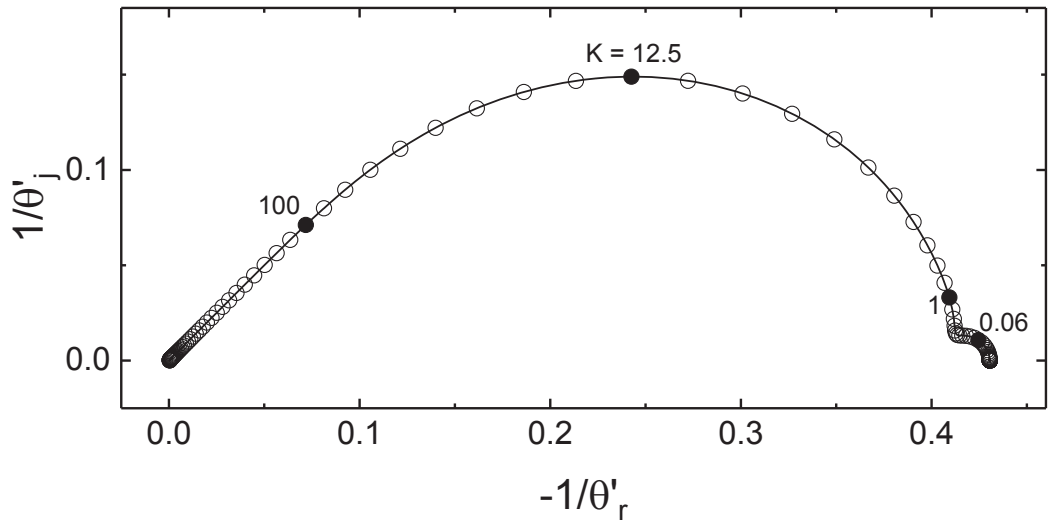


Figure 5-17. Calculated dimensionless diffusion impedance for the CGM with oxygen concentration of $5 \times 10^{-10} mol/cm^3$ corresponding to system parameters presented in Tables 5-1 and 5-2 and kinetic parameters shown in Table 5-4.

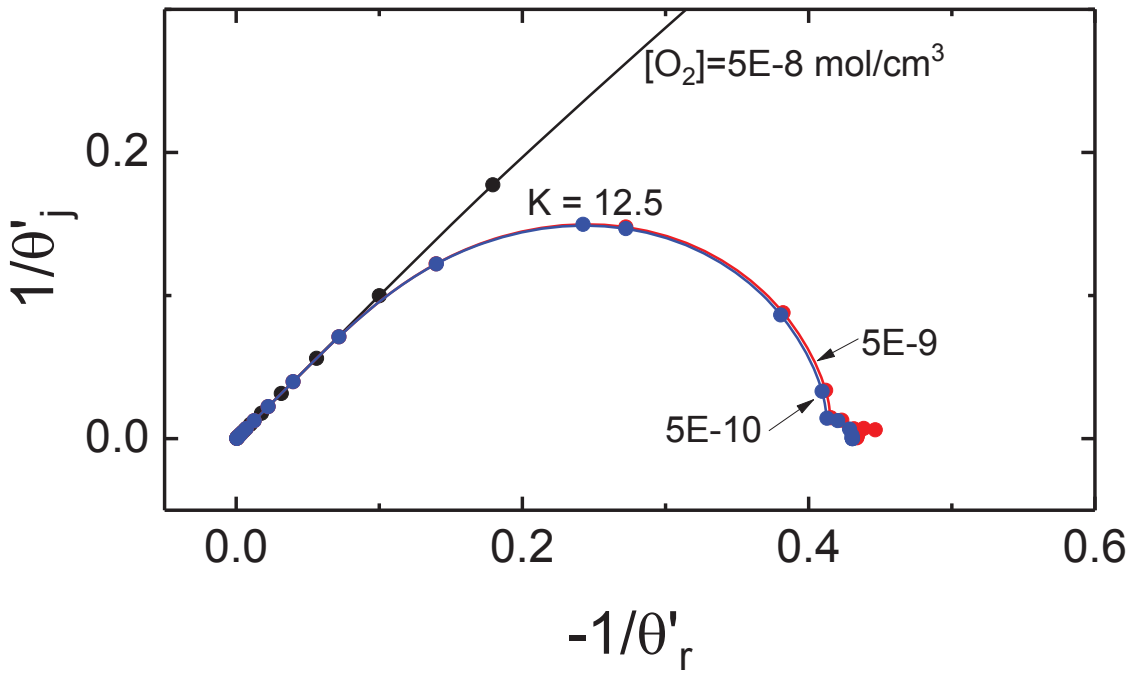


Figure 5-18. Calculated dimensionless diffusion impedance for the CGM with different oxygen concentrations and corresponding to system parameters presented in Tables 5-1 and 5-2 and kinetic parameters shown in Table 5-4.

CHAPTER 6
CONCLUSIONS

Mathematical models were developed for the impedance response for electrochemical systems. Diffusion or convective diffusion can influence the impedance response of a system associated with electrochemical reactions. The first models developed were for convective-diffusion for a rotating disk electrode and a submerged impinging jet electrode, in Chapter 3. The influence of a finite Schmidt number analysis impacted the dimensionless diffusion impedance at low frequency. The error of the impedance response using a finite Schmidt number analysis when compared to a infinite Schmidt number analysis was greater for a rotating disk electrode then for a submerged impinging jet electrode. The error for a system with a Schmidt number of 100 was up to 6.62% for a rotating disk electrode and up to 1.82% for a submerged impinging jet electrode. The convective-diffusion impedance simulations were performed in MATLAB, and the impedance is calculated using the oscillating concentration of the reacting species. The development of models for convective-diffusion impedance served as a foundation for the study of systems in which homogeneous reactions influence the impedance of electrochemical systems.

The second model explored a convective-diffusion rotating disk electrode where a homogenous reaction influences the system. The model included a homogeneous reaction in the electrolyte where species AB reacts reversibly to form A- and B+, and B+ reacts electrochemically on a rotating disk electrode to produce B. An analytic expression for velocity was employed that combined a three-term velocity expansion near the electrode surface to a three-term expansion that applied far from the electrode. The nonlinear expression for the homogeneous reaction was employed in which the concentrations of both A- and B+ were assumed to be dependent on position. This model provides an extension to the literature by using a nonlinear expression for the homogenous reaction and unique diffusion coefficients for each species. The resulting

convective-diffusion impedance had two asymmetric capacitive loop. The low frequency loop is associated with convective-diffusion impedance with a characteristic frequency of 2.5, in agreement with the impedance for a convective-diffusion system in the absence of a homogeneous reaction. The other loop, at a higher frequency, is associated with the homogeneous reaction. For an infinitely fast homogeneous reaction, the system is shown to behave as though AB is the electroactive species. Even though the assumption of a linear expression for the homogeneous reaction was relaxed, a modified Gerischer impedance was found to provide a good fit to the simulated data. The model was developed in FORTRAN, and a steady-state solution containing four variables was solved followed by a solution in the frequency-domain involving eight variables. The oscillating concentration of B+ was used to obtain the impedance spectrum. The development of this model chemical/electrochemical system guided development of models for the enzyme-based biosensors.

A continuous glucose monitor is a real world application of a homogeneous reaction influencing the electrochemical reaction. This system included a diffusion through a hydrogel like medium where glucose would react in an enzymatic reaction to form hydrogen peroxide, which can be detected electrochemically. The model accounts for a glucose limiting membrane GLM, which controls the amount of glucose participating in the enzymatic reaction, and a glucose oxidase enzyme layer. The glucose oxidase was assumed to be immobilized within a thin film adjacent to the electrode. In the glucose oxidase layer, a process of enzymatic catalysis transforms the glucose into hydrogen peroxide. The electrochemical reaction produced a current response that corresponds to the overall concentration of glucose in the subcutaneous fluid. The model development required two steps. The nonlinear coupled differential equations governing this system were solved under the assumption of a steady state. The steady-state concentrations resulting from the steady-state simulation were used in the solution of the linearized set of differential equations describing the sinusoidal steady state. The enzymatic catalysis

was treated in terms of four homogeneous reactions. A FORTRAN code was used to solve the steady-state equations for 12 variables which were used subsequently to solve the 24 frequency-domain equations. As before, the oscillating concentration of the electroactive species, hydrogen peroxide in this case, was used to obtain the impedance results.

CHAPTER 7
FUTURE WORK

7.1 Future Investigation of Continuous Glucose Monitor Code

A FORTRAN code was created to investigate the impedance response of a continuous glucose monitor. Initial simulations and results are discussed in Chapter 5. A more extensive parameter study of the kinetic parameters is still necessary to understand the system. Other parameters besides homogeneous kinetics will also need to be explored. The overall impedance needs to be examined as well.

7.1.1 CGM Parameter Study

An extensive parameter study needs to occur to fully understand the capabilities of the code as well as the influence of parameters on the system. An initial aggressive study of the kinetic parameters is in progress. The parameters being investigated are shown in Table 7-1, where Systems 1-3 are the systems described in section 5.2.1. The parameters were chosen arbitrarily due to the millions of combinations of kinetic parameters that could be tested. After analysis of the parameters listed more parameters can be chosen based on the knowledge gained from this initial study. The dimensionless diffusion impedance for all 10 systems noted in Table 7-1 are shown in Figure 7-1. From these diffusion impedances there are three distinct shapes. Systems 1 and 6 show an inductive loop at low frequency, presented in Figures 7-1A and 7-1F. Two capacitive loops are observed for Systems 2, 3 and 10, shown in Figures 7-1B, 7-1C, and 7-1J. The last distinct shape is a single capacitive loop, which was observed for Systems 4, 5, 7, 8, and 9, presented in Figures 7-1D, 7-1E, 7-1G, 7-1H, and 7-1I. These different shapes of diffusion impedance need to be examined further to understand their influence of the overall impedance. Other kinetic parameters may lead to more shapes of diffusion impedance and will be analyzed in the future.

Further analysis of other parameters in the CGM that still need to be explored include

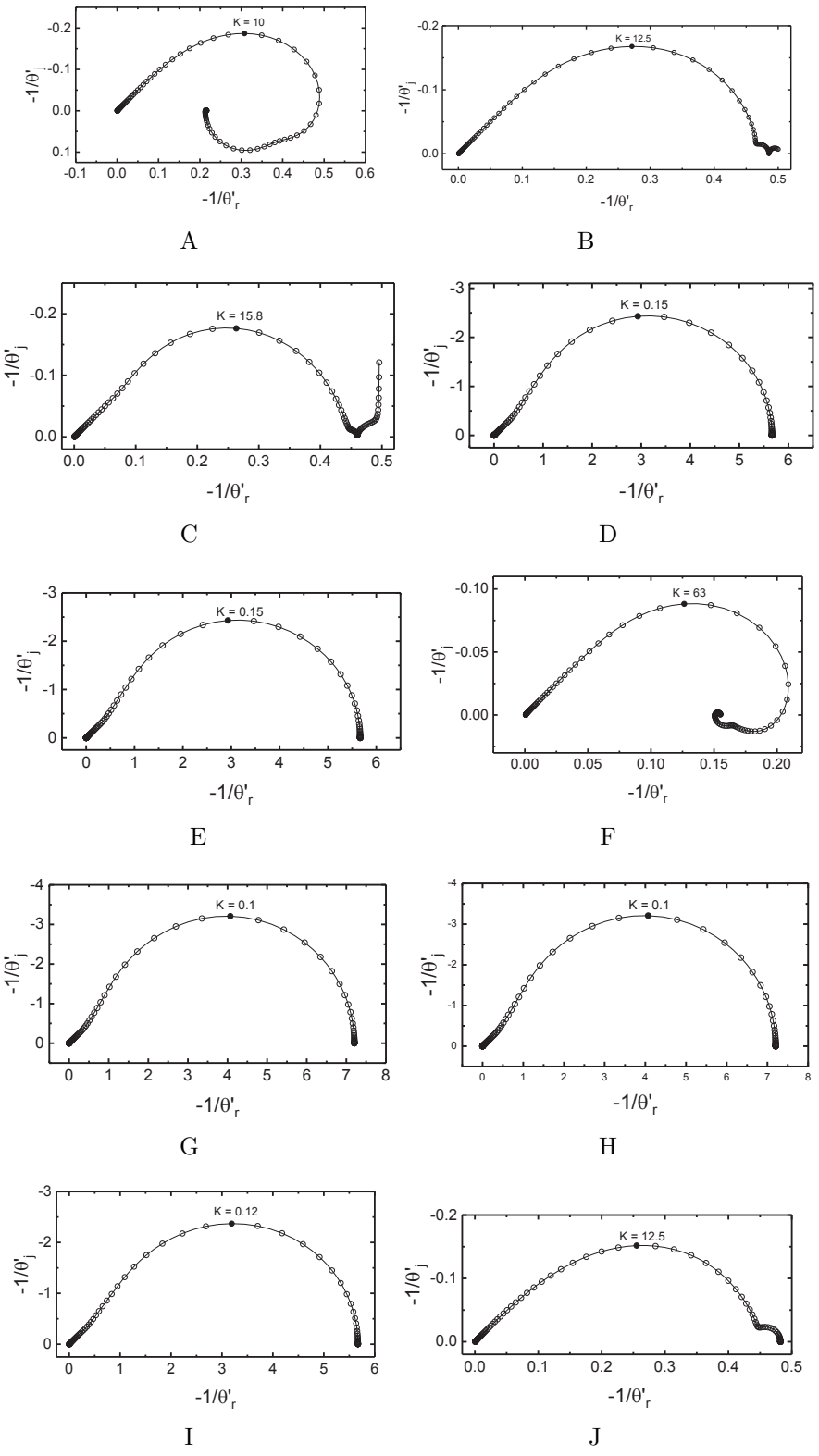


Figure 7-1. Dimensionless diffusion impedances for different kinetic parameters; A) System 1 B) System 2 C) System 3 D) System 4 E) System 5 F) System 6 G) System 7 H) System 8 I) System 9 J) System 10

Table 7-1. Values for kinetic parameter study where the forward rates all have units of $cm^3/mol/s$ and equilibrium rates have units of mol/cm^3.

System	k_{f1}	K_{eq1}	k_{f2}	k_{f3}	K_{eq3}	k_{f4}
1	10^6	10^{-2}	10^2	10^6	10^{-2}	10^2
2	10^5	10^2	10^3	10^5	10^2	10^3
3	10^8	10^1	10^7	10^8	10^1	10^7
4	10^5	10^2	10^3	10^3	10^2	10^1
5	10^6	10^1	10^3	10^3	10^1	10^1
6	10^6	10^1	10^3	10^6	10^1	10^3
7	10^6	10^1	10^3	10^2	10^1	10^1
8	10^3	10^1	10^3	10^2	10^1	10^1
9	10^3	10^2	10^5	10^3	10^2	10^5
10	10^8	10^1	10^7	10^5	10^1	10^4

1. Heterogeneous reaction rates
2. Bulk concentrations of glucose
3. Bulk concentrations of oxygen
4. Activity of glucose oxidase
5. Layer thickness of GOX
6. Layer thickness of GLM
7. Porosity factors for all diffusing species
8. Diffusion coefficients of all species
9. Partition coefficients for all species

Exploring the affects of the parameters will give a deeper understanding of how the CGM works as well as understanding the influence of parameters on the impedance.

7.1.2 Overall Impedance Analysis

The overall impedance analysis still needs to be done. There are possible problems with conducting the overall impedance analysis at the potential used in this dissertation because when operating at the mass-transfer-limiting current the current will not have a large magnitude, ($\triangle I$), so the impedance is extremely large and hard to characterize.

Conducting an impedance analysis at half the mass-transfer-limiting current may show more interesting and characterizable features.

7.2 Influence of Coupled Faradaic and Charging Currents on EIS

The influence of coupled faradaic and charging currents on impedance spectroscopy was simulated and analyzed by Wu et al. [2]. Preliminary experimental results do not perfectly agree with simulations. Further analysis of the experimental results will be conducted along with more finite-element three-dimensional simulations and a finite-difference one-dimensional simulation to further characterize and understand this phenomenon.

7.2.1 History of Coupled Charging and Faradaic Currents

In the 1960s, a controversy emerged over the correct method for determining models for impedance response. The controversy centered on whether faradaic and charging currents in an electrochemical system should be considered to be coupled or separate. Models for impedance typically assume that faradaic and charging currents are not coupled. A previous member of Dr. Orazem's group, Shao-Ling Wu, showed for a rotating disk electrode, that coupling charging and faradaic currents results in frequency dispersion. This effect could be distinguished from the frequency dispersion known to be caused by the disk geometry. The goal of my work is to develop a one-dimensional model which will have lower computational complexity than the models for a disk electrode and will isolate the frequency dispersion caused by coupled charging and faradaic currents. This work will facilitate exploration of the influence of different models for the double layer. During the 1960's a controversy appeared in the electrochemical literature over the correct method for deriving models for impedance. Sluyters treated the total passage of current through an electrode as simply the addition of faradaic and charging current[86]. This approach was criticized by Delahay, who said that the faradaic and charging currents could not be considered separately[87, 88, 89]. He believed that the flux of the reacting species should contribute to the faradaic current and also to charging the double layer. In spite

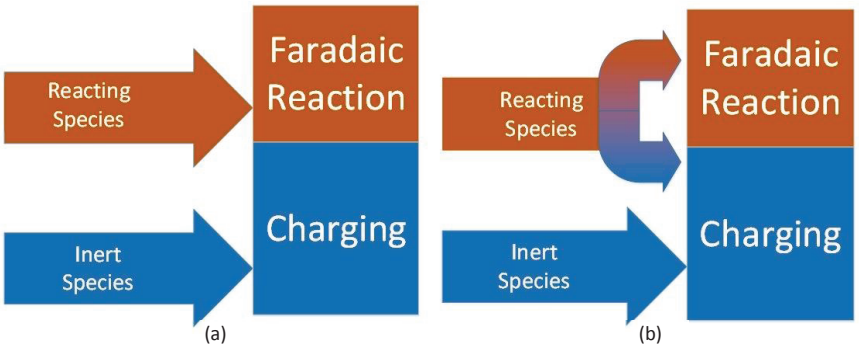


Figure 7-2. Schematic representation illustrating the contribution of the reacting species to the charging of the electrode-electrolyte interface corresponding to: a) the case with *a-priori* separation (APS); and b) the case with no *a-priori* separation (NAPS). Taken from Wu et al.[2]

of Delahay's objections, impedance models today rely on the assumption that faradaic and charging current are separable[5, 90, 91]. The Sluyters approach prevailed in large part because analytic solutions are possible for models that allow decoupling of faradaic and charging currents. To treat the reacting species as contributing to both the faradaic reaction and charging the surface requires coupling an explicit model of the double layer to the convective diffusion equations for each ionic species. Such models require numerical simulations.

Nisancioglu and Newman developed a framework for coupling the charging and faradaic currents[92]. They suggested that the coupling only has a significant effect for well supported electrolytes. Wu et al. followed Nissancioglu's and Newman's approach in their model for a rotating disk electrode[2]. They found that coupling causes a frequency dispersion, even in the presence of a well supported electrolyte.

The difference between a-priori separation (APS) and no a-priori separation (NAPS) of the faradaic and charging current in an electrochemical system is shown schematically in Figure 7-2[2]. With APS, the flux of the reacting species contributes only to the faradaic reaction, Figure 7-2 (a). In the NAPS case, the reacting species contribute to both the faradaic reaction and, along with the inert species, to the charging current, Figure 7-2 (b).

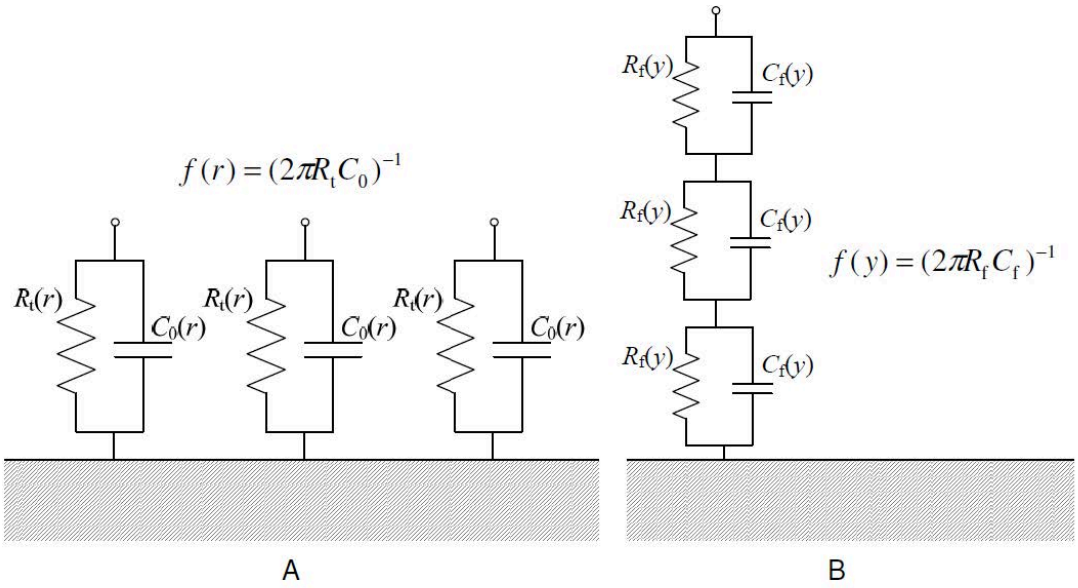


Figure 7-3. Schematic representation of time-constant distribution. Where A) is along the area of electrode-electrolyte interface; and B) is along the direction normal to the electrode surface[1].

7.2.2 Constant-Phase Elements

A constant-phase element (CPE) is a circuit element that displays a constant phase angle. This could be a resistor, capacitor or inductor. However, the electrical double-layer at an electrode does not generally behave as a pure capacitance, but rather an impedance displaying a frequency phase angle different from 90 degrees[82]. The impedance for a CPE is

$$Z_{\mathrm{CPE}} = \frac{1}{(\mathrm{j}\omega)^{\alpha} Q} \tag{7–1}$$

where α and Q are constants. When $\alpha = 1$, Q has units of capacitance. When $\alpha \neq 1$, Q has units of $\mathrm{s}^{\alpha}/\Omega\mathrm{cm}^2$. Frequently, $0.5 < \alpha < 1$ for an electrochemical interface of a real cell.

Non-ideal behavior leading to a CPE can be attributed to the frequency or time-constant distribution. The frequency dispersion can occur along the area of the electrode or along the direction normal to the electrode surface. The surface distribution can arise from

surface differences [83]. A normal distribution can be attributed to composition of oxide layers [84] or to porosity [85]. Figure 7-3 shows the surface and normal distributions using equivalent circuits, where R_t is the charge-transfer resistance, C_0 is the double-layer capacitance, and R_f and C_f are the resistance and capacitance of an oxide film.

The variations of reaction reactivity and double-layer capacitance at the electrode–electrolyte interface cause a frequency or time-constant distribution at the electrode surface. Variations of film properties in an oxide layer can cause a frequency distribution too. These distributions are observed during the impedance measurements in the form of a CPE. The presence of a CPE behavior, is very common even for a homogenous and smooth surface.

7.2.3 Electrochemical Instrumentation

A picture of the electrochemical instrumentation used to conduct the EIS experiments is presented in Figure 7-4. A Ag/AgCl electrode was used as the reference electrode. The working electrode was either platinum or glassy carbon and the counter electrode was a platinum mesh.

7.2.4 CV Curves and EIS Experimental Results

The CV curves in Figure 7-5 shows results for a non-polished (Figure 7-5A) and polished (Figure 7-5B) glassy carbon electrode, respectively, for a 10mM $Fe(CN)_6$(II)/(III) and 0.5M KCl electrolyte. The sweep rate varied from 10 mV/s to 10 V/s. The largest sweep rate of 10 V/s gave the biggest current response. The polarization curves for different rotation rates are shown in Figure 7-6. The polarization curves were taken at a sweep rate of 50 mV/s for rotation rates from 300 RPM to 2400 RPM. The non-polished electrode, Figure 7-6A, approached the mass-transfer-limited current slower then for the polished electrode, Figure 7-6B. For both electrodes, the higher the rotation speeds led to higher current response, but the non-polished electrode had higher currents then the polished electrode. The impedance for 500 RPM for fractions of the limiting current for both the non-polished electrode and polished electrode are presented in Figure 7-7. The

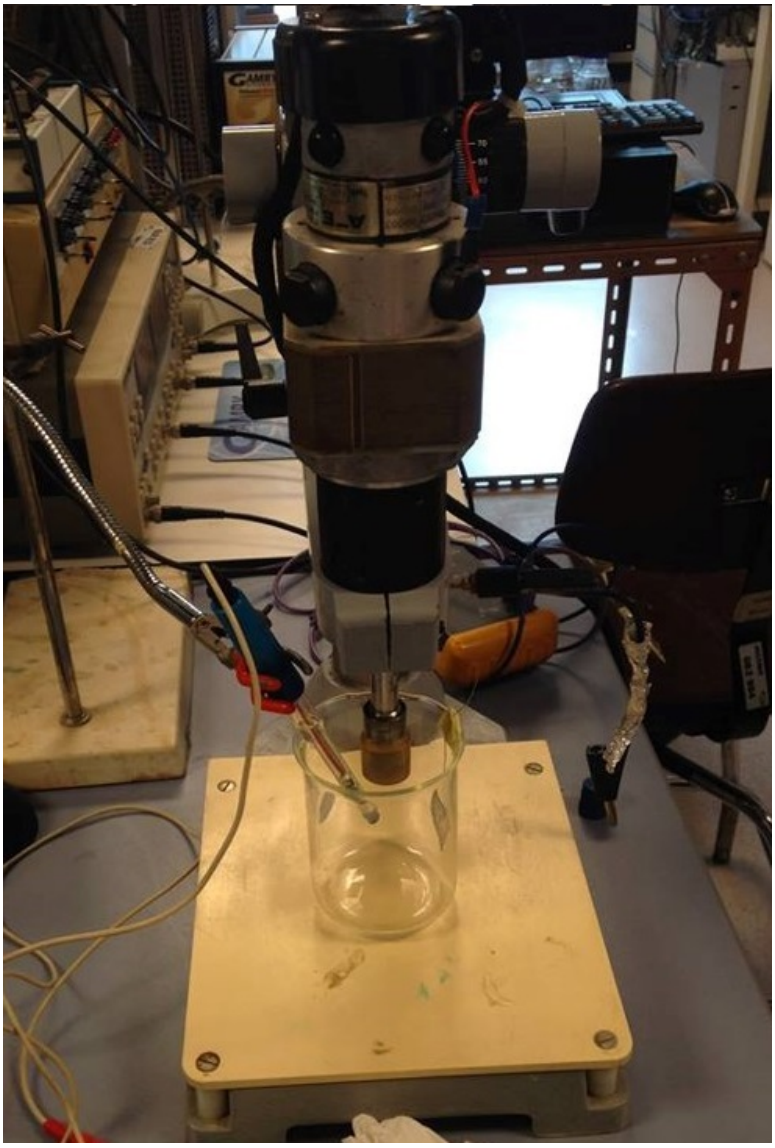
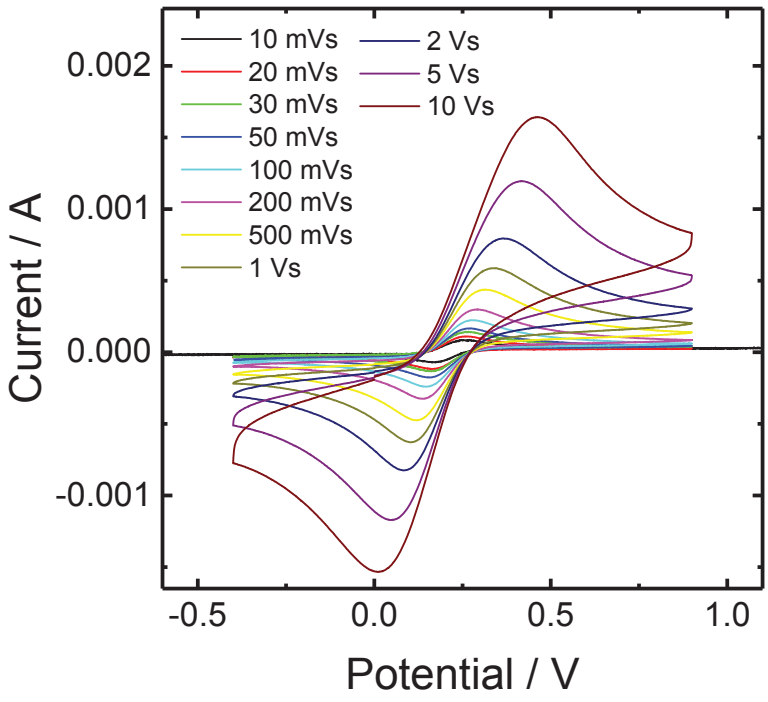


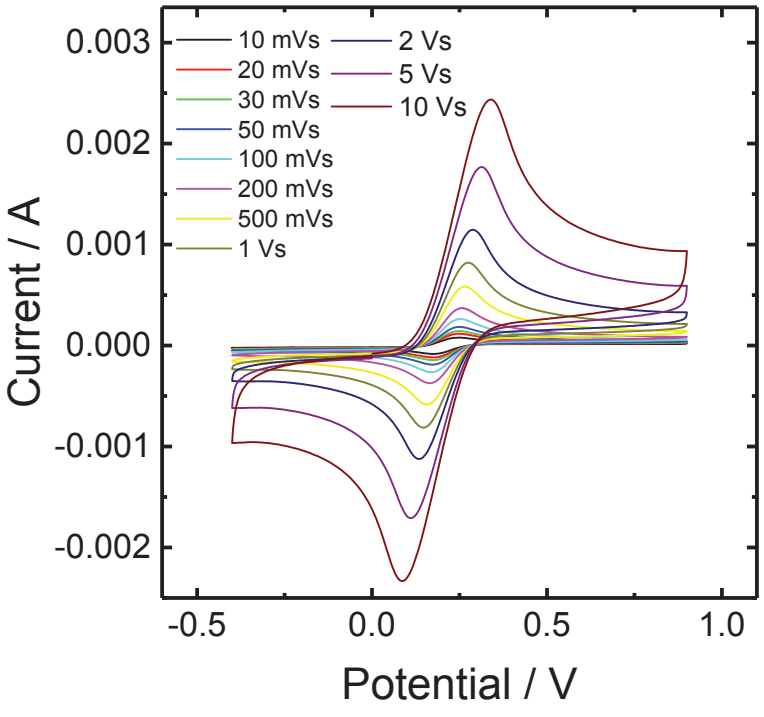
Figure 7-4. Electrochemical instrumentation used to conduct the EIS experiments

non-polished electrode, Figure 7-7A, had impedances that were double the impedances for the polished electrode, Figure 7-7B. The high-frequency capacitive loop was larger for the non-polished electrode.

Another way to look at the impedance data is in the form of an adjusted phase angle plot, presented in Figure 7-8.

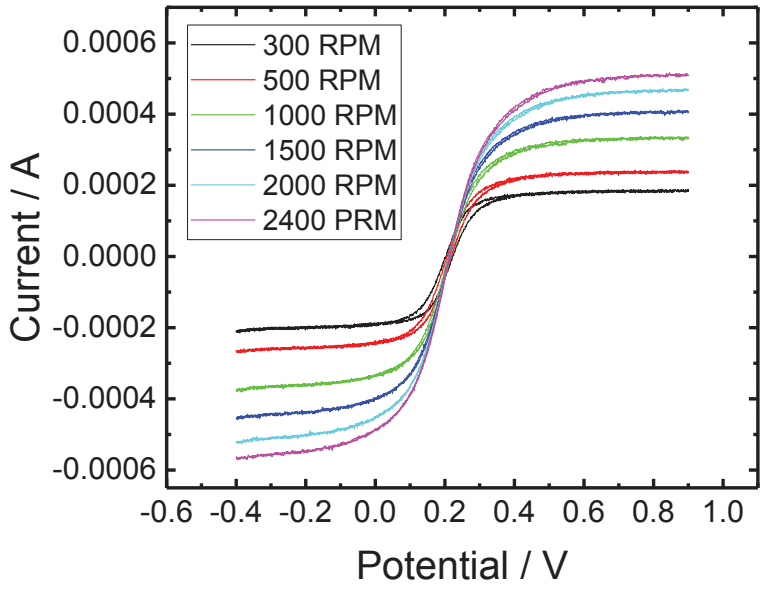


A

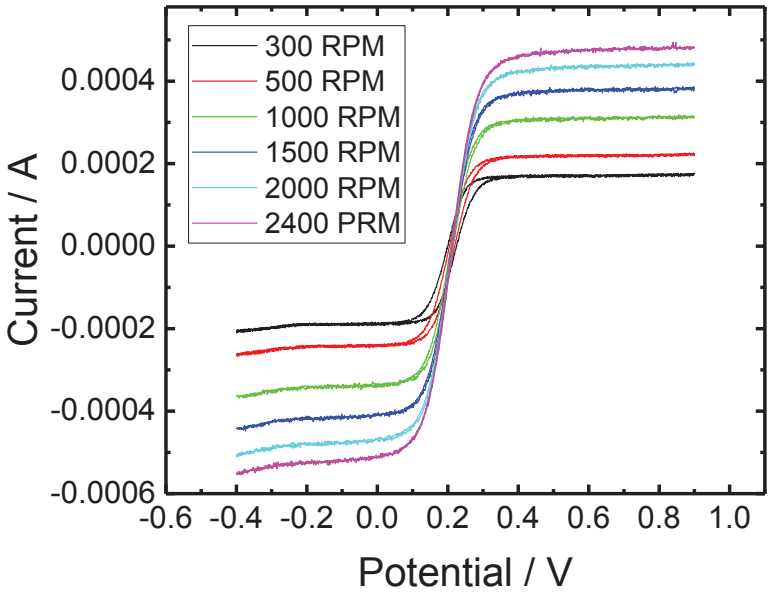


B

Figure 7-5. CV curves with different sweep rates from 10 mV/s to 10 V/s for the A) non-polished and B) polished glassy carbon electrode with a 10mM $Fe(CN)_6$(II)/(III) and 0.5M KCl electrolyte.

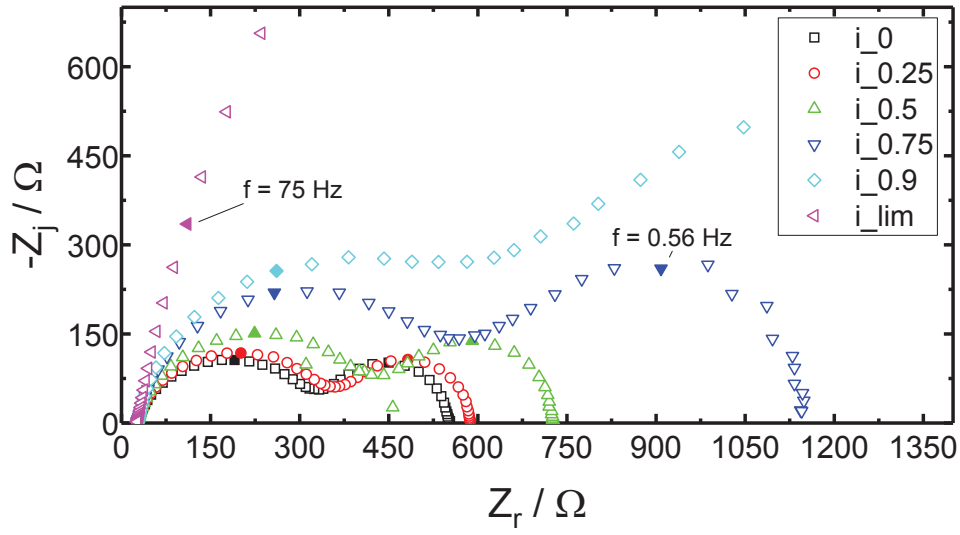


A

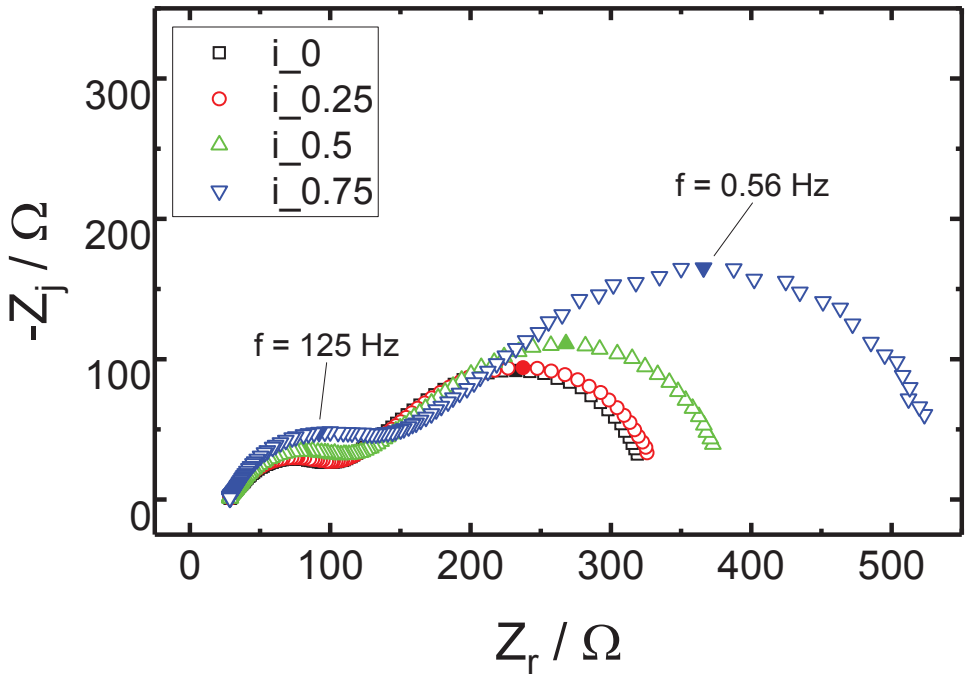


B

Figure 7-6. Polarization curves with different rotation rates from 300 RPM to 2400 RPM for the; A) non-polished and B) polished glassy carbon electrode with a 10mM $Fe(CN)_6$(II)/(III) and 0.5M KCl electrolyte.

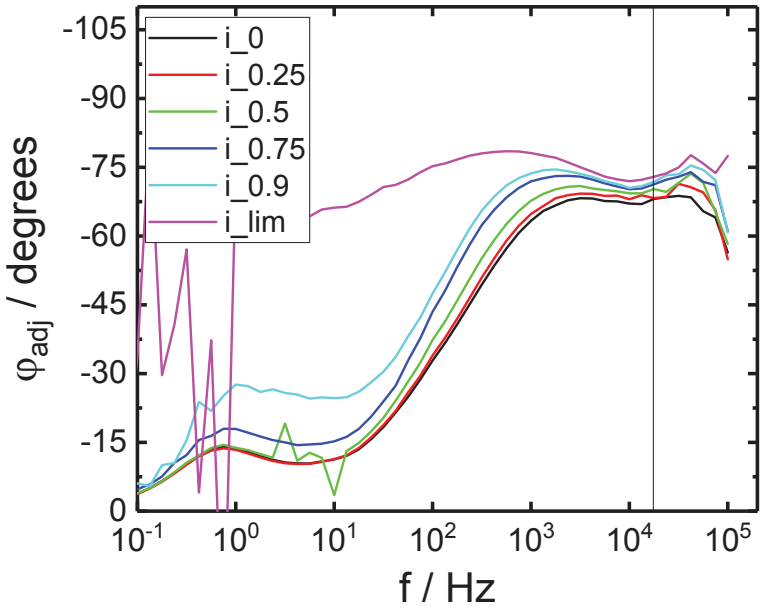


A

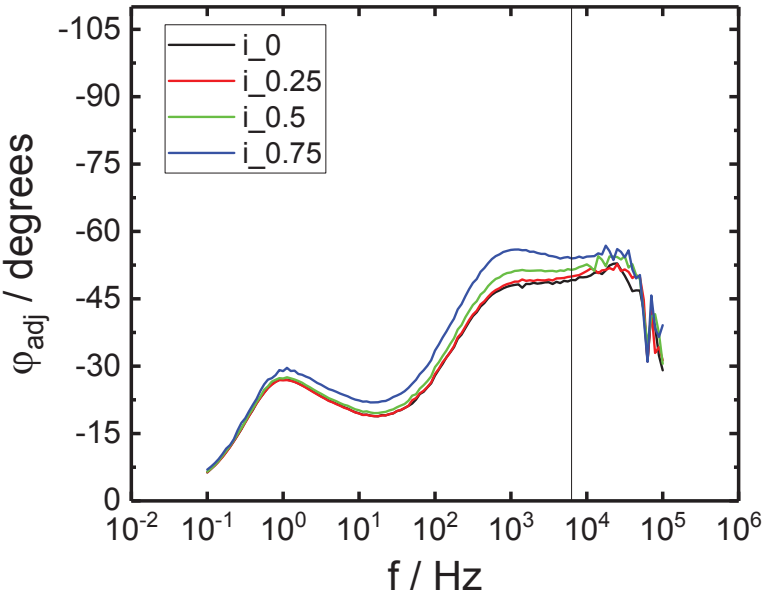


B

Figure 7-7. Impedance spectrum with a rotation speed of 500 RPM for A) non-polished and B) polished glassy carbon electrode with a 10mM $Fe(CN)_6$(II)/(III) and 0.5M KCl electrolyte.



A



B

Figure 7-8. Adjusted phase angle with a rotation speed of 500 RPM for A) non-polished and B) polished glassy carbon electrode with a 10mM $Fe(CN)_6$(II)/(III) and 0.5M KCl electrolyte.

APPENDIX A
BAND

Electrochemical systems can be modeled mathematically as boundary value problems which consist of sets of non-linear, coupled, second-order differential equations [93]. The algorithm BAND was developed in Fortran by Newman. [74] A linear set of coupled, second-order differential equations can be written as

$$\sum_{\mathrm{k}=1}^{\mathrm{n}} a_{\mathrm{i,k}}(x)\frac{\mathrm{d}^2 c_{\mathrm{k}}(x)}{\mathrm{d}x^2} + b_{\mathrm{i,k}}(x)\frac{\mathrm{d}c_{\mathrm{k}}(x)}{\mathrm{d}x} + d_{\mathrm{i,k}}(x)c_{\mathrm{k}}(x) = g_{\mathrm{i}}(x) \tag{A–1}$$

There are n equations of the form of equation (A–1), where k is the index representing the dependent variable and i is the equation number. A finite difference approximation to the derivatives appearing in equation (A–1), accurate to the order h^2.

$$\frac{\mathrm{d}^2 c_{\mathrm{k}}}{\mathrm{d}x^2} = \frac{c_{\mathrm{k}}(x_{\mathrm{j}}+h) + c_{\mathrm{k}}(x_{\mathrm{j}}-h) - 2c_{\mathrm{k}}(x_{\mathrm{j}})}{h^2} + \mathcal{O}(h^2) \tag{A–2}$$

and

$$\frac{\mathrm{d}c_{\mathrm{k}}}{\mathrm{d}x} = \frac{c_{\mathrm{k}}(x_{\mathrm{j}}+h) - c_{\mathrm{k}}(x_{\mathrm{j}}-h)}{2h} + \mathcal{O}(h^2) \tag{A–3}$$

These are the approximations to the derivatives at the point x_{j} in the mesh used for numerical solution. The size between mesh points is h. Equations (A–2) and (A–3) go into equation (A–1) to form

$$\sum_{\mathrm{k}=1}^{\mathrm{n}} A_{\mathrm{i,k}}(j)C_{\mathrm{k}}(\mathrm{j}-1) + B_{\mathrm{i,k}}(j)C_{\mathrm{k}}(\mathrm{j}) + D_{\mathrm{i,k}}(j)c_{\mathrm{k}}(x_{\mathrm{j}+1}) = G_{\mathrm{i}}(j) \tag{A–4}$$

which is evaluated at position $x_{\mathrm{j}-1} = x_{\mathrm{j}} - h$. $B_{\mathrm{i,k}}(j)$ is the coefficient, in equation i, at the position x_{j} of the dependent variable c_{k}. The coefficients in equation ((A–4)) are

$$A_{\mathrm{i,k}}(\mathrm{j}) = a_{\mathrm{i,k}}(x_{\mathrm{j}}) - \frac{h}{2}b_{\mathrm{i,k}}(x_{\mathrm{j}}) \tag{A–5}$$

$$B_{\mathrm{i,k}}(\mathrm{j}) = -2a_{\mathrm{i,k}}(x_{\mathrm{j}}) - h^2 d_{\mathrm{i,k}}(x_{\mathrm{j}}) \tag{A–6}$$

$$D_{\mathrm{i,k}}(\mathrm{j}) = a_{\mathrm{i,k}}(x_{\mathrm{j}}) + \frac{h}{2}b_{\mathrm{i,k}}(x_{\mathrm{j}}) \tag{A–7}$$

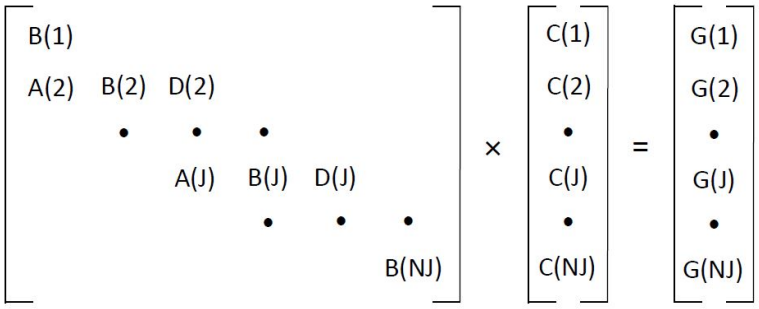


Figure A-1. Matrix defining BAND[93].

and

$$G_{\mathrm{i}}(\mathrm{j}) = h^2 g_{\mathrm{i}}(x_{\mathrm{j}}) \tag{A–8}$$

The simple boundary conditions

$$\sum_{\mathrm{k}=1}^{\mathrm{n}} e_{\mathrm{i,k}}(x) c_{\mathrm{k}} = f_{\mathrm{i}}(x) \tag{A–9}$$

for the first mesh point can be written

$$\sum_{\mathrm{k}=1}^{\mathrm{n}} B_{\mathrm{i,k}}(1) C_{\mathrm{k}}(1) = G_{\mathrm{i}}(1) \tag{A–10}$$

and the last mesh point can be written

$$\sum_{\mathrm{k}=1}^{\mathrm{n}} B_{\mathrm{i,k}}(NJ) C_{\mathrm{k}}(NJ) = G_{\mathrm{i}}(NJ) \tag{A–11}$$

These governing difference equations, equations (A–4), (A–10), and (A–11), can be written conveniently in matrix form (see Figure A-1). Or as

$$\mathbf{MT} = \mathbf{Z} \tag{A–12}$$

where the $\mathbf{M}$ is an n by n matrix and $\mathbf{T}$ and $\mathbf{G}$ are n-dimensional vectors. Equation (A–12) can be solved by decomposing $\mathbf{M}$ into triangular matrices. The upper and lower triangle matrices will be denoted $\mathbf{U}$ and $\mathbf{L}$ so

$$\mathbf{MT} = \mathbf{LUT} = \mathbf{Z} \tag{A–13}$$

Letting

$$\mathbf{UT} = \xi \tag{A–14}$$

equation (A–13) becomes

$$\mathbf{L}\xi = \mathbf{Z} \tag{A–15}$$

The unknown $\mathbf{T}$ can be solved by equation (A–14) after $\mathbf{L}$, $\mathbf{U}$ and ξ are known. Forward and backward substitutions can be made to solve equations (A–15) and (A–14). These governing matrix equations can be solved with Newman's BAND algorithm.

The behavior of BAND was studied by Curtis et al. [94] and White [93]. Curtis et al. compared the BAND method to the deBoor method. Sometimes the BAND algorithm signals the matrix as falsely singular and the error "DETERMINANT = 0 AT J = 2" is called and in this case it is better to rewrite the finite difference equations or use a different method for solving the problem.

The following codes are Newman's BAND algorithm and matrix inversion code written in MATLAB® and then FORTRAN. The MATLAB® code was used for models for convective-diffusion impedance, Chapter 3, and the FORTRAN code was used for all the other models in this dissertation, Chapters 4 and 5.

Code A.1. MATLAB BAND and MATINV code

```
% ***************************************************************
%  MATLAB version of BAND
%  The following is a version of BAND (and MATINV) obtained by
%   translating the FORTRAN version in Newman.
% ***************************************************************

function [y]=band(J) % i.e. 'J' = 'I' from test_band
global A B C D E G X Y NPOINTS N NJ NP1 DETERMINANT
% ***************************************************************
% Case where J negative
% ***************************************************************
if (J-2) < 0 % i.e. image point (I=1) outside boundary (left hand side)

    NP1=N+1; % I runs from 1:NPOINTS and NPOINTS = NJ, while N = # eqn
    for I=1:N
        D(I,2*N+1)=G(I);
        for L=1:N
            LPN=L+N;
            D(I,LPN)=X(I,L);
        end
    end
    %  ***************************************************************
    %   Calling Matinv
    %  ***************************************************************
    matinv(N,2*N+1);
    if (DETERMINANT == 0)
        sprintf('DETERMINANT = 0 at J= %6.3f',J)
        return % stod tidligere break
    else
        for K=1:N
            E(K,NP1,1)=D(K,2*N+1);
            for L=1:N
                E(K,L,1) = -D(K,L);
                LPN = L+N;
                X(K,L) = -D(K,LPN);
            end
        end
        return
    end
end
% ***************************************************************
% Case where J-2 equal to zero (I=2, i.e. first real point)
% ***************************************************************
if((J-2)==0)

    for I=1:N
        for K=1:N
            for L=1:N
                D(I,K)=D(I,K)+A(I,L)*X(L,K);
            end
        end
    end
end
% ***************************************************************
% Case where J-2 is greater than zero
% ***************************************************************
```

```
57 if (J-NJ) < 0
58
59     for I=1:N
60         D(I,NP1) = -G(I);
61         for L=1:N
62             D(I,NP1)=D(I,NP1)+A(I,L)*E(L,NP1,J-1);
63             for K=1:N
64                 B(I,K)=B(I,K)+A(I,L)*E(L,K,J-1);
65             end
66         end
67     end
68 else
69
70     for I=1:N
71         for L=1:N
72             G(I)=G(I)-Y(I,L)*E(L,NP1,J-2);
73             for M=1:N
74                 A(I,L)=A(I,L)+Y(I,M)*E(M,L,J-2);
75             end
76         end
77     end
78     for I=1:N
79         D(I,NP1) = -G(I);
80         for L=1:N
81             D(I,NP1)=D(I,NP1)+A(I,L)*E(L,NP1,J-1);
82             for K=1:N
83                 B(I,K)=B(I,K)+A(I,L)*E(L,K,J-1);
84             end
85         end
86     end
87 end
88 %  ************************************************************************
89 %    Calling Matinv
90 %  ************************************************************************
91
92 matinv(N,NP1);
93 if DETERMINANT ~= 0
94
95     for K=1:N
96         for M=1:NP1
97             E(K,M,J) = -D(K,M);
98         end
99     end
100 else
101     sprintf('DETERMINANT = 0 at J= %6.3f',J)
102     return % stod tidligere break
103 end
104
105 if (J-NJ)<0
106
107     return
108 else
109
110     for K = 1:N
111
112         C(K,J) = E(K,NP1,J);
113     end
114     for JJ = 2:NJ
```

```
115
116         M = NJ - JJ + 1;
117          for K=1:N
118              C(K,M) = E(K,NP1,M);
119               for L=1:N
120                   C(K,M)=C(K,M)+E(K,L,M)*C(L,M+1);
121              end
122          end
123      end
124      for L=1:N
125
126          for K=1:N
127              C(K,1) = C(K,1) + X(K,L)*C(L,3);
128          end
129      end
130 end
131
132 return
133
134 %*********************************************************************
```

Code A.2. FORTRAN BAND Code

```
C       SUBROUTINE  BAND(J)
C
      SUBROUTINE BAND(J)
      IMPLICIT DOUBLE PRECISION (A-H,O-Z)
C
      DIMENSION E(4,5,100001)
      COMMON/BAB/ A(4,4),B(4,4),C(4,100001),D(4,9),G(4),X(4,4),Y(4,4)
      COMMON/NSN/ N,NJ
     k
      SAVE E, NP1
 101  FORMAT(15H DETERM=0 AT J=,I4)
      IF (J-2) 1,6,8
   1  NP1=N+1
      DO 2 I=1,N
      D(I,2*N+1)=G(I)
      DO 2 L=1,N
      LPN=L+N
   2  D(I,LPN)=X(I,L)
      CALL MATINV(N, 2*N+1,DETERM)
C
      IF (DETERM) 4,3,4
   3  PRINT 101,J
   4  DO 5 K=1,N
      E(K,NP1,1)=D(K,2*N+1)
      DO 5 L=1,N
      E(K,L,1)=-D(K,L)
      LPN=L+N
   5  X(K,L)=-D(K,LPN)
      RETURN
C
   6  DO 7 I=1,N
      DO 7 K=1,N
      DO 7 L=1,N
   7  D(I,K)=D(I,K)+A(I,L)*X(L,K)
   8  IF (J-NJ) 11,9,9
   9  DO 10 I=1,N
      DO 10 L=1,N
      G(I)=G(I)-Y(I,L)*E(L,NP1,J-2)
      DO 10 M=1,N
  10  A(I,L)=A(I,L)+Y(I,M)*E(M,L,J-2)
  11  DO 12 I=1,N
      D(I,NP1)=-G(I)
      DO 12 L=1,N
      D(I,NP1)=D(I,NP1)+A(I,L)*E(L,NP1,J-1)
      DO 12 K=1,N
  12  B(I,K)=B(I,K)+A(I,L)*E(L,K,J-1)
C
      CALL MATINV(N,NP1,DETERM)
      IF (DETERM) 14, 13, 14
  13  PRINT 101,J
  14  DO 15 K=1,N
      DO 15 M=1,NP1
  15  E(K,M,J)=-D(K,M)
      IF (J-NJ) 20,16,16
  16  DO 17 K=1,N
  17  C(K,J)=E(K,NP1,J)
```

```
57          DO  18  JJ=2,NJ
58          M=NJ-JJ+1
59          DO  18  K=1,N
60          C(K,M)=E(K,NP1,M)
61          DO  18  L=1,N
62       18 C(K,M)=C(K,M)+E(K,L,M)*C(L,M+1)
63          DO  19  L=1,N
64          DO  19  K=1,N
65       19 C(K,1)=C(K,1)+X(K,L)*C(L,3)
66
67       20 RETURN
68
69          END
```

Code A.3. FORTRAN MATINV Code

```
C       SUBROUTINE MATINV
        SUBROUTINE MATINV(N,M,DETERM)
        IMPLICIT DOUBLE PRECISION (A-H,O-Z)
        COMMON/BAB/ A(4,4), B(4,4),C(4,100001),D(4,9),G(4),X(4,4),Y(4,4)
        COMMON/NSN/ NTEMP, NJ
        DIMENSION ID(4)
        DETERM=1.01
        DO 1 I=1,N
    1   ID(I)=0
        DO 18 NN=1,N
        BMAX=1.1
        DO 6 I=1,N
        IF(ID(I).NE.0) GO TO 6
        BNEXT=0.0
        BTRY=0.0
        DO 5 J=1,N
        IF (ID(J).NE.0) GO TO 5
        IF (DABS(B(I,J)).LE.BNEXT) GO TO 5
        BNEXT=DABS(B(I,J))
        IF (BNEXT.LE.BTRY) GO TO 5
        BNEXT=BTRY
        BTRY=DABS(B(I,J))
        JC=J
    5   CONTINUE
        IF(BNEXT.GE.BMAX*BTRY) GO TO 6
        BMAX=BNEXT/BTRY
        IROW=I
        JCOL=JC
    6   CONTINUE
        IF (ID(JC).EQ.0) GO TO 8
        DETERM=0.0
        RETURN
    8   ID(JCOL)=1
        IF (JCOL.EQ.IROW) GO TO 12
        DO 10 J=1,N
        SAVE=B(IROW,J)
        B(IROW,J)=B(JCOL,J)
   10   B(JCOL,J)=SAVE
        DO 11 K=1,M
        SAVE=D(IROW,K)
        D(IROW,K)=D(JCOL,K)
   11   D(JCOL,K)=SAVE
   12   F=1.0/B(JCOL,JCOL)
        DO 13 J=1,N
   13   B(JCOL,J)=B(JCOL,J)*F
        DO 14 K=1,M
   14   D(JCOL,K)=D(JCOL,K)*F
        DO 18 I=1,N
        IF (I.EQ.JCOL)  GO TO 18
        F=B(I,JCOL)
        DO 16   J=1,N
   16   B(I,J)=B(I,J)-F*B(JCOL,J)
        DO 17 K=1,M
   17   D(I,K)=D(I,K)-F*D(JCOL,K)
   18   CONTINUE
        RETURN
```

```
57          END
```

APPENDIX B
CODES FOR ROTATING DISK ELECTRODE

This appendix contains the different MatLab codes used to solve the convective-diffusion equation for a rotating disk electrode in Chapter 3. The first MatLab code is to solve the infinite Schmidt convective-diffusion equation or the term 0 finite Schmidt convective-diffusion equation. The second Matlab code solve for term 1 of the finite Schmidt convective-diffusion equation. The last MatLab code solves for term 2 of the finite Schmidt convective-diffusion equation. The solutions to each code are imputed into equation (3–32) to obtain a solution for the finite Schmidt convective-diffusion equation.

Code B.1. Finite Schmidt Convection Diffusion Term 0 for rotating disk electrode

```
%This problem solves for the first term of the convective-diffusion equation with a finite schmidt number
%d^2(theta0)/d(xi)^2+3*xi^2*d(theta0)/d(xi)-j*K*theta0=0

clc; close all;clear all;
format longE;
global A B C D G X Y N NJ

h=0.01;                         %Step-size
logK=-2:.05:2;                  %Frequency range 0.001 to 1000
K=10.^logK;
K
x=0:h:10;                       %Range of x
NJ=length(x);
NJ
ZZd=zeros(1,length(K));         %Place holder for values of diffusion impedance

c_0=1;              %Boundary condition for x=0
c_e=0;              %Boundary condition for x=end (NJ)
tol=0;              %Absolute tolerance

%Define initial guess
conc(1,1:NJ)=0.5;           %Conc(1,-) is the real component of theta0
conc(2,1:NJ)=0.5;           %Conc(2,-) is the imaginary component of theta0
N=2;

for kk=1:length(K)

error=1;
jcount=0;
jcountmax=4;                %Number of iterations the program will allow to converge
while error>=tol;
X=zeros(N,N);
Y=X;

jcount=jcount+1;
for J=1:NJ;
    A=zeros(N,N);
    B=A;
    D=A;
    %Boundary condition at J=1
    if J==1;
        G(1)=conc(1,J)-c_0;     %Real equation, so BC at electrode surface is 1
        B(1,1)=-1;

        G(2)=conc(2,J)-0;       %Imaginary equation, so BC at electrode surface is 0
        B(2,2)=-1;

        error=abs(G(1))+abs(G(2));

    %Region between boundaries
    elseif (J<=(NJ-1));
```

```
52          G(1)=conc(1,J+1)-2*conc(1,J)+conc(1,J-1)+h^2*K(kk)*conc(2,J)+1.5*h^3*(
    J-1)^2*(conc(1,J+1)-conc(1,J-1));
53          A(1,1)=-1+1.5*h^3*(J-1)^2;
54          B(1,1)=2;
55          D(1,1)=-1-1.5*h^3*(J-1)^2;
56          B(1,2)=-h^2*K(kk);
57
58          G(2)=conc(2,J+1)-2*conc(2,J)+conc(2,J-1)-h^2*K(kk)*conc(1,J)+1.5*h^3*(
    J-1)^2*(conc(2,J+1)-conc(2,J-1));
59          A(2,2)=-1+1.5*h^3*(J-1)^2;
60          B(2,2)=2;
61          D(2,2)=-1-1.5*h^3*(J-1)^2;
62          B(2,1)=h^2*K(kk);
63
64          error=error+abs(G(1))+abs(G(2));
65
66      %Boundary condition at J=NJ
67      else
68          G(1)=conc(1,J)-0;      %BC at far away point is always 0
69          B(1,1)=-1;
70
71          G(2)=conc(2,J)-0;      %BC at far away point is always 0
72          B(2,2)=-1;
73
74          error=error+abs(G(1))+abs(G(2));
75       end;
76
77 band(J)
78 end;
79 conc=conc+C;          %C comes from band(J) program
80
81 if jcount>=jcountmax, break, end;
82 end;
83
84 theta0=complex(conc(1,:),conc(2,:));                                    %Creating
    theta from conc(1,-) and conc(2,-) values
85 dtheta0=gamma(4/3)*(-theta0(3)+4*theta0(2)-3*theta0(1))/(2*h)      %dtheta
    accurate to h^2
86
87
88 Zd=-1/dtheta0;                %turning theta into Impedance
89
90 % str=sprintf('K = %g: Zd = %.10e + j%.10e', K(kk), real(Zd),imag(Zd)); disp(
    str);
91 % str=sprintf('dtheta0 = %g', dtheta0); disp(str);
92 % str=sprintf('%.10e', K(kk)); disp(str);
93 % str=sprintf('%.10e', real(Zd)); disp(str);
94 % str=sprintf('%.10e', imag(Zd)); disp(str);
95
96 figure(1)
97 plot(x,conc(1,:),'-r')
98 hold on;
99 plot(x,conc(2,:),'-g')
100 axis([0 2 -.4 1]);
101 legend('theta_r','theta_i');
102 title('Dimensionless Concentration away from Electrode Surface');
103 xlabel('length');
104 ylabel('theta');
```

```
105
106 ZZd(kk)=Zd;
107 Ktemp=K(kk);
108 fname=sprintf('theta0_%i',kk);
109 save(fname,'theta0','Ktemp','h');
110 end
111
112 fname=sprintf('zero_order_%i.txt',NJ);
113 fid = fopen(fname, 'wt');
114 fprintf(fid, '%14.6e; %14.6e \n', h,h^2);
115 for kk=1:length(K);
116 fprintf(fid, '%12.6e; %20.12e; %20.12e; \n', K(kk),real(ZZd(kk)),imag(ZZd(kk))
    );
117 end;
118 fclose(fid);
119
120 figure(2)
121 plot(real(ZZd),-imag(ZZd),'-ks');hold on; axis equal;
122 Zfilm=tanh(sqrt(j*K))./sqrt(j*K);
123 plot(real(Zfilm),-imag(Zfilm),'--b');
124 legend('MatLab Data','Finite Film Thickness tanh(sqrt(j*K))/sqrt(j*K)');
125 title('Nyquist plot');
126 xlabel('Real part of Impedance');
127 ylabel('Imaginary part of Impedance');
128
129 figure(3)
130 loglog(K, -imag(ZZd),'--k'); hold on;
131 loglog(K, real(ZZd),'--b');
132 legend ('-Imaginary','Real');
133 title('Impedance vs Frequency');
134 xlabel('Dimensionless Frequency');
135 ylabel('Impedance');
```

Code B.2. Finite Schmidt Convection Diffusion Term 1 for rotating disk electrode

```
%This problem solves for the first term of the convective-diffusion equation with a finite schmidt number
%d^2(theta0)/d(xi)^2+3*xi^2*d(theta0)/d(xi)-j*K*theta0=0

clc; close all;clear all;
format longE;
global A B C D G X Y N NJ

h=0.01;                         %Step-size
logK=-2:.05:2;                  %Frequency range 0.001 to 1000
K=10.^logK;
x=0:h:10;                        %Range of x
NJ=length(x);
ZZd=zeros(1,length(K));         %Place holder for values of diffusion impedance
aa=0.51023;
bb=-0.61592;
cc=(3/aa^4)^(1/3);

c_0=1;              %Boundary condition for x=0
c_e=0;              %Boundary condition for x=end (NJ)
tol=0;              %Absolute tolerance

%Define initial guess
conc(1,1:NJ)=0.5;           %Conc(1,-) is the real component of theta0
conc(2,1:NJ)=0.5;             %Conc(2,-) is the imaginary component of theta0
N=2;

for kk=1:length(K)
fname=sprintf('theta0_%i',kk);
load(fname);
flag=0;if(abs(Ktemp-K(kk))~=0);flag=3;quit;end;

error=1;
jcount=0;
jcountmax=4;                %Number of iterations the program will allow to converge
while error>=tol;
X=zeros(N,N);
Y=X;

jcount=jcount+1;
for J=1:NJ;
    A=zeros(N,N);
    B=A;
    D=A;
    %Boundary condition at J=1
    if J==1;
        G(1)=conc(1,J);      %Real equation, so BC at electrode surface is 0
        B(1,1)=-1;

        G(2)=conc(2,J);         %Imaginary equation, so BC at electrode surface is 0
        B(2,2)=-1;

        error=abs(G(1))+abs(G(2));

```

```
54      %Region between boundaries
55      elseif (J<=(NJ-1));
56          G(1)=conc(1,J+1)-2*conc(1,J)+conc(1,J-1)+h^2*K(kk)*conc(2,J)+1.5*h^3*(
        J-1)^2*(conc(1,J+1)-conc(1,J-1))...
57              -cc*(h^4)*((J-1)^3)*(real(theta0(J+1))-real(theta0(J-1)))/2;
58          A(1,1)=-1+1.5*h^3*(J-1)^2;
59          B(1,1)=2;
60          D(1,1)=-1-1.5*h^3*(J-1)^2;
61          B(1,2)=-h^2*K(kk);
62
63          G(2)=conc(2,J+1)-2*conc(2,J)+conc(2,J-1)-h^2*K(kk)*conc(1,J)+1.5*h^3*(
        J-1)^2*(conc(2,J+1)-conc(2,J-1))...
64              -cc*(h^4)*((J-1)^3)*(imag(theta0(J+1))-imag(theta0(J-1)))/2;
65          A(2,2)=-1+1.5*h^3*(J-1)^2;
66          B(2,2)=2;
67          D(2,2)=-1-1.5*h^3*(J-1)^2;
68          B(2,1)=h^2*K(kk);
69
70          error=error+abs(G(1))+abs(G(2));
71
72      %Boundary condition at J=NJ
73      else
74          G(1)=conc(1,J)-0;      %BC at far away point is always 0
75          B(1,1)=-1;
76
77          G(2)=conc(2,J)-0;      %BC at far away point is always 0
78          B(2,2)=-1;
79
80          error=error+abs(G(1))+abs(G(2));
81       end;
82
83  band(J)
84  end;
85  conc=conc+C;          %C comes from band(J) program
86
87  if jcount>=jcountmax, break, end;
88  end;
89
90  theta1=complex(conc(1,:),conc(2,:));                                    %Creating
        theta from conc(1,-) and conc(2,-) values
91  dtheta1=gamma(4/3)*(-theta1(3)+4*theta1(2)-3*theta1(1))/(2*h);      %dtheta
        accurate to h^2
92  dtheta0=gamma(4/3)*(-theta0(3)+4*theta0(2)-3*theta0(1))/(2*h);
93
94  Zd=dtheta1/(dtheta0^2);                %turning theta into Impedance
95
96  %str=sprintf('K = %.10e: Zd = %.10e + j%.10e', K(kk), real(Zd),imag(Zd)); disp
        (str);
97  %str=sprintf('dtheta1 = %g', dtheta1); disp(str);
98
99  str=sprintf('%.10e', K(kk)); disp(str);
100 str=sprintf('%.10e', real(Zd)); disp(str);
101 str=sprintf('%.10e', imag(Zd)); disp(str);
102
103 figure(1)
104 plot(x,conc(1,:),'-r')
105 hold on;
106 plot(x,conc(2,:),'-g')
```

```
107 axis([0 2 -.4 1]);
108 legend('theta_r','theta_i');
109 title('Dimensionless Concentration away from Electrode Surface');
110 xlabel('length');
111 ylabel('theta');
112
113 ZZd(kk)=Zd;
114 Ktemp=K(kk);
115 fname=sprintf('theta1_%i',kk);
116 save(fname,'theta1','Ktemp','h');
117 end
118
119 fname=sprintf('first_order_%i.txt',NJ);
120 fid = fopen(fname, 'wt');
121 fprintf(fid, '%14.6e; %14.6e \n', h,h^2);
122 for kk=1:length(K);
123 fprintf(fid, '%12.6e; %20.12e; %20.12e; \n', K(kk),real(ZZd(kk)),imag(ZZd(kk))
    );
124 end;
125 fclose(fid);
126
127 figure(2)
128 plot(real(ZZd),-imag(ZZd),'-ks');hold on; axis equal;
129 title('Nyquist plot');
130 xlabel('Real part of Impedance');
131 ylabel('Imaginary part of Impedance');
132
133 figure(3)
134 semilogx(K, -imag(ZZd),'--k'); hold on;
135 semilogx(K, real(ZZd),'--b');
136 legend ('-Imaginary','Real');
137 title('Impedance vs Frequency');
138 xlabel('Dimensionless Frequency');
139 ylabel('Impedance');
```

Code B.3. Finite Schmidt Convection Diffusion Term 2 for rotating disk electrode

```
%This problem solves for the first term of the convective-diffusion equation with a finite schmidt number
%d^2(theta0)/d(xi)^2+3*xi^2*d(theta0)/d(xi)-j*K*theta0=0

clc; close all;clear all;
format longE;
global A B C D G X Y N NJ

h=0.001;                        %Step-size
logK=-5:.05:5;                   %Frequency range 0.001 to 1000
K=10.^logK;
x=0:h:10;                       %Range of x
NJ=length(x);
ZZd=zeros(1,length(K));         %Place holder for values of diffusion impedance
aa=0.51023;
bb=-0.61592;
cc=(3/aa^4)^(1/3);

c_0=1;              %Boundary condition for x=0
c_e=0;              %Boundary condition for x=end (NJ)
tol=0;              %Absolute tolerance

%Define initial guess
conc(1,1:NJ)=0.5;           %Conc(1,-) is the real component of theta0
conc(2,1:NJ)=0.5;             %Conc(2,-) is the imaginary component of theta0
N=2;

for kk=1:length(K)
fname=sprintf('theta0_%i',kk); % load solution for Theta-0
load(fname);
flag=0;if(abs(Ktemp-K(kk))~=0);flag=3;quit;end;
fname=sprintf('theta1_%i',kk); % load solution for Theta-1
load(fname);
flag=0;if(abs(Ktemp-K(kk))~=0);flag=3;quit;end;

error=1;
jcount=0;
jcountmax=4;                %Number of iterations the program will allow to converge
while error>=tol;
X=zeros(N,N);
Y=X;

jcount=jcount+1;
for J=1:NJ;
    A=zeros(N,N);
    B=A;
    D=A;
    %Boundary condition at J=1
     if J==1;
         G(1)=conc(1,J);      %Real equation, so BC at electrode surface is 0
         B(1,1)=-1;

         G(2)=conc(2,J);          %Imaginary equation, so BC at electrode surface is 0
         B(2,2)=-1;
```

```
54
55          error=abs(G(1))+abs(G(2));
56
57      %Region between boundaries
58      elseif (J<=(NJ-1));
59          G(1)=conc(1,J+1)-2*conc(1,J)+conc(1,J-1)+h^2*K(kk)*conc(2,J)+1.5*h^3*(
    J-1)^2*(conc(1,J+1)-conc(1,J-1))...
60              -bb*(3/aa)^(5/3)*(h^5)*((J-1)^4)*(real(theta0(J+1))-real(theta0(J
    -1)))/12  ...
61              -cc*(h^4)*((J-1)^3)*(real(theta1(J+1))-real(theta1(J-1)))/2;
62          A(1,1)=-1+1.5*h^3*(J-1)^2;
63          B(1,1)=2;
64          D(1,1)=-1-1.5*h^3*(J-1)^2;
65          B(1,2)=-h^2*K(kk);
66
67          G(2)=conc(2,J+1)-2*conc(2,J)+conc(2,J-1)-h^2*K(kk)*conc(1,J)+1.5*h^3*(
    J-1)^2*(conc(2,J+1)-conc(2,J-1))...
68              -bb*(3/aa)^(5/3)*(h^5)*((J-1)^4)*(imag(theta0(J+1))-imag(theta0(J
    -1)))/12  ...
69              -cc*(h^4)*((J-1)^3)*(imag(theta1(J+1))-imag(theta1(J-1)))/2;
70          A(2,2)=-1+1.5*h^3*(J-1)^2;
71          B(2,2)=2;
72          D(2,2)=-1-1.5*h^3*(J-1)^2;
73          B(2,1)=h^2*K(kk);
74
75          error=error+abs(G(1))+abs(G(2));
76
77      %Boundary condition at J=NJ
78      else
79          G(1)=conc(1,J)-0;      %BC at far away point is always 0
80          B(1,1)=-1;
81
82          G(2)=conc(2,J)-0;      %BC at far away point is always 0
83          B(2,2)=-1;
84
85          error=error+abs(G(1))+abs(G(2));
86       end;
87
88  band(J)
89  end;
90  conc=conc+C;          %C comes from band(J) program
91
92  if jcount>=jcountmax, break, end;
93  end;
94
95  theta2=complex(conc(1,:),conc(2,:));                                %Creating
        theta from conc(1,-) and conc(2,-) values
96  dtheta0=gamma(4/3)*(-theta0(3)+4*theta0(2)-3*theta0(1))/(2*h);
97  dtheta1=gamma(4/3)*(-theta1(3)+4*theta1(2)-3*theta1(1))/(2*h);      %dtheta
        accurate to h^2
98  dtheta2=gamma(4/3)*(-theta2(3)+4*theta2(2)-3*theta2(1))/(2*h);
99
100 Zd=-(1/dtheta0)*((dtheta1/dtheta0)^2-dtheta2/dtheta0);               %turning
        theta into Impedance
101 ZZd(kk)=Zd;
102
103 %str=sprintf('K = %.10e: Zd = %.10e + j%.10e', K(kk), real(Zd),imag(Zd)); disp
        (str);
```

```
%str=sprintf('dtheta2 = %g', dtheta2); disp(str);

str=sprintf('%.10e', K(kk)); disp(str);
str=sprintf('%.10e', real(Zd)); disp(str);
str=sprintf('%.10e', imag(Zd)); disp(str);

figure(1)
plot(x,conc(1,:),'-r')
hold on;
plot(x,conc(2,:),'-g')
axis([0 2 -.4 1]);
legend('theta_r','theta_i');
title('Dimensionless Concentration away from Electrode Surface');
xlabel('length');
ylabel('theta');


end

fname=sprintf('first_order_%i.txt',NJ);
fid = fopen(fname, 'wt');
fprintf(fid, '%14.6e; %14.6e \n', h,h^2);
for kk=1:length(K);
fprintf(fid, '%12.6e; %20.12e; %20.12e; \n', K(kk),real(ZZd(kk)),imag(ZZd(kk))
    );
end;
fclose(fid);

figure(2)
plot(real(ZZd),-imag(ZZd),'-ks');hold on; axis equal;
title('Nyquist plot');
xlabel('Real part of Impedance');
ylabel('Imaginary part of Impedance');

figure(3)
semilogx(K, -imag(ZZd),'--k'); hold on;
semilogx(K, real(ZZd),'--b');
legend ('-Imaginary','Real');
title('Impedance vs Frequency');
xlabel('Dimensionless Frequency');
ylabel('Impedance');
```

APPENDIX C
CODES FOR IMPINGING JET ELECTRODE

This appendix contains the different MatLab codes used to solve the convective-diffusion equation for a submerged impinging jet electrode in Chapter 3. The first MatLab code is to solve the infinite Schmidt convective-diffusion equation or the term 0 finite Schmidt convective-diffusion equation. The second Matlab code solve for term 1 of the finite Schmidt convective-diffusion equation. The last MatLab code solves for term 2 of the finite Schmidt convective-diffusion equation. The solutions to each code are imputed into equation (3–32) to obtain a solution for the finite Schmidt convective-diffusion equation.

Code C.1. Finite Schmidt Convection Diffusion Term 0 for an impinging jet electrode

```
%This problem solves for the first term of the convective-diffusion equation with a finite schmidt number
%d^2(theta0)/d(xi)^2+3*xi^2*d(theta0)/d(xi)-j*K*theta0=0

clc; close all;clear all;
format longE;
global A B C D G X Y N NJ

h=0.01;                         %Step-size
logK=-2:.05:4;                   %Frequency range 0.001 to 1000
K=10.^logK;
x=0:h:10;                        %Range of x
NJ=length(x);
ZZd=zeros(1,length(K));          %Place holder for values of diffusion impedance

c_0=1;              %Boundary condition for x=0
c_e=0;              %Boundary condition for x=end (NJ)
tol=0;              %Absolute tolerance

conc_real=zeros(length(x),length(K));
conc_imag=zeros(length(x),length(K));

%Define initial guess
conc(1,1:NJ)=0.5;           %Conc(1,-) is the real component of theta0
conc(2,1:NJ)=0.5;             %Conc(2,-) is the imaginary component of theta0
N=2;

for kk=1:length(K)

error=1;
jcount=0;
jcountmax=4;                %Number of iterations the program will allow to converge
while error>=tol;
X=zeros(N,N);
Y=X;

jcount=jcount+1;
for J=1:NJ;
    A=zeros(N,N);
    B=A;
    D=A;
    %Boundary condition at J=1
     if J==1;
         G(1)=conc(1,J)-c_0;      %Real equation, so BC at electrode surface is 1
         B(1,1)=-1;

         G(2)=conc(2,J)-0;        %Imaginary equation, so BC at electrode surface is 0
         B(2,2)=-1;

          error=abs(G(1))+abs(G(2));

    %Region between boundaries
     elseif (J<=(NJ-1));
```

```
53          G(1)=conc(1,J+1)-2*conc(1,J)+conc(1,J-1)+h^2*K(kk)*conc(2,J)+1.5*h^3*(
    J-1)^2*(conc(1,J+1)-conc(1,J-1));
54          A(1,1)=-1+1.5*h^3*(J-1)^2;
55          B(1,1)=2;
56          D(1,1)=-1-1.5*h^3*(J-1)^2;
57          B(1,2)=-h^2*K(kk);
58
59          G(2)=conc(2,J+1)-2*conc(2,J)+conc(2,J-1)-h^2*K(kk)*conc(1,J)+1.5*h^3*(
    J-1)^2*(conc(2,J+1)-conc(2,J-1));
60          A(2,2)=-1+1.5*h^3*(J-1)^2;
61          B(2,2)=2;
62          D(2,2)=-1-1.5*h^3*(J-1)^2;
63          B(2,1)=h^2*K(kk);
64
65          error=error+abs(G(1))+abs(G(2));
66
67      %Boundary condition at J=NJ
68      else
69          G(1)=conc(1,J)-0;      %BC at far away point is always 0
70          B(1,1)=-1;
71
72          G(2)=conc(2,J)-0;      %BC at far away point is always 0
73          B(2,2)=-1;
74
75          error=error+abs(G(1))+abs(G(2));
76      end;
77
78 band(J)
79 end;
80 conc=conc+C;          %C comes from band(J) program
81
82 if jcount>=jcountmax, break, end;
83 end;
84
85 theta0=complex(conc(1,:),conc(2,:));                                    %Creating
    theta from conc(1,-) and conc(2,-) values
86 dtheta0=gamma(4/3)*(-theta0(3)+4*theta0(2)-3*theta0(1))/(2*h);      %dtheta
    accurate to h^2
87
88 Zd=-1/dtheta0;                %turning theta into Impedance
89
90 % str=sprintf('K = %g: Zd = %.10e + j%.10e', K(kk), real(Zd),imag(Zd)); disp(
    str);
91 % str=sprintf('dtheta0 = %g', dtheta0); disp(str);
92
93 str=sprintf('%.10e', K(kk)); disp(str);
94 str=sprintf('%.10e', real(Zd)); disp(str);
95 str=sprintf('%.10e', imag(Zd)); disp(str);
96
97 figure(1)
98 plot(x,conc(1,:),'-r')
99 hold on;
100 plot(x,conc(2,:),'-g')
101 axis([0 2 -.4 1]);
102 legend('theta_r','theta_i');
103 title('Dimensionless Concentration away from Electrode Surface');
104 xlabel('length');
105 ylabel('theta');
```

```
106
107 for kkk=1:length(x)
108     conc_real(kkk,kk)=(conc(1,kkk));
109     conc_imag(kkk,kk)=(conc(2,kkk));
110 end
111
112 ZZd(kk)=Zd;
113
114 Ktemp=K(kk);
115 fname=sprintf('theta0_%i',kk);
116 save(fname,'theta0','Ktemp','h');
117 end
118
119 fname=sprintf('zero_order_%i.txt',NJ);
120 fid = fopen(fname, 'wt');
121 fprintf(fid, '%14.6e; %14.6e \n', h,h^2);
122 for kk=1:length(K);
123 fprintf(fid, '%12.6e; %20.12e; %20.12e; \n', K(kk),real(ZZd(kk)),imag(ZZd(kk))
    );
124 end;
125 fclose(fid);
126
127 figure(2)
128 plot(real(ZZd),-imag(ZZd),'-ks');hold on; axis equal;
129 Zfilm=tanh(sqrt(j*K))./sqrt(j*K);
130 plot(real(Zfilm),-imag(Zfilm),'--b');
131 legend('MatLab Data','Finite Film Thickness tanh(sqrt(j*K))/sqrt(j*K)');
132 title('Nyquist plot');
133 xlabel('Real part of Impedance');
134 ylabel('Imaginary part of Impedance');
135
136 str=sprintf('%.10e', real(Zfilm)); disp(str);
137
138 figure(3)
139 loglog(K, -imag(ZZd),'--k'); hold on;
140 loglog(K, real(ZZd),'--b');
141 legend ('-Imaginary','Real');
142 title('Impedance vs Frequency');
143 xlabel('Dimensionless Frequency');
144 ylabel('Impedance');
145
146 realZfilm=real(Zfilm);
147 imagZfilm=imag(Zfilm);
148
149 realZZd=real(ZZd)';
150 imagZZd=imag(ZZd)';
```

Code C.2. Finite Schmidt Convection Diffusion Term 1 for an impinging jet electrode

```
%This problem solves for the second term of the convective-diffusion equation with a finite schmidt number

clc; close all;clear all;
format longE;
global A B C D G X Y N NJ

h=0.01;                          %Step-size
logK=-2:.05:4;                    %Frequency range 0.001 to 1000
K=10.^logK;
x=0:h:10;                         %Range of x
NJ=length(x);
ZZd=zeros(1,length(K));           %Place holder for values of diffusion impedance
aa=0.51023;
bb=-0.61592;
cc=(3/(1.352)^4)^(1/3);

c_0=1;              %Boundary condition for x=0
c_e=0;              %Boundary condition for x=end (NJ)
tol=0;              %Absolute tolerance

conc_real=zeros(length(x),length(K));
conc_imag=zeros(length(x),length(K));

%Define initial guess
conc(1,1:NJ)=0.5;           %Conc(1,-) is the real component of theta0
conc(2,1:NJ)=0.5;             %Conc(2,-) is the imaginary component of theta0
N=2;

for kk=1:length(K)
fname=sprintf('theta0_%i',kk);
load(fname);
flag=0;if(abs(Ktemp-K(kk))~=0);flag=3;quit;end;

error=1;
jcount=0;
jcountmax=4;                %Number of iterations the program will allow to converge
while error>=tol;
X=zeros(N,N);
Y=X;

jcount=jcount+1;
for J=1:NJ;
    A=zeros(N,N);
    B=A;
    D=A;
    %Boundary condition at J=1
     if J==1;
         G(1)=conc(1,J);      %Real equation, so BC at electrode surface is 0
         B(1,1)=-1;

         G(2)=conc(2,J);         %Imaginary equation, so BC at electrode surface is 0
         B(2,2)=-1;

```

```
54          error=abs(G(1))+abs(G(2));
55
56      %Region between boundaries
57      elseif (J<=(NJ-1));
58          G(1)=conc(1,J+1)-2*conc(1,J)+conc(1,J-1)+h^2*K(kk)*conc(2,J)+1.5*h^3*(
    J-1)^2*(conc(1,J+1)-conc(1,J-1))...
59              -cc*(h^4)*((J-1)^3)*(real(theta0(J+1))-real(theta0(J-1)))/2;
60          A(1,1)=-1+1.5*h^3*(J-1)^2;
61          B(1,1)=2;
62          D(1,1)=-1-1.5*h^3*(J-1)^2;
63          B(1,2)=-h^2*K(kk);
64
65          G(2)=conc(2,J+1)-2*conc(2,J)+conc(2,J-1)-h^2*K(kk)*conc(1,J)+1.5*h^3*(
    J-1)^2*(conc(2,J+1)-conc(2,J-1))...
66              -cc*(h^4)*((J-1)^3)*(imag(theta0(J+1))-imag(theta0(J-1)))/2;
67          A(2,2)=-1+1.5*h^3*(J-1)^2;
68          B(2,2)=2;
69          D(2,2)=-1-1.5*h^3*(J-1)^2;
70          B(2,1)=h^2*K(kk);
71
72          error=error+abs(G(1))+abs(G(2));
73
74      %Boundary condition at J=NJ
75      else
76          G(1)=conc(1,J)-0;      %BC at far away point is always 0
77          B(1,1)=-1;
78
79          G(2)=conc(2,J)-0;      %BC at far away point is always 0
80          B(2,2)=-1;
81
82          error=error+abs(G(1))+abs(G(2));
83       end;
84
85  band(J)
86  end;
87  conc=conc+C;          %C comes from band(J) program
88
89  if jcount>=jcountmax, break, end;
90  end;
91
92  theta1=complex(conc(1,:),conc(2,:));                                %Creating
        theta from conc(1,-) and conc(2,-) values
93  dtheta1=gamma(4/3)*(-theta1(3)+4*theta1(2)-3*theta1(1))/(2*h);      %dtheta
        accurate to h^2
94  dtheta0=gamma(4/3)*(-theta0(3)+4*theta0(2)-3*theta0(1))/(2*h);
95
96  Zd=dtheta1/(dtheta0^2);                %turning theta into Impedance
97
98  %str=sprintf('K = %.10e: Zd = %.10e + j%.10e', K(kk), real(Zd),imag(Zd)); disp
        (str);
99  %str=sprintf('dtheta1 = %g', dtheta1); disp(str);
100 str=sprintf('%.10e', K(kk)); disp(str);
101 str=sprintf('%.10e', real(Zd)); disp(str);
102 str=sprintf('%.10e', imag(Zd)); disp(str);
103
104 figure(1)
105 plot(x,conc(1,:),'-r')
106 hold on;
```

```
plot(x,conc(2,:),'-g')
axis([0 2 -.4 1]);
legend('theta_r','theta_i');
title('Dimensionless Concentration away from Electrode Surface');
xlabel('length');
ylabel('theta');

for kkk=1:length(x)
    conc_real(kkk,kk)=(conc(1,kkk));
    conc_imag(kkk,kk)=(conc(2,kkk));
end

ZZd(kk)=Zd;
Ktemp=K(kk);
fname=sprintf('theta1_%i',kk);
save(fname,'theta1','Ktemp','h');
end

fname=sprintf('first_order_%i.txt',NJ);
fid = fopen(fname, 'wt');
fprintf(fid, '%14.6e; %14.6e \n', h,h^2);
for kk=1:length(K);
fprintf(fid, '%12.6e; %20.12e; %20.12e; \n', K(kk),real(ZZd(kk)),imag(ZZd(kk))
    );
end;
fclose(fid);

figure(2)
plot(real(ZZd),-imag(ZZd),'-ks');hold on; axis equal;
title('Nyquist plot');
xlabel('Real part of Impedance');
ylabel('Imaginary part of Impedance');

figure(3)
semilogx(K, -imag(ZZd),'--k'); hold on;
semilogx(K, real(ZZd),'--b');
legend ('-Imaginary','Real');
title('Impedance vs Frequency');
xlabel('Dimensionless Frequency');
ylabel('Impedance');

realZZd=real(ZZd)';
imagZZd=imag(ZZd)';
```

Code C.3. Finite Schmidt Convection Diffusion Term 2 for an impinging jet electrode

```
%This problem solves for the third term of the convective-diffusion equation
    with a finite schmidt number

clc; close all;clear all;
format longE;
global A B C D G X Y N NJ

h=0.01;                          %Step-size
logK=-2:.05:4;                    %Frequency range 0.001 to 1000
K=10.^logK;
x=0:h:10;                         %Range of x
NJ=length(x);
ZZd=zeros(1,length(K));          %Place holder for values of diffusion impedance
aa=0.51023;
bb=-0.61592;
cc=(3/(1.352)^4)^(1/3);

c_0=1;              %Boundary condition for x=0
c_e=0;              %Boundary condition for x=end (NJ)
tol=0;              %Absolute tolerance

conc_real=zeros(length(x),length(K));
conc_imag=zeros(length(x),length(K));


%Define initial guess
conc(1,1:NJ)=0.5;           %Conc(1,-) is the real component of theta0
conc(2,1:NJ)=0.5;             %Conc(2,-) is the imaginary component of theta0
N=2;

for kk=1:length(K)
fname=sprintf('theta0_%i',kk); % load solution for Theta-0
load(fname);
flag=0;if(abs(Ktemp-K(kk))~=0);flag=3;quit;end;
fname=sprintf('theta1_%i',kk); % load solution for Theta-1
load(fname);
flag=0;if(abs(Ktemp-K(kk))~=0);flag=3;quit;end;

error=1;
jcount=0;
jcountmax=4;                %Number of iterations the program will allow to
    converge
while error>=tol;
X=zeros(N,N);
Y=X;

jcount=jcount+1;
for J=1:NJ;
    A=zeros(N,N);
    B=A;
    D=A;
    %Boundary condition at J=1
    if J==1;
        G(1)=conc(1,J);      %Real equation, so BC at electrode surface is 0
        B(1,1)=-1;

```

```
55          G(2)=conc(2,J);         %Imaginary equation, so BC at electrode surface
    is 0
56          B(2,2)=-1;
57
58          error=abs(G(1))+abs(G(2));
59
60      %Region between boundaries
61      elseif (J<=(NJ-1));
62          G(1)=conc(1,J+1)-2*conc(1,J)+conc(1,J-1)+h^2*K(kk)*conc(2,J)+1.5*h^3*(
    J-1)^2*(conc(1,J+1)-conc(1,J-1))...
63              -cc*(h^4)*((J-1)^3)*(real(theta1(J+1))-real(theta1(J-1)))/2;
64          A(1,1)=-1+1.5*h^3*(J-1)^2;
65          B(1,1)=2;
66          D(1,1)=-1-1.5*h^3*(J-1)^2;
67          B(1,2)=-h^2*K(kk);
68
69          G(2)=conc(2,J+1)-2*conc(2,J)+conc(2,J-1)-h^2*K(kk)*conc(1,J)+1.5*h^3*(
    J-1)^2*(conc(2,J+1)-conc(2,J-1))...
70              -cc*(h^4)*((J-1)^3)*(imag(theta1(J+1))-imag(theta1(J-1)))/2;
71          A(2,2)=-1+1.5*h^3*(J-1)^2;
72          B(2,2)=2;
73          D(2,2)=-1-1.5*h^3*(J-1)^2;
74          B(2,1)=h^2*K(kk);
75
76          error=error+abs(G(1))+abs(G(2));
77
78      %Boundary condition at J=NJ
79      else
80          G(1)=conc(1,J)-0;       %BC at far away point is always 0
81          B(1,1)=-1;
82
83          G(2)=conc(2,J)-0;       %BC at far away point is always 0
84          B(2,2)=-1;
85
86          error=error+abs(G(1))+abs(G(2));
87       end;
88
89  band(J)
90  end;
91  conc=conc+C;            %C comes from band(J) program
92
93  if jcount>=jcountmax, break, end;
94  end;
95
96  theta2=complex(conc(1,:),conc(2,:));                                %Creating
        theta from conc(1,-) and conc(2,-) values
97  dtheta0=gamma(4/3)*(-theta0(3)+4*theta0(2)-3*theta0(1))/(2*h);
98  dtheta1=gamma(4/3)*(-theta1(3)+4*theta1(2)-3*theta1(1))/(2*h);      %dtheta
        accurate to h^2
99  dtheta2=gamma(4/3)*(-theta2(3)+4*theta2(2)-3*theta2(1))/(2*h);
100
101 Zd=-(1/dtheta0)*((dtheta1/dtheta0)^2-dtheta2/dtheta0);              %turning
        theta into Impedance
102 ZZd(kk)=Zd;
103
104 str=sprintf('%.10e', K(kk)); disp(str);
105 str=sprintf('%.10e', real(Zd)); disp(str);
106 str=sprintf('%.10e', imag(Zd)); disp(str);
```

```
107
108 figure(1)
109 plot(x,conc(1,:),'-r')
110 hold on;
111 plot(x,conc(2,:),'-g')
112 axis([0 2 -.4 1]);
113 legend('theta_r','theta_i');
114 title('Dimensionless Concentration away from Electrode Surface');
115 xlabel('length');
116 ylabel('theta');
117
118 for kkk=1:length(x)
119     conc_real(kkk,kk)=(conc(1,kkk));
120     conc_imag(kkk,kk)=(conc(2,kkk));
121 end
122
123 end
124
125 fname=sprintf('first_order_%i.txt',NJ);
126 fid = fopen(fname, 'wt');
127 fprintf(fid, '%14.6e; %14.6e \n', h,h^2);
128 for kk=1:length(K);
129 fprintf(fid, '%12.6e; %20.12e; %20.12e; \n', K(kk),real(ZZd(kk)),imag(ZZd(kk))
    );
130 end;
131 fclose(fid);
132
133 figure(2)
134 plot(real(ZZd),-imag(ZZd),'-ks');hold on; axis equal;
135 title('Nyquist plot');
136 xlabel('Real part of Impedance');
137 ylabel('Imaginary part of Impedance');
138
139 figure(3)
140 semilogx(K, -imag(ZZd),'--k'); hold on;
141 semilogx(K, real(ZZd),'--b');
142 legend ('-Imaginary','Real');
143 title('Impedance vs Frequency');
144 xlabel('Dimensionless Frequency');
145 ylabel('Impedance');
146
147 realZZd=real(ZZd)';
148 imagZZd=imag(ZZd)';
```

APPENDIX D
CODES FOR CONVECTIVE DIFFUSION IMPEDANCE WITH HOMOGENEOUS REACTION

This appendix contains the different FORTRAN codes that produced the results for the solution of the convective-diffusion equation with homogeneous reaction. This appendix also has the Matlab codes to plot the steady-state results, to plot a polarization curve from the steady state results and a code to create the impedance from the results of the oscillating FORTRAN code.

D.1 Input for Convective Diffusion Impedance with Homogeneous Reactions Code

The following codes are the input files for the convective diffusion equation with homogeneous reaction. The input files are broken into two files. The first input file has all the input parameters except the input potential. The input code has the number of species being solved, the total number of points, the number of points until the coupler, the distance of the reaction region in cm, the distance of the inner layer in cm. The input file also includes the rotation speed of the rotating disk, in RPM, and the kinematic viscosity of the solution, in cm^2/s. The rate constants in the input file are the equilibrium rate of the homogeneous reaction, in mol/cm^3, the backward rate of the homogeneous reaction, in $cm^3/mol \cdot s$, the rate constant for the heterogeneous reaction of the reacting species, and the tafel kinetics value for the heterogeneous reaction. The input file includes the error allowed for the BIG values, which is discussed in section 2.2.2. And the end of the input file has the specific values to describe each species in the system, including diffusion coefficients in cm^2/s, the charge, a character name, and the concentration value in the bulk in mol/cm^3. The second input file contain the potential, in volts, used in the heterogeneous reaction. The potential is in it's own input file to allow a polarization curve to be calculated easier.

```
4
12001
8001
0.004
0.024
2000.
0.01
1.E-6
1.E7
2.E-12
19.9
1.E-12
1.684E-5 0. AB 0.01
1.957E-5 -1. A- 0.0001
1.902E-5 +1. B+ 0.0001


C     line 1 is the number of species
C     line 2 is NJ
C     line 3 is KJ, number of points in the reaction
C     line 4 is the distance of the inner reaction layer in cm (5um)
C     line 5 is the distance of the rest of the inner layer in cm (10um)
C     line 6 is the rotation speed in rpm
C     line 7 is the kinematic viscosity, cm^2/s
C     line 8 is the equilibrium rate of rxn, mol/cm^3
C     line 9 is the backward rate of reaction, cm^3/(mol*s)
C     line 10 is the rate constant for the flux of the reacting species
C     line 11 is the tafel b value for the flux of the reacting species
C     line 12 is the error allowed for the BIGs
C     lines 13-15 specify values used to describe each species in the system
```

Code D.1. Input file for the Convective Diffusion with Homogeneous Reaction

```
-0.5
```

Code D.2. Potential input file for the Convective Diffusion with Homogeneous Reaction

D.2 Steady-State Convective Diffusion Impedance with Homogeneous Reactions Code

This section contains the steady-state FORTRAN codes used to solve the convective diffusion equation with a homogeneous reaction. The mathematical workup for these codes are in Chapter 4. The FORTRAN codes are followed by two Matlab codes. The first Matlab code takes the output from the steady-state FORTRAN code and plots the data. The second Matlab code creates a polarization curve by running the executable created from the FORTRAN code for different input potentials.

The first section in the code, called CONVDIFF, is the main program, which outlines the global variables and sets up calling files to save over as output files as well as calling the input files. Then the subroutines that are called in the main program are all shown. The subroutine VELOCITY creates the velocity profile for the code. It was necessary to create the velocity profile using the two different mesh sizes, schematically shown in 4-3. The VELOCITY subroutine also finds the exact value of the velocity at the half mesh points on either side of the coupler, at $J = KJ$. The subroutine BC1 solves the boundary condition at the electrode surface. The subroutines REACTION and INNER solve the non–linear coupled differential equations using mesh sizes HH and H, respectively. The subroutine COUPLER, sets the flux at KJ equal using two flux expression equations. The mathematics behind the coupler are discussed in chapter 4 section **??**. Finally the subroutine BCNJ solves the boundary condition in the bulk. BAND and MATINV, in A, are called in order to solve the steady-state solution.

Code D.3. Steady State Convective Diffusion with Homogeneous Reaction Main Program

```
C     Convective Diffusion Equation with Homogeneous Reaction
C     3 species system
C     SPECIES 1 = AB, SPECIES 2 = A-, SPECIES 3 = B+
C     Species 3 is the reacting species
C     This is the steady state solution only
C     It should be ran prior to cdh_os.for
C     The input file is the same for both

      PROGRAM CONVDIFF
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(4,4),B(4,4),C(4,100001),D(4,9),G(4),X(4,4),Y(4,4)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(3,100001),RXN(100001),DIFF(3),H,EBIG,IJ,KJ,HH
      COMMON/BCI/ VEL1, FLUX
      COMMON/VET/ VEL(100001), Y1, VELH12, VELHH12
      COMMON/VETT/ VELNEARH(100001),VELFARH(100001),FVELH(100001)
      COMMON/VEKK/ VELNEARHH(100001),VELFARHH(100001),FVELHH(100001)
      COMMON/EXTRA/ Z(6),REF(6)
      COMMON/BUL/ CBULK(6), JCOUNT
      COMMON/RTE/ rateb, equilib

      CHARACTER REF*6

  102 FORMAT (/30H THE NEXT RUN DID NOT CONVERGE)
  103 FORMAT ('Error=',E16.6/(1X,'Species=',A6,2X,'C at Electrode=',
     1 E12.5,2X,'C at Bulk=',E12.5))
  104 FORMAT (A6)
  105 FORMAT (E12.7)
  333 FORMAT (4x,'AB'12x,'A-',12x,'B+', 12x, 'RXN')
  300 FORMAT (20x,'G(1)'14x,'G(2)',14x,'G(3)', 14x, 'RXN')
  301 FORMAT (5x,'J=' I5,  8E18.9)
  334 FORMAT (4(E25.15,5X))
  302 FORMAT ('Iteration='I4)
  305 FORMAT (E20.12,3X,E20.12,3X,E20.12,3X,E20.12)

      OPEN(UNIT=13, FILE='cdh_out.txt')
      CLOSE(UNIT=13, STATUS='DELETE')
      OPEN(UNIT=13, FILE='cdh_out.txt')

      OPEN(UNIT=16, FILE='VEL.txt')
      CLOSE(UNIT=16, STATUS='DELETE')
      OPEN(UNIT=16, FILE='VEL.txt')

      OPEN(12,FILE='cdh_G_out.txt')
      CLOSE(12, STATUS='DELETE')
      OPEN(12, FILE='cdh_G_out.txt')
      WRITE(12,300)

      OPEN(UNIT=17, FILE='VEL12.txt')
      CLOSE(UNIT=17, STATUS='DELETE')
      OPEN(UNIT=17, FILE='VEL12.txt')

      OPEN(14,file='cdh_in.txt',status='old')
      READ(14,*) N,NJ,KJ,Y1,Y2,RPM,ANU,equilib,rateb,AKB,BB,EBIG
      READ(14,*) (DIFF(I),Z(I),REF(I),CBULK(I),I=1,(N-1))

```

```
57        OPEN(11,file='potential_in.txt',status='old')
58        READ(11,*) V
59
60 C      Constants
61        F=96487.
62        ROT=RPM*2*3.141592653589793/60
63        PRINT *, 'ROT=', ROT
64 C       H=(ANU/ROT)**(1./2.)*(3.*DIFF(3)/(0.51023*ANU))**(1./3.)
65 C       1     *1.00E-3
66
67        PRINT *, 'Y2=', Y2
68        H=Y2/(NJ-KJ)
69        PRINT *, 'H=', H
70
71        PRINT *, 'Y1=', Y1
72        HH=Y1/(KJ-1)
73        PRINT *, 'HH=', HH
74
75        DELTA=(ANU/ROT)**(1./2.)*(3.*DIFF(3)/(0.51023*ANU))**(1./3.)*5
76        PRINT *, 'DELTA=', DELTA
77
78        OPEN(15,FILE='cdh_ssvalues_out.txt')
79        CLOSE(15, STATUS='DELETE')
80        OPEN(15, FILE='cdh_ssvalues_out.txt')
81   337 FORMAT (I2/I7/I7/E15.8/E15.8/E15.8/E15.8/E15.8/E15.4/E15.4/E15.4)
82        write (15,337) N,NJ,KJ,H,HH,RPM, ANU,DIFF(3),AKB,BB,V
83
84 c      CALL THE SUBROUTINE TO CALCULATE THE VELOCITY
85        CALL VELOCITY (ROT, ANU)
86        OPEN(17,FILE='VEL12.txt')
87        CLOSE(17, STATUS='DELETE')
88        OPEN(17, FILE='VEL12.txt')
89   338 FORMAT (E20.13/E20.13)
90        write (17,338) VELH12, VELHH12
91
92 C      Create flux of the reacting species constants
93        FLUX=-AKB*exp(-BB*V)/F/Z(3)
94        PRINT *, 'FLUX=', FLUX
95
96        DO 21 J=1,NJ
97        DO 21 I=1,N-1
98        C(I,J)=0.0
99     21 CONC(I,J)=CBULK(I)
100       JCOUNT=0
101       TOL=1.E-15*N*NJ/1000
102       PRINT *, 'TOL=', TOL
103    22 JCOUNT=JCOUNT+1
104       AMP=0.0
105       J=0
106       DO 23 I=1,N
107       DO 23 K=1,N
108       Y(I,K)=0.0
109    23 X(I,K)=0.0
110    24 J=J+1
111       DO 25  I=1,N
112       G(I)=0.0
113       DO 25 K=1,N
114       A(I,K)=0.0
```

```
115         B(I,K)=0.0
116      25 D(I,K)=0.0
117
118         IF (J.EQ.1) CALL BC1(J)
119         IF (J.GT.1 .AND. J.LT.KJ) CALL REACTION(J)
120         IF (J.EQ.KJ) CALL COUPLER(J)
121         IF (J.GT.KJ .AND. J.LT.NJ) CALL INNER(J)
122         IF (J.EQ.NJ)  CALL BCNJ(J)
123         CALL BAND(J)
124
125         AMP=AMP+DABS(G(1))+DABS(G(2))+DABS(G(3))+DABS(G(4))
126
127         IF (J.LT.NJ)  GO TO 24
128
129         PRINT *, 'ERROR=', AMP
130
131         DO 16 K=1,NJ
132         RXN(K)=RXN(K)+C(4,K)
133         DO 16 I=1,N-1
134         IF (C(I,K).LT.-0.999*CONC(I,K)) C(I,K)=-0.999*CONC(I,K)
135         IF (C(I,K).GT. 999.*CONC(I,K)) C(I,K)= 999.*CONC(I,K)
136         CONC(I,K)=CONC(I,K)+C(I,K)
137      16 CONTINUE
138
139         WRITE(12,302) (JCOUNT)
140
141 c       If the error is less then the tolerance, finish program
142         IF (DABS(AMP).LT.DABS(TOL)) GO TO  15
143
144 c       If the error is greater then tolerance, do another iteration
145      33 IF (JCOUNT.LE.19) GO TO 22
146         print 102
147
148      15 PRINT 103, AMP,(REF(I),CONC(I,1),CONC(I,NJ),I=1,N-1)
149
150         PRINT *,'JCOUNT=',JCOUNT
151
152         DO 19 J=1,NJ
153         BIG=RXN(J)
154         BIG2=1.0E-40
155      19 IF (ABS(BIG).LE.BIG2) RXN(J)=0.0
156
157         DO 18 J=1,NJ
158         DO 18 I=1,N-1
159         BIG=CONC(I,J)
160         BIG2=1.0E-40
161      18 IF (ABS(BIG).LE.BIG2) CONC(I,J)=0.0
162
163         WRITE(13,334) (CONC(1,J),CONC(2,J),CONC(3,J),RXN(J),J=1,NJ)
164
165         END PROGRAM CONVDIFF
```

Code D.4. Steady-State Convective Diffusion with Homogeneous Reaction Subroutine to Create the Velocity Profile

```
      SUBROUTINE VELOCITY (ROT, ANU)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(3,100001),RXN(100001),DIFF(3),H,EBIG,IJ,KJ,HH
      COMMON/VET/ VEL(100001), Y1, VELH12, VELHH12
      COMMON/VETT/ VELNEARH(100001),VELFARH(100001),FVELH(100001)
      COMMON/VEKK/ VELNEARHH(100001),VELFARHH(100001),FVELHH(100001)
      COMMON/BCI/ VEL1, FLUX

  305 FORMAT (E20.12,3X,E20.12,3X,E20.12,3X,E20.12)

C     Create a term for (rot/nu)^(1/2)
      ROTNU=(ROT/ANU)**(1./2.)
      PRINT *, 'ROTNU=', ROTNU
c     Create term for 2*A/c, in term 2 of expansion
      FAR2=2*0.92486353/0.88447411
c     Create term for (A^2+B^2)/2*c^2, in term 3 of expansion
      FAR3=(0.92486353**2.+1.20221175**2.)/(2.*0.88447411**3.)
c     Create term for A(A^2+B^2)/6*c^5, in term 4 of expansion
      FAR4=0.92486353*(0.92486353**2.+1.20221175**2.)/
     1    (6.*0.88447411**5.)
C     Create a term for (rot*nu)^(1/2)
      ROTNU2=(ROT*ANU)**(1./2.)


C     CREATE VELOCITY PROFILE
      DO 221 I=1,KJ
      VELNEARHH(I)=-ROT**(3./2.)*0.510232618867*(HH*(I-1))**2/ANU**(.5)
     1            +((1./3.)*(HH*(I-1))**3.*ROT**2/ANU)
     2     +((-0.615922014399/6.)*(HH*(I-1))**4.*ROT**(2.5)/(ANU**(1.5)))
      VELFARHH(I)=-ROTNU2*0.88447411
     1        +ROTNU2*FAR2*EXP(-0.88447411*(HH*(I-1))*ROTNU)
     2        -ROTNU2*FAR3*EXP(-2*0.88447411*(HH*(I-1))*ROTNU)
     3        +ROTNU2*FAR4*EXP(-3*0.88447411*(HH*(I-1))*ROTNU)
      FVELHH(I)=1/(1+(EXP(-20*ROTNU*(((HH*(I-1))-1.25*(1/ROTNU))))))
      VEL(I)=((1-FVELHH(I))*VELNEARHH(I))+(FVELHH(I)*VELFARHH(I))
  221 CONTINUE

      VEL(1)=0.
      VEL1=(0.5)*VEL(2)
C      PRINT *, 'VEL1=', VEL1
      WRITE(16,305) (VELNEARHH(I),VELFARHH(I),FVELHH(I),VEL(I),I=1,KJ)

C     CREATE VELOCITY PROFILE
      DO 222 I=1,(NJ-KJ)+1
      VELNEARH(I)=-ROT**(3./2.)*0.510232618867*(Y1+(H*(I)))**2/ANU**(.5)
     1            +((1./3.)*(Y1+(H*(I)))**3.*ROT**2/ANU)
     2  +((-0.615922014399/6.)*(Y1+(H*(I)))**4.*ROT**(2.5)/(ANU**(1.5)))
      VELFARH(I)=-ROTNU2*0.88447411
     1        +ROTNU2*FAR2*EXP(-0.88447411*(Y1+(H*(I)))*ROTNU)
     2        -ROTNU2*FAR3*EXP(-2*0.88447411*(Y1+(H*(I)))*ROTNU)
     3        +ROTNU2*FAR4*EXP(-3*0.88447411*(Y1+(H*(I)))*ROTNU)
      FVELH(I)=1/(1+(EXP(-20*ROTNU*(((Y1+(H*(I)))-1.25*(1/ROTNU))))))
      VEL(KJ+I)=((1-FVELH(I))*VELNEARH(I))+(FVELH(I)*VELFARH(I))
```

```
 222  CONTINUE

      WRITE(16,305) (VELNEARH(I-KJ),VELFARH(I-KJ),FVELH(I-KJ),
     1       VEL(I),I=KJ+1,NJ)

      VELNEARHH12=-ROT**(3./2.)*0.5102326188867*(HH*(FLOAT(KJ)-1.5))**2
     1          /ANU**(.5)
     1              +((1./3.)*(HH*(FLOAT(KJ)-1.5))**3.*ROT**2/ANU)
     2      +((-0.6159220143999/6.)*(HH*(FLOAT(KJ)-1.5))**4.*ROT**(2.5)/
     3       (ANU**(1.5)))
      VELFARHH12=-ROTNU2*0.88447411
     1         +ROTNU2*FAR2*EXP(-0.88447411*(HH*(FLOAT(KJ)-1.5))*ROTNU)
     2         -ROTNU2*FAR3*EXP(-2*0.88447411*(HH*(FLOAT(KJ)-1.5))*ROTNU)
     3         +ROTNU2*FAR4*EXP(-3*0.88447411*(HH*(FLOAT(KJ)-1.5))*ROTNU)
      FVELHH12=1./(1.+(EXP(-20.*ROTNU*(((HH*(FLOAT(KJ)-1.5))-
     1          1.25*(1/ROTNU))))))
      VELHH12=((1.-FVELHH12)*VELNEARHH12)+(FVELHH12*VELFARHH12)
      VELHH122=(VEL(KJ)+VEL(KJ-1))/2.
      PRINT *, 'VELHH122=', VELHH122
      PRINT *, 'HH*(FLOAT(KJ)-1.5', HH*(FLOAT(KJ)-1.5)

      VELNEARH12=-ROT**(3./2.)*.5102326188867*(Y1+(H*(0.5)))**2/ANU**(.5)
     1              +((1./3.)*(Y1+(H*(0.5)))**3.*ROT**2/ANU)
     2       +((-0.6159220143999/6.)*(Y1+(H*(0.5)))**4.*ROT**(2.5)/
     3       (ANU**(1.5)))
      VELFARH12=-ROTNU2*0.88447411
     1         +ROTNU2*FAR2*EXP(-0.88447411*(Y1+(H*(0.5)))*ROTNU)
     2         -ROTNU2*FAR3*EXP(-2*0.88447411*(Y1+(H*(0.5)))*ROTNU)
     3         +ROTNU2*FAR4*EXP(-3*0.88447411*(Y1+(H*(0.5)))*ROTNU)
      FVELH12=1/(1+(EXP(-20*ROTNU*(((Y1+(H*(0.5)))-1.25*(1/ROTNU))))))
      VELH12=((1-FVELH12)*VELNEARH12)+(FVELH12*VELFARH12)
      VELH122=(VEL(KJ)+VEL(KJ+1))/2.
      PRINT *, 'VELH122=', VELH122
      PRINT *, 'Y1+(H*(0.5)', Y1+(H*(0.5))

      PRINT *, 'VELKJ=', VEL(KJ)

      RETURN
      END
```

Code D.5. Steady-State Convective Diffusion with Homogeneous Reaction Subroutine for the Electrode Boundary Condition

```
1     SUBROUTINE BC1(J)
2       IMPLICIT DOUBLE PRECISION (A-H, O-Z)
3       COMMON/BAB/ A(4,4),B(4,4),C(4,100001),D(4,9),G(4),X(4,4),Y(4,4)
4       COMMON/NSN/ N, NJ
5       COMMON/VAR/ CONC(3,100001),RXN(100001),DIFF(3),H,EBIG,IJ,KJ,HH
6       COMMON/BCI/ VEL1, FLUX
7       COMMON/RTE/ rateb, equilib
8
9    301 FORMAT (5x,'J=' I5,  8E18.9)
10
11 C     For AB  non reacting species
12      G(1)=2.*DIFF(1)*(CONC(1,J+1)-CONC(1,J))/HH**2.
13     1    -VEL1/HH*(CONC(1,J+1)-CONC(1,J))
14     2     -(3.*RXN(J)+RXN(J+1))/4.
15      B(1,1)=2.*DIFF(1)/HH**2.-VEL1/HH
16      D(1,1)=-2.*DIFF(1)/HH**2.+VEL1/HH
17      B(1,4)=+0.75
18      D(1,4)=+0.25
19
20      BIG=ABS(2.*DIFF(1)*(CONC(1,J+1))/HH**2.)
21      BIG2=ABS(2.*DIFF(1)*CONC(1,J)/HH**2.)
22      IF (BIG2.GT.BIG) BIG=BIG2
23      BIG3=ABS(VEL1/HH*CONC(1,J+1))
24      IF (BIG3.GT.BIG) BIG=BIG3
25      BIG4=ABS(VEL1/HH*CONC(1,J))
26      IF (BIG4.GT.BIG) BIG=BIG4
27      IF (ABS(-3.*RXN(J)/4.).GT.BIG) BIG=ABS(-3.*RXN(J)/4.)
28      IF (ABS(-RXN(J+1)/4.).GT.BIG) BIG=ABS(-RXN(J+1)/4.)
29      IF (ABS(G(1)).LT.BIG*EBIG) G(1)=0
30
31 C     For A+  non reacting species
32      G(2)=2.*DIFF(2)*(CONC(2,J+1)-CONC(2,J))/HH**2.
33     1    -VEL1/HH*(CONC(2,J+1)-CONC(2,J))
34     2     +(3.*RXN(J)+RXN(J+1))/4.
35      B(2,2)=2.*DIFF(2)/HH**2.-VEL1/HH
36      D(2,2)=-2.*DIFF(2)/HH**2.+VEL1/HH
37      B(2,4)=-0.75
38      D(2,4)=-0.25
39
40      BIG=ABS(2.*DIFF(2)*(CONC(2,J+1))/HH**2.)
41      BIG2=ABS(2.*DIFF(2)*CONC(2,J)/HH**2.)
42      IF (BIG2.GT.BIG) BIG=BIG2
43      BIG3=ABS(VEL1/HH*CONC(2,J+1))
44      IF (BIG3.GT.BIG) BIG=BIG3
45      BIG4=ABS(VEL1/HH*CONC(2,J))
46      IF (BIG4.GT.BIG) BIG=BIG4
47      IF (ABS(-3.*RXN(J)/4.).GT.BIG) BIG=ABS(-3.*RXN(J)/4.)
48      IF (ABS(-RXN(J+1)/4.).GT.BIG) BIG=ABS(-RXN(J+1)/4.)
49      IF (ABS(G(2)).LT.BIG*EBIG) G(2)=0
50
51 C     For B- reacting species
52      G(3)=2.*DIFF(3)*(CONC(3,J+1)-CONC(3,J))/HH**2.
53     1     -VEL1/HH*(CONC(3,J+1)-CONC(3,J))
54     2      +FLUX*CONC(3,J)/(HH/2.)
```

```
55        3      +(3.*RXN(J)+RXN(J+1))/4.
56         B(3,3)=2.*DIFF(3)/HH**2.-VEL1/HH-FLUX/(HH/2.)
57         D(3,3)=-2.*DIFF(3)/HH**2.+VEL1/HH
58         B(3,4)=-0.75
59         D(3,4)=-0.25
60
61         BIG=ABS(2.*DIFF(3)*(CONC(3,J+1))/HH**2.)
62         BIG2=ABS(2.*DIFF(3)*CONC(3,J)/HH**2.)
63         IF (BIG2.GT.BIG) BIG=BIG2
64         BIG3=ABS(VEL1/HH*CONC(3,J+1))
65         IF (BIG3.GT.BIG) BIG=BIG3
66         BIG4=ABS(VEL1/HH*CONC(3,J))
67         IF (BIG4.GT.BIG) BIG=BIG4
68         BIG5=ABS(-FLUX*CONC(3,J)/(HH/2.))
69         IF (BIG5.GT.BIG) BIG=BIG5
70         IF (ABS(-3.*RXN(J)/4.).GT.BIG) BIG=ABS(-3.*RXN(J)/4.)
71         IF (ABS(-RXN(J+1)/4.).GT.BIG) BIG=ABS(-RXN(J+1)/4.)
72         IF (ABS(G(3)).LT.BIG*EBIG) G(3)=0
73
74 C       For Reaction term
75         IF (rateb.EQ.0) GO TO 214
76         EPS=RXN(J)/(rateb*CONC(2,J)*CONC(3,J))
77         IF (ABS(EPS).GT.0.2) GO TO 214
78
79         REM=LOG(1.+EPS)
80       IF (ABS(REM).LT.1.0E-09)REM=EPS*(1.-EPS*(.5-EPS*(1./3.-EPS/4.)))
81
82         G(4)=LOG(equilib*CONC(1,J)/CONC(2,J)/CONC(3,J))-REM
83         B(4,1)=-1./CONC(1,J)
84         B(4,2)=-((RXN(J)/(rateb*CONC(3,J)*CONC(2,J)**2.))*
85        1       (1./(1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J)))))+1./CONC(2,J)
86         B(4,3)=-((RXN(J)/(rateb*CONC(2,J)*CONC(3,J)**2.))*
87        1       (1./(1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J)))))+1./CONC(3,J)
88         B(4,4)=1./(rateb*CONC(3,J)*CONC(2,J)*
89        1      (1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J))))
90
91         BIG=ABS(REM)
92         BIG2=ABS(LOG(equilib*CONC(1,J)/CONC(2,J)/CONC(3,J)))
93         IF (BIG2.GT.BIG) BIG=BIG2
94         IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0
95         GO TO 212
96
97   214 G(4)=-RXN(J)+rateb*(equilib*CONC(1,J)-CONC(2,J)*CONC(3,J))
98         B(4,1)=-rateb*equilib
99         B(4,2)=rateb*CONC(3,J)
100        B(4,3)=rateb*CONC(2,J)
101        B(4,4)=1.
102
103        BIG=ABS(rateb*equilib*CONC(1,J))
104        BIG2=ABS(rateb*CONC(2,J)*CONC(3,J))
105        IF (BIG2.GT.BIG) BIG=BIG2
106        BIG3=ABS(rateb*equilib*CONC(1,J))
107        IF (BIG3.GT.BIG) BIG=BIG3
108        IF (ABS(RXN(J)).GT.BIG) BIG=ABS(RXN(J))
109        IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0
110        CONTINUE
111
112   212  WRITE(12,301) J, (G(K),K=1,N)
```

```
113         RETURN
114         END
```

Code D.6. Steady State Convective Diffusion with Homogeneous Reaction Subroutine for the Reaction Region

```
      SUBROUTINE REACTION(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(4,4),B(4,4),C(4,100001),D(4,9),G(4),X(4,4),Y(4,4)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(3,100001),RXN(100001),DIFF(3),H,EBIG,IJ,KJ,HH
      COMMON/RTE/ rateb, equilib
      COMMON/VET/ VEL(100001), Y1, VELH12, VELHH12
  301 FORMAT (5x,'J=' I5,  8E18.9)

C     For AB
      G(1)=DIFF(1)*(CONC(1,J+1)-2.*CONC(1,J)+CONC(1,J-1))/HH**2.
     1     -VEL(J)*(CONC(1,J+1)-CONC(1,J-1))/(2.*HH)-RXN(J)
      B(1,1)=2.*DIFF(1)/HH**2.
      D(1,1)=-DIFF(1)/HH**2.+VEL(J)/(2.*HH)
      A(1,1)=-DIFF(1)/HH**2.-VEL(J)/(2.*HH)
      B(1,4)=+1.
      BIG=ABS(DIFF(1)*CONC(1,J+1)/HH**2.)
      BIG2=ABS(-1.*DIFF(1)*2.*CONC(1,J)/HH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(DIFF(1)*CONC(1,J-1)/HH**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      BIG4=ABS(-1.*VEL(J)*CONC(1,J+1)/(2.*HH))
      IF (BIG4.GT.BIG) BIG=BIG4
      BIG5=ABS(VEL(J)*CONC(1,J-1)/(2.*HH))
      IF (BIG5.GT.BIG) BIG=BIG5
      IF (ABS(RXN(J)).GT.BIG) BIG=ABS(RXN(J))
      IF (ABS(G(1)).LT.BIG*EBIG) G(1)=0

C     For A-
      G(2)=DIFF(2)*(CONC(2,J+1)-2.*CONC(2,J)+CONC(2,J-1))/HH**2.
     1     -VEL(J)*(CONC(2,J+1)-CONC(2,J-1))/(2.*HH)+RXN(J)
      B(2,2)=2.*DIFF(2)/HH**2.
      D(2,2)=-DIFF(2)/HH**2.+VEL(J)/(2.*HH)
      A(2,2)=-DIFF(2)/HH**2.-VEL(J)/(2.*HH)
      B(2,4)=-1.
      BIG=ABS(DIFF(2)*CONC(2,J+1)/HH**2.)
      BIG2=ABS(-1.*DIFF(2)*2.*CONC(2,J)/HH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(DIFF(2)*CONC(2,J-1)/HH**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      BIG4=ABS(-1.*VEL(J)*CONC(2,J+1)/(2.*HH))
      IF (BIG4.GT.BIG) BIG=BIG4
      BIG5=ABS(VEL(J)*CONC(2,J-1)/(2.*HH))
      IF (BIG5.GT.BIG) BIG=BIG5
      IF (ABS(RXN(J)).GT.BIG) BIG=ABS(RXN(J))
      IF (ABS(G(2)).LT.BIG*EBIG) G(2)=0

C     For B- The reacting species
      G(3)=DIFF(3)*(CONC(3,J+1)-2.*CONC(3,J)+CONC(3,J-1))/HH**2.
     1     -VEL(J)*(CONC(3,J+1)-CONC(3,J-1))/(2.*HH)+RXN(J)
      B(3,3)=2*DIFF(3)/HH**2.
      D(3,3)=-DIFF(3)/HH**2.+VEL(J)/(2.*HH)
      A(3,3)=-DIFF(3)/HH**2.-VEL(J)/(2.*HH)
      B(3,4)=-1.
```

```
      BIG=ABS(DIFF(3)*CONC(3,J+1)/HH**2.)
      BIG2=ABS(-1.*DIFF(3)*2.*CONC(3,J)/HH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(DIFF(3)*CONC(3,J-1)/HH**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      BIG4=ABS(-1.*VEL(J)*CONC(3,J+1)/(2.*HH))
      IF (BIG4.GT.BIG) BIG=BIG4
      BIG5=ABS(VEL(J)*CONC(3,J-1)/(2.*HH))
      IF (BIG5.GT.BIG) BIG=BIG5
      IF (ABS(RXN(J)).GT.BIG) BIG=ABS(RXN(J))
      IF (ABS(G(3)).LT.BIG*EBIG) G(3)=0

C     For Reaction term
      IF (rateb.EQ.0) GO TO 210
      EPS=RXN(J)/(rateb*CONC(2,J)*CONC(3,J))
      IF (ABS(EPS).GT.0.2) GO TO 210
      REM=LOG(1.+EPS)
    IF (ABS(REM).LT.1.0E-09)REM=EPS*(1.-EPS*(.5-EPS*(1./3.-EPS/4.)))
      G(4)=LOG(equilib*CONC(1,J)/CONC(2,J)/CONC(3,J))-REM
      B(4,1)=-1./CONC(1,J)
      B(4,2)=-((RXN(J)/(rateb*CONC(3,J)*CONC(2,J)**2.))*
     1      (1./(1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J)))))+1./CONC(2,J)
      B(4,3)=-((RXN(J)/(rateb*CONC(2,J)*CONC(3,J)**2.))*
     1      (1./(1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J)))))+1./CONC(3,J)
      B(4,4)=1./(rateb*CONC(3,J)*CONC(2,J)*
     1     (1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J))))
      BIG=ABS(REM)
      BIG2=ABS(LOG(equilib*CONC(1,J)/CONC(2,J)/CONC(3,J)))
      IF (BIG2.GT.BIG) BIG=BIG2
      IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0
      GO TO 212

  210 G(4)=-RXN(J)+rateb*(equilib*CONC(1,J)-CONC(2,J)*CONC(3,J))
      B(4,1)=-rateb*equilib
      B(4,2)=rateb*CONC(3,J)
      B(4,3)=rateb*CONC(2,J)
      B(4,4)=1.
      BIG=ABS(rateb*equilib*CONC(1,J))
      BIG2=ABS(rateb*CONC(2,J)*CONC(3,J))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(rateb*equilib*CONC(1,J))
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(RXN(J)).GT.BIG) BIG=ABS(RXN(J))
      IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0
      CONTINUE

c     SAVE G OUT DATA
  212 DO 11 I=2,20
   11 If (I.EQ.J) WRITE(12,301) J, (G(K),K=1,N)
      DO 12 I=100,120
   12 If (I.EQ.J) WRITE(12,301) J, (G(K),K=1,N)
      IF (J.EQ.KJ/2) THEN
      WRITE(12,301) J, (G(K),K=1,N)
      END IF
      RETURN
      END
```

Code D.7. Steady State Convective Diffusion with Homogeneous Reaction Subroutine for the Coupler

```
      SUBROUTINE COUPLER(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(4,4),B(4,4),C(4,100001),D(4,9),G(4),X(4,4),Y(4,4)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(3,100001),RXN(100001),DIFF(3),H,EBIG,IJ,KJ,HH
      COMMON/RTE/ rateb, equilib
      COMMON/VET/ VEL(100001), Y1, VELH12, VELHH12

  301 FORMAT (5x,'J=' I5,  8E18.9)

      PRINT *, 'VELH12=', VELH12
      PRINT *, 'VELHH12=', VELHH12

      COEFF1H=DIFF(1)/(H)
      COEFF1HH=DIFF(1)/(HH)
      COEFF2H=DIFF(2)/(H)
      COEFF2HH=DIFF(2)/(HH)
      COEFF3H=DIFF(3)/(H)
      COEFF3HH=DIFF(3)/(HH)

      PRINT *, 'AB_H=', H
      PRINT *, 'AB_HH=', HH
      PRINT *, 'kj=', KJ

C     For AB
      G(1)=COEFF1H*(CONC(1,J+1)-CONC(1,J))
     1     -VELH12*(CONC(1,J+1)+CONC(1,J))/2.
     2     +(H/2.)*(CONC(1,J+1)+3.*CONC(1,J))/4.*(VELH12-VEL(J))/(H/2.)
     3     -(H/2.)*(RXN(J+1)+3.*RXN(J))/4.
     4     -COEFF1HH*(CONC(1,J)-CONC(1,J-1))
     5     +VELHH12*(CONC(1,J)+CONC(1,J-1))/2.
     6    +(HH/2.)*(CONC(1,J-1)+3.*CONC(1,J))/4.*(VEL(J)-VELHH12)/(HH/2.)
     7     -(HH/2.)*(RXN(J-1)+3.*RXN(J))/4.
      B(1,1)=COEFF1H+VELH12/2.-(H/2.)*3./4.*(VELH12-VEL(J))/(H/2.)
     1       +COEFF1HH-VELHH12/2.-(HH/2.)*3./4.*(VEL(J)-VELHH12)/(HH/2.)
      D(1,1)=-COEFF1H+VELH12/2.-(H/2.)/4.*(VELH12-VEL(J))/(H/2.)
      A(1,1)=-COEFF1HH-VELHH12/2.-(HH/2.)/4.*(VEL(J)-VELHH12)/(HH/2.)
      B(1,4)=(H/2.)*(3./4.)+(HH/2.)*(3./4.)
      D(1,4)=(H/2.)*(1./4.)
      A(1,4)=(HH/2.)*(1./4.)

      BIG=ABS(COEFF1H*CONC(1,J+1))
      BIG2=ABS(COEFF1H*CONC(1,J))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(-COEFF1HH*CONC(1,J))
      IF (BIG3.GT.BIG) BIG=BIG3
      BIG4=ABS(-COEFF1HH*CONC(1,J-1))
      IF (BIG4.GT.BIG) BIG=BIG4
      BIG5=ABS((H/2.)*(RXN(J+1)/4.))
      IF (BIG5.GT.BIG) BIG=BIG5
      BIG6=ABS((H/2.)*(3.*RXN(J))/4.)
      IF (BIG6.GT.BIG) BIG=BIG6
      BIG7=ABS((HH/2.)*(RXN(J-1)/4.))
      IF (BIG7.GT.BIG) BIG=BIG7
```

```
      BIG8=ABS((HH/2.)*(3.*RXN(J))/4.)
      IF (BIG8.GT.BIG) BIG=BIG8
      BIG9=ABS(VELH12*(CONC(1,J))/2.)
      IF (BIG9.GT.BIG) BIG=BIG9
      BIG10=ABS(VELH12*(CONC(1,J+1))/2.)
      IF (BIG10.GT.BIG) BIG=BIG10
      BIG11=ABS(VELHH12*(CONC(1,J-1))/2.)
      IF (BIG11.GT.BIG) BIG=BIG11
      BIG12=ABS(VELHH12*(CONC(1,J))/2.)
      IF (BIG12.GT.BIG) BIG=BIG12
      BIG13=ABS((H/2.)*CONC(1,J+1)/4.*(VELH12-VEL(J))/(H/2.))
      IF (BIG12.GT.BIG) BIG=BIG13
      BIG14=ABS((H/2.)*3./4.*CONC(1,J)*(VELH12-VEL(J))/(H/2.))
      IF (BIG12.GT.BIG) BIG=BIG14
      BIG15=ABS((HH/2.)*CONC(1,J-1)/4.*(VEL(J)-VELHH12)/(HH/2.))
      IF (BIG12.GT.BIG) BIG=BIG15
      BIG16=ABS((HH/2.)*3./4.*CONC(1,J)*(VEL(J)-VELHH12)/(HH/2.))
      IF (BIG12.GT.BIG) BIG=BIG16
      IF (ABS(G(1)).LT.BIG*EBIG) G(1)=0

      PRINT *, 'A_H=', H
      PRINT *, 'A_HH=', HH

C     For A-
      G(2)=COEFF2H*(CONC(2,J+1)-CONC(2,J))
     1     -VELH12*(CONC(2,J+1)+CONC(2,J))/2.
     2     +(H/2.)*(CONC(2,J+1)+3.*CONC(2,J))/4.*(VELH12-VEL(J))/(H/2.)
     3     +(H/2.)*(RXN(J+1)+3.*RXN(J))/4.
     4     -COEFF2HH*(CONC(2,J)-CONC(2,J-1))
     5     +VELHH12*(CONC(2,J)+CONC(2,J-1))/2.
     6    +(HH/2.)*(CONC(2,J-1)+3.*CONC(2,J))/4.*(VEL(J)-VELHH12)/(HH/2.)
     7     +(HH/2.)*(RXN(J-1)+3.*RXN(J))/4.
      B(2,2)=COEFF2H+VELH12/2.-(H/2.)*3./4.*(VELH12-VEL(J))/(H/2.)
     1       +COEFF2HH-VELHH12/2.-(HH/2.)*3./4.*(VEL(J)-VELHH12)/(HH/2.)
      D(2,2)=-COEFF2H+VELH12/2.-(H/2.)/4.*(VELH12-VEL(J))/(H/2.)
      A(2,2)=-COEFF2HH-VELHH12/2.-(HH/2.)/4.*(VEL(J)-VELHH12)/(HH/2.)
      B(2,4)=-(H/2.)*(3./4.)-(HH/2.)*(3./4.)
      D(2,4)=-(H/2.)*(1./4.)
      A(2,4)=-(HH/2.)*(1./4.)

      BIG=ABS(COEFF2H*CONC(2,J+1))
      BIG2=ABS(COEFF2H*CONC(2,J))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(-COEFF2HH*CONC(2,J))
      IF (BIG3.GT.BIG) BIG=BIG3
      BIG4=ABS(-COEFF2HH*CONC(2,J-1))
      IF (BIG4.GT.BIG) BIG=BIG4
      BIG5=ABS((H/2.)*(RXN(J+1)/4.))
      IF (BIG5.GT.BIG) BIG=BIG5
      BIG6=ABS((H/2.)*(3.*RXN(J))/4.)
      IF (BIG6.GT.BIG) BIG=BIG6
      BIG7=ABS((HH/2.)*(RXN(J-1)/4.))
      IF (BIG7.GT.BIG) BIG=BIG7
      BIG8=ABS((HH/2.)*(3.*RXN(J))/4.)
      IF (BIG8.GT.BIG) BIG=BIG8
      BIG9=ABS(VELH12*(CONC(2,J))/2.)
      IF (BIG9.GT.BIG) BIG=BIG9
      BIG10=ABS(VELH12*(CONC(2,J+1))/2.)
```

```
      IF (BIG10.GT.BIG) BIG=BIG10
      BIG11=ABS(VELHH12*(CONC(2,J-1))/2.)
      IF (BIG11.GT.BIG) BIG=BIG11
      BIG12=ABS(VELHH12*(CONC(2,J))/2.)
      IF (BIG12.GT.BIG) BIG=BIG12
      BIG13=ABS((H/2.)/4.*CONC(2,J+1)*(VELH12-VEL(J))/(H/2.))
      IF (BIG12.GT.BIG) BIG=BIG13
      BIG14=ABS((H/2.)*3./4.*CONC(2,J)*(VELH12-VEL(J))/(H/2.))
      IF (BIG12.GT.BIG) BIG=BIG14
      BIG15=ABS((HH/2.)/4.*CONC(2,J-1)*(VEL(J)-VELHH12)/(HH/2.))
      IF (BIG12.GT.BIG) BIG=BIG15
      BIG16=ABS((HH/2.)*3./4.*CONC(2,J)*(VEL(J)-VELHH12)/(HH/2.))
      IF (BIG12.GT.BIG) BIG=BIG16
      IF (ABS(G(2)).LT.BIG*EBIG) G(2)=0

      PRINT *, 'B_H=', H
      PRINT *, 'B_HH=', HH

      PRINT *, 'COEFF3H=', COEFF3H
      PRINT *, 'COEFF3HH=', COEFF3HH
      PRINT *, 'DIFF(3)=', DIFF(3)

C     For B- The reacting species
      G(3)=COEFF3H*(CONC(3,J+1)-CONC(3,J))
     1     -VELH12*(CONC(3,J+1)+CONC(3,J))/2.
     2     +(H/2.)*(CONC(3,J+1)+3.*CONC(3,J))/4.*(VELH12-VEL(J))/(H/2.)
     3     +(H/2.)*(RXN(J+1)+3.*RXN(J))/4.
     4     -COEFF3HH*(CONC(3,J)-CONC(3,J-1))
     5     +VELHH12*(CONC(3,J)+CONC(3,J-1))/2.
     6    +(HH/2.)*(CONC(3,J-1)+3.*CONC(3,J))/4.*(VEL(J)-VELHH12)/(HH/2.)
     7     +(HH/2.)*(RXN(J-1)+3.*RXN(J))/4.
      B(3,3)=COEFF3H+VELH12/2.-(H/2.)*3./4.*(VELH12-VEL(J))/(H/2.)
     1       +COEFF3HH-VELHH12/2.-(HH/2.)*3./4.*(VEL(J)-VELHH12)/(HH/2.)
      D(3,3)=-COEFF3H+VELH12/2.-(H/2.)/4.*(VELH12-VEL(J))/(H/2.)
      A(3,3)=-COEFF3HH-VELHH12/2.-(HH/2.)/4.*(VEL(J)-VELHH12)/(HH/2.)
      B(3,4)=-(H/2.)*(3./4.)-(HH/2.)*(3./4.)
      D(3,4)=-(H/2.)*(1./4.)
      A(3,4)=-(HH/2.)*(1./4.)

      BIG=ABS(COEFF3H*CONC(3,J+1))
      BIG2=ABS(COEFF3H*CONC(3,J))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(-COEFF3HH*CONC(3,J))
      IF (BIG3.GT.BIG) BIG=BIG3
      BIG4=ABS(-COEFF3HH*CONC(3,J-1))
      IF (BIG4.GT.BIG) BIG=BIG4
      BIG5=ABS((H/2.)*(RXN(J+1)/4.))
      IF (BIG5.GT.BIG) BIG=BIG5
      BIG6=ABS((H/2.)*(3.*RXN(J))/4.)
      IF (BIG6.GT.BIG) BIG=BIG6
      BIG7=ABS((HH/2.)*(RXN(J-1)/4.))
      IF (BIG7.GT.BIG) BIG=BIG7
      BIG8=ABS((HH/2.)*(3.*RXN(J))/4.)
      IF (BIG8.GT.BIG) BIG=BIG8
      BIG9=ABS(VELH12*(CONC(3,J))/2.)
      IF (BIG9.GT.BIG) BIG=BIG9
      BIG10=ABS(VELH12*(CONC(3,J+1))/2.)
      IF (BIG10.GT.BIG) BIG=BIG10
```

```
171       BIG11=ABS(VELHH12*(CONC(3,J-1))/2.)
172       IF (BIG11.GT.BIG) BIG=BIG11
173       BIG12=ABS(VELHH12*(CONC(3,J))/2.)
174       IF (BIG12.GT.BIG) BIG=BIG12
175       BIG13=ABS((H/2.)/4.*CONC(3,J+1)*(VELH12-VEL(J))/(H/2.))
176       IF (BIG12.GT.BIG) BIG=BIG13
177       BIG14=ABS((H/2.)*3./4.*CONC(3,J)*(VELH12-VEL(J))/(H/2.))
178       IF (BIG12.GT.BIG) BIG=BIG14
179       BIG15=ABS((HH/2.)/4.*CONC(3,J-1)*(VEL(J)-VELHH12)/(HH/2.))
180       IF (BIG12.GT.BIG) BIG=BIG15
181       BIG16=ABS((HH/2.)*3./4.*CONC(3,J)*(VEL(J)-VELHH12)/(HH/2.))
182       IF (BIG12.GT.BIG) BIG=BIG16
183       IF (ABS(G(3)).LT.BIG*EBIG) G(3)=0
184
185 C     For Reaction term
186       IF (rateb.EQ.0) GO TO 210
187       EPS=RXN(J)/(rateb*CONC(2,J)*CONC(3,J))
188       IF (ABS(EPS).GT.0.2) GO TO 210
189
190       REM=LOG(1.+EPS)
191       IF (ABS(REM).LT.1.0E-09)REM=EPS*(1.-EPS*(.5-EPS*(1./3.-EPS/4.)))
192
193       G(4)=LOG(equilib*CONC(1,J)/CONC(2,J)/CONC(3,J))-REM
194       B(4,1)=-1./CONC(1,J)
195       B(4,2)=-((RXN(J)/(rateb*CONC(3,J)*CONC(2,J)**2.))*
196      1       (1./(1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J)))))+1./CONC(2,J)
197       B(4,3)=-((RXN(J)/(rateb*CONC(2,J)*CONC(3,J)**2.))*
198      1       (1./(1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J)))))+1./CONC(3,J)
199       B(4,4)=1./(rateb*CONC(3,J)*CONC(2,J)*
200      1     (1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J))))
201
202       BIG=ABS(REM)
203       BIG2=ABS(LOG(equilib*CONC(1,J)/CONC(2,J)/CONC(3,J)))
204       IF (BIG2.GT.BIG) BIG=BIG2
205       IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0
206       GO TO 212
207
208   210 G(4)=-RXN(J)+rateb*(equilib*CONC(1,J)-CONC(2,J)*CONC(3,J))
209       B(4,1)=-rateb*equilib
210       B(4,2)=rateb*CONC(3,J)
211       B(4,3)=rateb*CONC(2,J)
212       B(4,4)=1.
213
214       BIG=ABS(rateb*equilib*CONC(1,J))
215       BIG2=ABS(rateb*CONC(2,J)*CONC(3,J))
216       IF (BIG2.GT.BIG) BIG=BIG2
217       BIG3=ABS(rateb*equilib*CONC(1,J))
218       IF (BIG3.GT.BIG) BIG=BIG3
219       IF (ABS(RXN(J)).GT.BIG) BIG=ABS(RXN(J))
220       IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0
221       CONTINUE
222
223   212 WRITE(12,301) J, (G(K),K=1,N)
224       RETURN
225       END
```

Code D.8. Steady State Convective Diffusion with Homogeneous Reaction Subroutine for the Inner Region

```
      SUBROUTINE INNER(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(4,4),B(4,4),C(4,100001),D(4,9),G(4),X(4,4),Y(4,4)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(3,100001),RXN(100001),DIFF(3),H,EBIG,IJ,KJ,HH
      COMMON/VET/ VEL(100001), Y1, VELH12, VELHH12
      COMMON/RTE/ rateb, equilib

  301 FORMAT (5x,'J=' I5,  8E18.9)
C     For AB
      G(1)=DIFF(1)*(CONC(1,J+1)-2.*CONC(1,J)+CONC(1,J-1))/H**2.
     1     -VEL(J)*(CONC(1,J+1)-CONC(1,J-1))/(2.*H)-RXN(J)
      B(1,1)=2*DIFF(1)/H**2
      D(1,1)=-DIFF(1)/H**2+VEL(J)/(2.*H)
      A(1,1)=-DIFF(1)/H**2-VEL(J)/(2.*H)
      B(1,4)=+1.

      BIG=ABS(DIFF(1)*CONC(1,J+1)/H**2.)
      BIG2=ABS(-1.*DIFF(1)*2.*CONC(1,J)/H**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(DIFF(1)*CONC(1,J-1)/H**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      BIG4=ABS(-1.*VEL(J)*CONC(1,J+1)/(2.*H))
      IF (BIG4.GT.BIG) BIG=BIG4
      BIG5=ABS(VEL(J)*CONC(1,J-1)/(2.*H))
      IF (BIG5.GT.BIG) BIG=BIG5
      IF (ABS(G(1)).LT.BIG*EBIG) G(1)=0

C     For A-
      G(2)=DIFF(2)*(CONC(2,J+1)-2.*CONC(2,J)+CONC(2,J-1))/H**2.
     1     -VEL(J)*(CONC(2,J+1)-CONC(2,J-1))/(2.*H)+RXN(J)
      B(2,2)=2*DIFF(2)/H**2
      D(2,2)=-DIFF(2)/H**2+VEL(J)/(2.*H)
      A(2,2)=-DIFF(2)/H**2-VEL(J)/(2.*H)
      B(2,4)=-1.

      BIG=ABS(DIFF(2)*CONC(2,J+1)/H**2.)
      BIG2=ABS(-1.*DIFF(2)*2.*CONC(2,J)/H**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(DIFF(2)*CONC(2,J-1)/H**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      BIG4=ABS(-1.*VEL(J)*CONC(2,J+1)/(2.*H))
      IF (BIG4.GT.BIG) BIG=BIG4
      BIG5=ABS(VEL(J)*CONC(2,J-1)/(2.*H))
      IF (BIG5.GT.BIG) BIG=BIG5
      IF (ABS(G(2)).LT.BIG*EBIG) G(2)=0

C     For B- The reacting species
      G(3)=DIFF(3)*(CONC(3,J+1)-2*CONC(3,J)+CONC(3,J-1))/H**2
     1     -VEL(J)*(CONC(3,J+1)-CONC(3,J-1))/(2.*H)+RXN(J)
      B(3,3)=2*DIFF(3)/H**2
      D(3,3)=-DIFF(3)/H**2+VEL(J)/(2.*H)
      A(3,3)=-DIFF(3)/H**2-VEL(J)/(2.*H)
      B(3,4)=-1.
```

```
55
56          BIG=ABS(DIFF(3)*CONC(3,J+1)/H**2.)
57          BIG2=ABS(-1.*DIFF(3)*2.*CONC(3,J)/H**2.)
58          IF (BIG2.GT.BIG) BIG=BIG2
59          BIG3=ABS(DIFF(3)*CONC(3,J-1)/H**2.)
60          IF (BIG3.GT.BIG) BIG=BIG3
61          BIG4=ABS(-1.*VEL(J)*CONC(3,J+1)/(2.*H))
62          IF (BIG4.GT.BIG) BIG=BIG4
63          BIG5=ABS(VEL(J)*CONC(3,J-1)/(2.*H))
64          IF (BIG5.GT.BIG) BIG=BIG5
65          IF (ABS(G(3)).LT.BIG*EBIG) G(3)=0
66
67 C        For Reaction term
68          IF (rateb.EQ.0) GO TO 210
69          EPS=RXN(J)/(rateb*CONC(2,J)*CONC(3,J))
70 C         PRINT *, 'EPS2=', EPS
71          IF (ABS(EPS).GT.0.2) GO TO 210
72          REM=LOG(1.+EPS)
73 C         PRINT *, 'REM2=', REM
74        IF (ABS(REM).LT.1.0E-09)REM=EPS*(1.-EPS*(.5-EPS*(1./3.-EPS/4.)))
75          G(4)=LOG(equilib*CONC(1,J)/CONC(2,J)/CONC(3,J))-REM
76          B(4,1)=-1./CONC(1,J)
77          B(4,2)=-((RXN(J)/(rateb*CONC(3,J)*CONC(2,J)**2.))*
78         1       (1./(1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J)))))+1./CONC(2,J)
79          B(4,3)=-((RXN(J)/(rateb*CONC(2,J)*CONC(3,J)**2.))*
80         1       (1./(1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J)))))+1./CONC(3,J)
81          B(4,4)=1./(rateb*CONC(3,J)*CONC(2,J)*
82         1      (1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J))))
83
84          BIG=ABS(REM)
85          BIG2=ABS(LOG(equilib*CONC(1,J)/CONC(2,J)/CONC(3,J)))
86          IF (BIG2.GT.BIG) BIG=BIG2
87          IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0
88          GO TO 209
89
90    210 G(4)=-RXN(J)+rateb*(equilib*CONC(1,J)-CONC(2,J)*CONC(3,J))
91          B(4,1)=-rateb*equilib
92          B(4,2)=rateb*CONC(3,J)
93          B(4,3)=rateb*CONC(2,J)
94          B(4,4)=1.
95
96          BIG=ABS(rateb*equilib*CONC(1,J))
97          BIG2=ABS(rateb*CONC(2,J)*CONC(3,J))
98          IF (BIG2.GT.BIG) BIG=BIG2
99          BIG3=ABS(rateb*equilib*CONC(1,J))
100         IF (BIG3.GT.BIG) BIG=BIG3
101         IF (ABS(RXN(J)).GT.BIG) BIG=ABS(RXN(J))
102         IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0
103         CONTINUE
104
105   209 IF (J.EQ.2) THEN
106         WRITE(12,301) J, (G(K),K=1,N)
107         ELSE IF (J.EQ.3) THEN
108         WRITE(12,301) J, (G(K),K=1,N)
109         ELSE IF (J.EQ.(NJ-1)) THEN
110         WRITE(12,301) J, (G(K),K=1,N)
111         END IF
112         RETURN
```

```
113         END
```

Code D.9. Steady State Convective Diffusion with Homogeneous Reaction Subroutine for the Bulk Boundary Condition

```
      SUBROUTINE BCNJ(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(4,4),B(4,4),C(4,100001),D(4,9),G(4),X(4,4),Y(4,4)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(3,100001),RXN(100001),DIFF(3),H,EBIG,IJ,KJ,HH
      COMMON/BUL/ CBULK(6), JCOUNT
      COMMON/RTE/ rateb, equilib
  301 FORMAT (5x,'J=' I5,  8E18.9)

   29 DO 14 I=1,N-1
      G(I)=CBULK(I)-CONC(I,J)
   14 B(I,I)=1.0
      DO 121 I=1,N-1
      IF (ABS(CONC(I,J)).GT.BIG) BIG=ABS(CONC(I,J))
  121 IF (ABS(G(I)).LT.BIG*EBIG) G(I)=0

C     For Reaction term
      IF (rateb.EQ.0) GO TO 207
      EPS=RXN(J)/(rateb*CONC(2,J)*CONC(3,J))
      IF (ABS(EPS).GT.0.2) GO TO 207
      REM=LOG(1.+EPS)
    IF (ABS(REM).LT.1.0E-09)REM=EPS*(1.-EPS*(.5-EPS*(1./3.-EPS/4.)))
      G(4)=LOG(equilib*CONC(1,J)/CONC(2,J)/CONC(3,J))-REM
      B(4,1)=-1./CONC(1,J)
      B(4,2)=-((RXN(J)/(rateb*CONC(3,J)*CONC(2,J)**2))*
     1      (1/(1+RXN(J)/(rateb*CONC(2,J)*CONC(3,J)))))+1/CONC(2,J)
      B(4,3)=-((RXN(J)/(rateb*CONC(2,J)*CONC(3,J)**2))*
     1      (1/(1+RXN(J)/(rateb*CONC(2,J)*CONC(3,J)))))+1/CONC(3,J)
      B(4,4)=1./(rateb*CONC(3,J)*CONC(2,J)*
     1     (1.+RXN(J)/(rateb*CONC(2,J)*CONC(3,J))))
      BIG=ABS(REM)
      BIG2=ABS(LOG(equilib*CONC(1,J)/CONC(2,J)/CONC(3,J)))
      IF (BIG2.GT.BIG) BIG=BIG2
      IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0
      GO TO 206

  207 G(4)=-RXN(J)+rateb*(equilib*CONC(1,J)-CONC(2,J)*CONC(3,J))
      B(4,1)=-rateb*equilib
      B(4,2)=rateb*CONC(3,J)
      B(4,3)=rateb*CONC(2,J)
      B(4,4)=1.
      BIG=ABS(rateb*equilib*CONC(1,J))
      BIG2=ABS(rateb*CONC(2,J)*CONC(3,J))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(rateb*equilib*CONC(1,J))
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(RXN(J)).GT.BIG) BIG=ABS(RXN(J))
      IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0

      CONTINUE
  206  WRITE(12,301) J, (G(K),K=1,N)
      PRINT *, 'ITERATION=', JCOUNT
      RETURN
      END
```

Code D.10. Matlab code to plot results from steady-state solutions

```
%Steady State
clc; close all;clear all;
format longE;

%Read constant values used in the Fortran code
M = dlmread('cdh_ssvalues_out.txt');

N=M(1);
NJ=M(2);
KJ=M(3);
H=M(4);
HH=M(5);
RPM=M(6);
ANU=M(7);
DiffB=M(8);
AKB=M(9);
BB=M(10);
POT=M(11);

ROT=RPM*2*3.141592653589793/60;

%Read the velocity profile stuff
V = dlmread('VEL.txt');

%Read the steady state values for CB
Bss1 = dlmread('cdh_out.txt');
Bss=Bss1(:,3);

%Other constants
F=96487;
alpha=0.51023*ROT^(3./2.)/ANU^(0.5);

%Create rates
ee=-BB*POT;
i=-Bss(1)*AKB*exp(ee);

i2=-F*DiffB*((-Bss(3)+4*Bss(2)-3*Bss(1))/(2*H));

%Create y values for plotting
y=zeros(NJ,1);

far=HH*(KJ-1);
y1=0:HH:far;

far1=H*(NJ-KJ);
y2=y1(KJ):H:y1(KJ)+far1;

for i=1:KJ-1
    y(i)=y1(i);
end
for i=KJ:NJ
    y(i)=y2(i-KJ+1);
end

figure(1) %Plot velocity
plot(y,V(:,1),'-b'); hold on;
```

```
plot(y,V(:,2),'-k');
plot(y,V(:,4),'-r');
%axis([0 y(2)*10 -1.25 0.25]);
title('Velocity Profile for H');
xlabel('Length, cm');
ylabel('Velocity, cm/s');


figure(3)
plot(y,Bss1(:,3),'-r'); hold on;
%axis([0 H*4000 0 10.05e-5]);
title('Steady State Concentration of B+ away from Electrode Surface');
xlabel('Length, cm');
ylabel('Concentration, moles/cm3');

figure(4)
plot(y,Bss1(:,1),'-b'); hold on;
%plot(y,Bss1(:,2),'-g'); hold on;
%axis([0 H*4000 0 10.05e-5]);
title('AB Steady State Concentration away from Electrode Surface');
xlabel('Length, cm');
ylabel('Concentration, moles/cm3');

figure(6)
%plot(y,Bss1(:,1),'-b'); hold on;
plot(y,Bss1(:,2),'-g'); hold on;
%axis([0 H*4000 0 10.05e-5]);
title('A- Steady State Concentration away from Electrode Surface');
xlabel('Length, cm');
ylabel('Concentration, moles/cm3');

figure(5)
plot(y,Bss1(:,4),'-k'); hold on;
title('Reaction term away from Electrode Surface');
xlabel('Length, cm');
ylabel('Rate of Reaction');
```

Code D.11. Matlab code to create and plot polarization curve

```
clc; close all;clear all;
format longE;

h=0.05;                     %Step-size
V=-2.5:h:-1.;               %Potential range
Current=length(V);          %Current to be saved

for k=1:length(V);
fileID = fopen('potential_in.txt','w');
fprintf(fileID,'%8.3f',V(k));

%Run the executable
system('cdh_ss.exe')
pause(0.01) %in seconds

%Read constant values used in the Fortran code
M = dlmread('cdh_ssvalues_out.txt');

N=M(1);
NJ=M(2);
KJ=M(3);
H=M(4);
HH=M(5);
RPM=M(6);
ANU=M(7);
DiffB=M(8);
AKB=M(9);
BB=M(10);
POT=M(11);

ROT=RPM*2*3.141592653589793/60;

%Read the steady state values for CB
Bss1 = dlmread('cdh_out.txt');
Bss=Bss1(:,3);

%Other constants
F=96487;
alpha=0.51023*ROT^(3./2.)/ANU^(0.5);

%Create rates
ee=-BB*POT;
i=-Bss(1)*AKB*exp(ee);

i2=-F*DiffB*((-Bss(3)+4*Bss(2)-3*Bss(1))/(2*H));

%Save current name
Current(k)=i;
end

figure(1)
plot(V,Current,'-. b'); hold on;
title('Polarization Curve');
```

D.3 Oscillating Convective Diffusion Impedance with Homogeneous Reactions Code

This section contains the oscillating FORTRAN codes used to solve the convective diffusion equation with a homogeneous reaction. It reads the steady-state input file in order to solve for the oscillating concentrations. The mathematical workup for these codes are in Chapter 4. The FORTRAN codes are followed a Matlab code that reads the oscillating concentration of the reacting species and creates the dimensionless diffusion-impedance.

The first section in the code, called CONVDIFFOSCILLATING, is the main program, which outlines the global variables and sets up calling files to save over as output files as well as calling the input files. Then the subroutines that are called in the main program are all shown. They are the same titled subroutines as the steady state.

Code D.12. Oscillating Convective Diffusion with Homogeneous Reaction Main Program

```
C     Convective Diffusion Equation with Homogeneous Reaction
C     3 species system
C     SPECIES 1 = AB, SPECIES 2 = A-, SPECIES 3 = B+
C     Species 3 is the reacting species
C     This is the unsteady state solution that will eventually lead to
c     the impedance!

C     This should be ran after cdh_ss.for
C     The input file is the same for both of these

      PROGRAM CONVDIFFOSCILLATING
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(8,8),B(8,8),C(8,100001),D(8,17),G(8),X(8,8),Y(8,8)
      COMMON/NST/ N, NJ
      COMMON/CON/ C1(2,100001),C2(2,100001),C3(2,100001),RXN(2,100001)
      COMMON/RTE/ rateb,equilib,H,EBIG,HH,KJ
      COMMON/BCI/ VEL1, FLUX, omega
      COMMON/CAR/ CONCSS(3,100001),CBULK(3),DIFF(3),Z(3),REF(3)
      COMMON/VAR/ RXNSS(100001),VELNEAR(100001),VELFAR(100001)
      COMMON/FRE/ CB(2010,100001), FREQ(100001),VEL(100001),FVEL(100001)
      COMMON/VEL12/  VELH12, VELHH12
      CHARACTER REF*6

  102 FORMAT (/30H THE NEXT RUN DID NOT CONVERGE)
  103 FORMAT ('Error=',E16.6/(1X,'Species=',A6,2X,'Conc at Electrode=',
     1 E12.5,2X,'Conc at Bulk=',E12.5))
  334 FORMAT (4(E25.15,5X))
  305 FORMAT (E20.12,3X,E20.12,3X,E20.12,3X,E20.12)
  335 FORMAT (8(E25.15,5X))
  336 FORMAT (1000(E25.15,1X))
  339 FORMAT (1000(E25.15,1X))
  301 FORMAT (5x,'J=' I5,  8E18.9)
  302 FORMAT ('Iteration='I4)

C     Read input values used in steady state
      open(10,file='cdh_in.txt',status='old')
      read(10,*) N,NJ,KJ,Y1,Y2,RPM,ANU,equilib,rateb,AKB,BB,EBIG
      read(10,*) (DIFF(I),Z(I),REF(I),CBULK(I),I=1,(N-1))

      open(18,file='potential_in.txt',status='old')
      read(18,*) V

C     Read steady state values from previous file
      OPEN(UNIT=11, FILE='cdh_out.txt')
      READ(11,334) (CONCSS(1,I),CONCSS(2,I),CONCSS(3,I),RXNSS(I),I=1,NJ)

C     Read velocity values from previous file
      OPEN(UNIT=12, FILE='VEL.txt')
      READ(12,305) (VELNEAR(I),VELFAR(I),FVEL(I),VEL(I),I=1,NJ)

      OPEN(UNIT=13, FILE='cdh_os_out.txt')
      CLOSE(UNIT=13, STATUS='DELETE')
      OPEN(UNIT=13, FILE='cdh_os_out.txt')

      OPEN(14,FILE='cdh_G_out.txt')
      CLOSE(14, STATUS='DELETE')
```

```
57        OPEN(14, FILE='cdh_G_out.txt')
58
59        OPEN(15,FILE='cdh_B_out.txt')
60        CLOSE(15, STATUS='DELETE')
61        OPEN(15, FILE='cdh_B_out.txt')
62
63        OPEN(16,FILE='cdh_values_out.txt')
64        CLOSE(16, STATUS='DELETE')
65        OPEN(16, FILE='cdh_values_out.txt')
66
67        OPEN(17,FILE='k_values_out.txt')
68        CLOSE(17, STATUS='DELETE')
69        OPEN(17, FILE='k_values_out.txt')
70
71 C      Constants
72        F=96487.
73        ROT=RPM*2*3.141592653589793/60
74
75        PRINT *, 'Y2=', Y2
76        H=Y2/(NJ-KJ)
77        PRINT *, 'H=', H
78
79        PRINT *, 'Y1=', Y1
80        HH=Y1/(KJ-1)
81        PRINT *, 'HH=', HH
82
83 C      Create flux of the reacting species constants
84        FLUX=-AKB*exp(-BB*V)/F/Z(3)
85        PRINT *, 'FLUX=', FLUX
86
87 C      Create charge transfer resistance
88        RTB=1/(AKB*BB*CONCSS(3,1)*EXP(-BB*V))
89        PRINT *, 'Charge Transfer Resistance=', RTB
90
91 C      Create delta
92        delta=(3.*DIFF(3)/.51023/ANU)**(1./3.)*(ANU/ROT)**(1./2.)
93
94        N=2*N
95        PRINT *, 'N=', N
96    337 FORMAT (I2/I7/I7/E15.8/E15.8/E15.8/E15.8/E15.8/E15.8/E15.8/
97       1      E15.8/E15.8/E15.8)
98        write (16,337) N,NJ,KJ,H,HH,V,AKB,BB,RTB,DIFF(3),delta,ROT,ANU
99
100       VEL1=0.25*VEL(2)
101
102 C     The number of points for frequency
103       NPTS=241
104       PRINT *, 'NPTS=', NPTS
105
106 c     Create range for the dimensionless frequency
107       DO 261 I=1,NPTS
108       FREQ(I)=10.**(-5.+0.05*(I-1.))
109   261 WRITE (17,339), FREQ(I)
110
111
112 C c       The number of points for frequency
113 C         NPTS=13
114 C         PRINT *, 'NPTS=', NPTS
```

```
C
C c      Create range for the dimensionless frequency
C        DO 261 I=1,NPTS
C        FREQ(I)=10.**(-3.+0.5*(I-1.))
C   261  WRITE (17,339), FREQ(I)

        DO 19 nf=1,NPTS
c        DO 19 nf=1,3

        PRINT *, 'FREQ(NF)=', FREQ(NF)
        omega=FREQ(NF)*DIFF(3)/(delta)**2

        PRINT *, 'omega=', omega
  340 FORMAT (E12.6)
        write (17,340), omega

C       Start actual code
        DO 20 J=1,NJ
        DO 20 I=1,N
   20 C(I,J)=0.0
        DO 21 J=1,NJ
        DO 21 K=1,2
        C1(K,J)=0.0
        C2(K,J)=0.0
        C3(K,J)=0.0
   21 RXN(K,J)=0.0
        JCOUNT=0
        TOL=1.E-10*N*NJ
   22 JCOUNT=JCOUNT+1
        AMP=0.0
        J=0
        DO 23 I=1,N
        DO 23 K=1,N
        Y(I,K)=0.0
   23 X(I,K)=0.0
   24 J=J+1
        DO 25  I=1,N
        G(I)=0.0
        DO 25 K=1,N
        A(I,K)=0.0
        B(I,K)=0.0
   25 D(I,K)=0.0

        IF (J.EQ.1) CALL BC1(J)
        IF (J.GT.1. AND. J.LT.KJ) CALL REACTION(J)
        IF (J.EQ.KJ) CALL COUPLER(J)
        IF (J.GT.KJ .AND. J.LT.NJ) CALL INNER(J)
        IF (J.EQ.NJ)  CALL BCNJ(J)
        CALL BAND(J)

        AMP=DABS(G(1))+DABS(G(2))+DABS(G(3))+DABS(G(4))+DABS(G(5))
       1     +DABS(G(6))+DABS(G(7))+DABS(G(8))

        IF (J.LT.NJ)  GO TO 24

        PRINT *, 'ERROR=', AMP

        DO 16 K=1,NJ
```

```
      DO 16 I=1,2
      C1(I,K)=C1(I,K)+C(I,K)
      C2(I,K)=C2(I,K)+C(I+2,K)
      C3(I,K)=C3(I,K)+C(I+4,K)
      RXN(I,K)=RXN(I,K)+C(I+6,K)
   16 CONTINUE

      WRITE(14,302) (JCOUNT)

      IF (DABS(AMP).LT.DABS(TOL)) GO TO 15

      IF (JCOUNT.LE.4) GO TO 22
      print 102

   15 CONTINUE
      PRINT *,'JCOUNT=',JCOUNT

      PRINT *, 'nf1=', nf

      DO 18 I=1,2
      DO 18 J=1,NJ
      BIG=C3(I,J)
      BIG2=1.0E-40
   18 IF (ABS(BIG).LE.BIG2) C3(I,J)=0.0

      WRITE(13,335) (C1(1,J),C1(2,J),C2(1,J),C2(2,J),C3(1,J),C3(2,J),
     1      RXN(1,J),RXN(2,J),J=1,NJ)

      DO 19 J=1,NJ
      CB(2*nf-1,J)=C3(1,J)
   19 CB(2*nf,J)=C3(2,J)

c     for some reason nf is one greater then necessary
      PRINT *, 'nf2=', nf

C      DO 17 I=1,2*nf-2
      nf=nf-1
    DO 17 J=1,NJ
   17 WRITE(15,336) (CB(I,J), I=1,2*nf)

  338 FORMAT (I5)
      write (16,338) nf


      XI=(ROT/ANU)**(0.5)*(3*DIFF(3)/.51023/ANU)**(-1./3.)
      PRINT *, 'XI/Y=', XI
      PRINT *, 'delta=', delta
      PRINT *, 'ROT=', ROT
      PRINT *, 'ANU=', ANU
      PRINT *, 'DIFF(3)=', DIFF(3)

      END PROGRAM CONVDIFFOSCILLATING
```

Code D.13. Oscillating Convective Diffusion with Homogeneous Reaction Subroutine for the Electrode Boundary Condition

```
      SUBROUTINE BC1(J)
       IMPLICIT DOUBLE PRECISION (A-H, O-Z)
       COMMON/BAT/ A(8,8),B(8,8),C(8,100001),D(8,17),G(8),X(8,8),Y(8,8)
       COMMON/NST/ N, NJ
       COMMON/CON/ C1(2,100001),C2(2,100001),C3(2,100001),RXN(2,100001)
       COMMON/RTE/ rateb,equilib,H,EBIG,HH,KJ
       COMMON/BCI/ VEL1, FLUX, omega
       COMMON/CAR/ CONCSS(3,100001),CBULK(3),DIFF(3),Z(3),REF(3)
       COMMON/VAR/ RXNSS(100001),VELNEAR(100001),VELFAR(100001)
       COMMON/FRE/ CB(2010,100001), FREQ(100001),VEL(100001),FVEL(100001)

  301 FORMAT (5x,'J=' I5,  16E15.6)

C      For AB  non reacting species
       G(1)=omega*(3.*C1(1,J)+C1(1,J+1))/4.
     1    +2.*DIFF(1)*(C1(1,J+1)-C1(1,J))/HH**2.
     1    -VEL1*(C1(1,J+1)-C1(1,J))/HH
     2    -(3.*RXN(1,J)+RXN(1,J+1))/4.
       B(1,1)=-omega*3./4.+2.*DIFF(1)/HH**2.-VEL1*HH
       D(1,1)=-omega*1./4.-2.*DIFF(1)/HH**2.+VEL1*HH
       B(1,7)=+3./4.
       D(1,7)=+1./4.

       G(2)=-omega*(3.*C1(2,J)+C1(2,J+1))/4.
     1    +2.*DIFF(1)*(C1(2,J+1)-C1(2,J))/HH**2.
     2    -VEL1*(C1(2,J+1)-C1(2,J))/HH
     3    -(3.*RXN(2,J)+RXN(2,J+1))/4.
       B(2,2)=omega*3./4.+2.*DIFF(1)/HH**2.-VEL1*HH
       D(2,2)=omega*1./4.-2.*DIFF(1)/HH**2.+VEL1*HH
       B(2,8)=+3./4.
       D(2,8)=+1./4.

C      For A+  non reacting species
       G(3)=omega*(3.*C2(2,J)+C2(2,J+1))/4.
     1    +2.*DIFF(2)*(C2(1,J+1)-C2(1,J))/HH**2.
     1    -VEL1*(C2(1,J+1)-C2(1,J))/HH
     2    +(3.*RXN(2,J)+RXN(2,J+1))/4.
       B(3,3)=-omega*3./4.+2.*DIFF(2)/HH**2.-VEL1*HH
       D(3,3)=-omega*1./4.-2.*DIFF(2)/HH**2.+VEL1*HH
       B(3,7)=-3./4.
       D(3,7)=-1./4.

       G(4)=-omega*(3.*C2(2,J)+C2(2,J+1))/4.
     1    +2.*DIFF(2)*(C2(2,J+1)-C2(2,J))/HH**2.
     2    -VEL1*(C2(2,J+1)-C2(2,J))/HH
     3    +(3.*RXN(2,J)+RXN(2,J+1))/4.
       B(4,4)=omega*3./4.+2.*DIFF(2)/HH**2.-VEL1*HH
       D(4,4)=omega*1./4.-2.*DIFF(2)/HH**2.+VEL1*HH
       B(4,8)=-3./4.
       D(4,8)=-1./4.

C      For B- reacting species
       G(5)=1-C3(1,J)
       B(5,5)=1.
```

```
55
56          G(6)=C3(2,J)
57          B(6,6)=-1.
58
59 C        For Reaction term
60          G(7)=-RXN(1,J)+rateb*equilib*C1(1,J)-rateb*CONCSS(2,J)*C3(1,J)
61         1         -rateb*CONCSS(3,J)*C2(1,J)
62          B(7,1)=-rateb*equilib
63          B(7,3)=rateb*CONCSS(3,J)
64          B(7,5)=rateb*CONCSS(2,J)
65          B(7,7)=+1.
66
67          G(8)=-RXN(2,J)+rateb*equilib*C1(2,J)-rateb*CONCSS(2,J)*C3(2,J)
68         1         -rateb*CONCSS(3,J)*C2(2,J)
69          B(8,2)=-rateb*equilib
70          B(8,4)=rateb*CONCSS(3,J)
71          B(8,6)=rateb*CONCSS(2,J)
72          B(8,8)=+1.
73
74          WRITE(14,301) J, (G(K),K=1,N)
75
76          RETURN
77          END
```

Code D.14. Oscillating Convective Diffusion with Homogeneous Reaction Subroutine for the Electrode Boundary Condition

```
      SUBROUTINE REACTION(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(8,8),B(8,8),C(8,100001),D(8,17),G(8),X(8,8),Y(8,8)
      COMMON/NST/ N, NJ
      COMMON/CON/ C1(2,100001),C2(2,100001),C3(2,100001),RXN(2,100001)
      COMMON/RTE/ rateb,equilib,H,EBIG,HH,KJ
      COMMON/BCI/ VEL1, FLUX, omega
      COMMON/CAR/ CONCSS(3,100001),CBULK(3),DIFF(3),Z(3),REF(3)
      COMMON/VAR/ RXNSS(100001),VELNEAR(100001),VELFAR(100001)
      COMMON/FRE/ CB(2010,100001), FREQ(100001),VEL(100001),FVEL(100001)

  301 FORMAT (5x,'J=' I5,  16E15.6)

C     For AB
      G(1)=omega*C1(2,J)
     1     +DIFF(1)*(C1(1,J+1)-2.*C1(1,J)+C1(1,J-1))/HH**2.
     2     -VEL(J)*(C1(1,J+1)-C1(1,J-1))/(2.*HH)
     3     -RXN(1,J)
      B(1,1)=2.*DIFF(1)/HH**2.
      A(1,1)=-DIFF(1)/HH**2.-VEL(J)/(2.*HH)
      D(1,1)=-DIFF(1)/HH**2.+VEL(J)/(2.*HH)
      B(1,2)=-omega
      B(1,7)=+1.

      G(2)=-omega*C1(1,J)
     1     +DIFF(1)*(C1(2,J+1)-2.*C1(2,J)+C1(2,J-1))/HH**2.
     2     -VEL(J)*(C1(2,J+1)-C1(2,J-1))/(2.*HH)
     3     -RXN(2,J)
      B(2,2)=2.*DIFF(1)/HH**2.
      A(2,2)=-DIFF(1)/HH**2.-VEL(J)/(2.*HH)
      D(2,2)=-DIFF(1)/HH**2.+VEL(J)/(2.*HH)
      B(2,1)=omega
      B(2,8)=+1.

C     For A-
      G(3)=omega*C2(2,J)+DIFF(2)*(C2(1,J+1)-2.*C2(1,J)+C2(1,J-1))/HH**2.
     1     -VEL(J)*(C2(1,J+1)-C2(1,J-1))/(2.*HH)+RXN(1,J)
      B(3,3)=2.*DIFF(2)/HH**2.
      A(3,3)=-DIFF(2)/HH**2.-VEL(J)/(2.*HH)
      D(3,3)=-DIFF(2)/HH**2.+VEL(J)/(2.*HH)
      B(3,4)=-omega
      B(3,7)=-1.

      G(4)=-omega*C2(1,J)+DIFF(2)*(C2(2,J+1)-2.*C2(2,J)+C2(2,J-1))/HH**2.
     1     -VEL(J)*(C2(2,J+1)-C2(2,J-1))/(2.*HH)+RXN(2,J)
      B(4,4)=2.*DIFF(2)/HH**2.
      A(4,4)=-DIFF(2)/HH**2.-VEL(J)/(2.*HH)
      D(4,4)=-DIFF(2)/HH**2.+VEL(J)/(2.*HH)
      B(4,3)=omega
      B(4,8)=-1.

C     For B+
      G(5)=omega*C3(2,J)-DIFF(3)*(C3(1,J+1)-2.*C3(1,J)+C3(1,J-1))/HH**2.
     1     -VEL(J)*(C3(1,J+1)-C3(1,J-1))/(2.*HH)+RXN(1,J)
```

```
55        B(5,5)=2.*DIFF(3)/HH**2.
56        A(5,5)=-DIFF(3)/HH**2.-VEL(J)/(2.*HH)
57        D(5,5)=-DIFF(3)/HH**2.+VEL(J)/(2.*HH)
58        B(5,6)=-omega
59        B(5,7)=-1.
60
61        G(6)=-omega*C3(1,J)-DIFF(3)*(C3(2,J+1)-2.*C3(2,J)+C3(2,J-1))/HH**2.
62       1      -VEL(J)*(C3(2,J+1)-C3(2,J-1))/(2.*HH)+RXN(2,J)
63        B(6,6)=2.*DIFF(3)/HH**2.
64        A(6,6)=-DIFF(3)/HH**2.-VEL(J)/(2.*HH)
65        D(6,6)=-DIFF(3)/HH**2.+VEL(J)/(2.*HH)
66        B(6,5)=omega
67        B(6,8)=-1.
68
69 C      For Reaction term
70        G(7)=-RXN(1,J)+rateb*equilib*C1(1,J)-rateb*CONCSS(2,J)*C3(1,J)
71       1        -rateb*CONCSS(3,J)*C2(1,J)
72        B(7,1)=-rateb*equilib
73        B(7,3)=rateb*CONCSS(3,J)
74        B(7,5)=rateb*CONCSS(2,J)
75        B(7,7)=+1.
76
77        G(8)=-RXN(2,J)+rateb*equilib*C1(2,J)-rateb*CONCSS(2,J)*C3(2,J)
78       1        -rateb*CONCSS(3,J)*C2(2,J)
79        B(8,2)=-rateb*equilib
80        B(8,4)=rateb*CONCSS(3,J)
81        B(8,6)=rateb*CONCSS(2,J)
82        B(8,8)=+1.
83
84    212 WRITE(12,301) J, (G(K),K=1,N)
85
86        RETURN
87        END
```

Code D.15. Oscillating Convective Diffusion with Homogeneous Reaction Subroutine for the Electrode Boundary Condition

```
      SUBROUTINE COUPLER(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(8,8),B(8,8),C(8,100001),D(8,17),G(8),X(8,8),Y(8,8)
      COMMON/NST/ N, NJ
      COMMON/CON/ C1(2,100001),C2(2,100001),C3(2,100001),RXN(2,100001)
      COMMON/RTE/ rateb,equilib,H,EBIG,HH,KJ
      COMMON/BCI/ VEL1, FLUX, omega
      COMMON/CAR/ CONCSS(3,100001),CBULK(3),DIFF(3),Z(3),REF(3)
      COMMON/VAR/ RXNSS(100001),VELNEAR(100001),VELFAR(100001)
      COMMON/VEL12/  VELH12, VELHH12
      COMMON/FRE/ CB(2010,100001), FREQ(100001),VEL(100001),FVEL(100001)
  301 FORMAT (5x,'J=' I5,  8E15.6)

C     For AB, real
      G(1)=-HH/2*omega*(C1(2,J+1)+3.*C1(2,J))/4.
     1     +H/2*omega*(C1(2,J-1)+3.*C1(2,J))/4.
     2     +DIFF(1)/H*(C1(1,J+1)-C1(1,J))
     3     -DIFF(1)/HH*(C1(1,J)-C1(1,J-1))
     4     -VELH12*(C1(1,J+1)+C1(1,J))/2.
     5     +VELHH12*(C1(1,J)+C1(1,J-1))/2.
     6     +(H/2.)*(C1(1,J+1)+3.*C1(1,J))/4.*(VELH12-VEL(J))/(H/2.)
     7     +(HH/2.)*(C1(1,J-1)+3.*C1(1,J))/4.*(VEL(J)-VELHH12)/(HH/2.)
     8     -(H/2.)*(RXN(1,J+1)+3.*RXN(1,J))/4.
     9     -(HH/2.)*(RXN(1,J-1)+3.*RXN(1,J))/4.
      B(1,1)=DIFF(1)/H+DIFF(1)/HH-(H/2.)*3./4.*(VELH12-VEL(J))/(H/2.)
     1       -(HH/2.)*3./4.*(VEL(J)-VELHH12)/(HH/2.)
      D(1,1)=-DIFF(1)/H+(VEL(J)+VEL(J+1))/4.
     1       -(H/2.)/4.*(VELH12-VEL(J))/(H/2.)
      A(1,1)=-DIFF(1)/HH-(VEL(J)+VEL(J-1))/4.
     1       -(HH/2.)/4.*(VEL(J)-VELHH12)/(HH/2.)
      B(1,2)=-HH/2*omega*(3./4.)-H/2*omega*(3./4.)
      D(1,2)=-HH/2*omega*(1./4.)
      A(1,2)=-H/2*omega*(1./4.)
      B(1,7)=(H/2.)*3./4.+(HH/2.)*(1./4.)
      D(1,7)=(H/2.)*3./4.
      A(1,7)=(HH/2.)*(1./4.)
C     For AB, imaginary
      G(2)=-HH/2*omega*(C1(1,J+1)+3.*C1(1,J))/4.
     1     -H/2*omega*(C1(1,J-1)+3.*C1(1,J))/4.
     2     +DIFF(1)/H*(C1(2,J+1)-C1(2,J))
     3     -DIFF(1)/HH*(C1(2,J)-C1(2,J-1))
     4     -(VEL(J)+VEL(J+1))/2.*(C1(2,J+1)+C1(2,J))/2.
     5     +(VEL(J)+VEL(J-1))/2.*(C1(2,J)+C1(2,J-1))/2.
     6     +(H/2.)*(C1(2,J+1)+3.*C1(2,J))/4.*(VELH12-VEL(J))/(H/2.)
     7     +(HH/2.)*(C1(2,J-1)+3.*C1(2,J))/4.*(VEL(J)-VELHH12)/(HH/2.)
     8     -(H/2.)*(RXN(2,J+1)+3.*RXN(2,J))/4.
     9     -(HH/2.)*(RXN(2,J-1)+3.*RXN(2,J))/4.
      B(2,2)=DIFF(1)/H+DIFF(1)/HH-(H/2.)*3./4.*(VELH12-VEL(J))/(H/2.)
     1       -(HH/2.)*3./4.*(VEL(J)-VELHH12)/(HH/2.)
      D(2,2)=-DIFF(1)/H+(VEL(J)+VEL(J+1))/4.
     1       -(H/2.)/4.*(VELH12-VEL(J))/(H/2.)
      A(2,2)=-DIFF(1)/HH-(VEL(J)+VEL(J-1))/4.
     1       -(HH/2.)/4.*(VEL(J)-VELHH12)/(HH/2.)
      B(2,1)=+HH/2*omega*(3./4.)+H/2*omega*(3./4.)
```

```
      D(2,1)=+HH/2*omega*(1./4.)
      A(2,1)=+H/2*omega*(1./4.)
      B(2,8)=(H/2.)*3./4.+(HH/2.)*(1./4.)
      D(2,8)=(H/2.)*3./4.
      A(2,8)=(HH/2.)*(1./4.)

C     For A-, real
      G(3)=-HH/2*omega*(C2(2,J+1)+3.*C2(2,J))/4.
     1     +H/2*omega*(C2(2,J-1)+3.*C2(2,J))/4.
     2     +DIFF(2)/H*(C2(1,J+1)-C2(1,J))
     3     -DIFF(2)/HH*(C2(1,J)-C2(1,J-1))
     4     -(VEL(J)+VEL(J+1))/2.*(C2(1,J+1)+C2(1,J))/2.
     5     +(VEL(J)+VEL(J-1))/2.*(C2(1,J)+C2(1,J-1))/2.
     6     +(H/2.)*(C2(1,J+1)+3.*C2(1,J))/4.*(VELH12-VEL(J))/(H/2.)
     7     +(HH/2.)*(C2(1,J-1)+3.*C2(1,J))/4.*(VEL(J)-VELHH12)/(HH/2.)
     8     +(H/2.)*(RXN(1,J+1)+3.*RXN(1,J))/4.
     9     +(HH/2.)*(RXN(1,J-1)+3.*RXN(1,J))/4.
      B(3,3)=DIFF(2)/H+DIFF(2)/HH-(H/2.)*3./4.*(VELH12-VEL(J))/(H/2.)
     1       -(HH/2.)*3./4.*(VEL(J)-VELHH12)/(HH/2.)
      D(3,3)=-DIFF(2)/H+(VEL(J)+VEL(J+1))/4.
     1       -(H/2.)/4.*(VELH12-VEL(J))/(H/2.)
      A(3,3)=-DIFF(2)/HH-(VEL(J)+VEL(J-1))/4.
     1       -(HH/2.)/4.*(VEL(J)-VELHH12)/(HH/2.)
      B(3,4)=-HH/2*omega*(3./4.)-H/2*omega*(3./4.)
      D(3,4)=-HH/2*omega*(1./4.)
      A(3,4)=-H/2*omega*(1./4.)
      B(3,7)=-(H/2.)*3./4.-(HH/2.)*(1./4.)
      D(3,7)=-(H/2.)*3./4.
      A(3,7)=-(HH/2.)*(1./4.)
C     For A-, imaginary
      G(4)=-HH/2*omega*(C2(1,J+1)+3.*C2(1,J))/4.
     1     -H/2*omega*(C2(1,J-1)+3.*C2(1,J))/4.
     2     +DIFF(2)/H*(C2(2,J+1)-C2(2,J))
     3     -DIFF(2)/HH*(C2(2,J)-C2(2,J-1))
     4     -(VEL(J)+VEL(J+1))/2.*(C2(2,J+1)+C2(2,J))/2.
     5     +(VEL(J)+VEL(J-1))/2.*(C2(2,J)+C2(2,J-1))/2.
     6     +(H/2.)*(C2(2,J+1)+3.*C2(2,J))/4.*(VELH12-VEL(J))/(H/2.)
     7     +(HH/2.)*(C2(2,J-1)+3.*C2(2,J))/4.*(VEL(J)-VELHH12)/(HH/2.)
     8     +(H/2.)*(RXN(2,J+1)+3.*RXN(2,J))/4.
     9     +(HH/2.)*(RXN(2,J-1)+3.*RXN(2,J))/4.
      B(4,4)=DIFF(2)/H+DIFF(2)/HH-(H/2.)*3./4.*(VELH12-VEL(J))/(H/2.)
     1       -(HH/2.)*3./4.*(VEL(J)-VELHH12)/(HH/2.)
      D(4,4)=-DIFF(2)/H+(VEL(J)+VEL(J+1))/4.
     1       -(H/2.)/4.*(VELH12-VEL(J))/(H/2.)
      A(4,4)=-DIFF(2)/HH-(VEL(J)+VEL(J-1))/4.
     1       -(HH/2.)/4.*(VEL(J)-VELHH12)/(HH/2.)
      B(4,3)=+HH/2*omega*(3./4.)+H/2*omega*(3./4.)
      D(4,3)=+HH/2*omega*(1./4.)
      A(4,3)=+H/2*omega*(1./4.)
      B(4,8)=-(H/2.)*3./4.-(HH/2.)*(1./4.)
      D(4,8)=-(H/2.)*3./4.
      A(4,8)=-(HH/2.)*(1./4.)

C     For B+, real
      G(5)=-HH/2*omega*(C3(2,J+1)+3.*C3(2,J))/4.
     1     +H/2*omega*(C3(2,J-1)+3.*C3(2,J))/4.
     2     +DIFF(3)/H*(C3(1,J+1)-C3(1,J))
     3     -DIFF(3)/HH*(C3(1,J)-C3(1,J-1))
```

```
     4      -(VEL(J)+VEL(J+1))/2.*(C3(1,J+1)+C3(1,J))/2.
     5      +(VEL(J)+VEL(J-1))/2.*(C3(1,J)+C3(1,J-1))/2.
     6      +(H/2.)*(C3(1,J+1)+3.*C3(1,J))/4.*(VELH12-VEL(J))/(H/2.)
     7      +(HH/2.)*(C3(1,J-1)+3.*C3(1,J))/4.*(VEL(J)-VELHH12)/(HH/2.)
     8      +(H/2.)*(RXN(1,J+1)+3.*RXN(1,J))/4.
     9      +(HH/2.)*(RXN(1,J-1)+3.*RXN(1,J))/4.
      B(5,5)=DIFF(3)/H+DIFF(3)/HH-(H/2.)*3./4.*(VELH12-VEL(J))/(H/2.)
     1          -(HH/2.)*3./4.*(VEL(J)-VELHH12)/(HH/2.)
      D(5,5)=-DIFF(3)/H+(VEL(J)+VEL(J+1))/4.
     1          -(H/2.)/4.*(VELH12-VEL(J))/(H/2.)
      A(5,5)=-DIFF(3)/HH-(VEL(J)+VEL(J-1))/4.
     1          -(HH/2.)/4.*(VEL(J)-VELHH12)/(HH/2.)
      B(5,6)=-HH/2*omega*(3./4.)-H/2*omega*(3./4.)
      D(5,6)=-HH/2*omega*(1./4.)
      A(5,6)=-H/2*omega*(1./4.)
      B(5,7)=-(H/2.)*3./4.-(HH/2.)*(1./4.)
      D(5,7)=-(H/2.)*3./4.
      A(5,7)=-(HH/2.)*(1./4.)
C     For B+, imaginary
      G(6)=-HH/2*omega*(C3(1,J+1)+3.*C3(1,J))/4.
     1     -H/2*omega*(C3(1,J-1)+3.*C3(1,J))/4.
     2     +DIFF(3)/H*(C3(2,J+1)-C3(2,J))
     3     -DIFF(3)/HH*(C3(2,J)-C(2,J-1))
     4      -(VEL(J)+VEL(J+1))/2.*(C3(2,J+1)+C3(2,J))/2.
     5      +(VEL(J)+VEL(J-1))/2.*(C3(2,J)+C3(2,J-1))/2.
     6      +(H/2.)*(C3(2,J+1)+3.*C3(2,J))/4.*(VELH12-VEL(J))/(H/2.)
     7      +(HH/2.)*(C3(2,J-1)+3.*C3(2,J))/4.*(VEL(J)-VELHH12)/(HH/2.)
     8      +(H/2.)*(RXN(2,J+1)+3.*RXN(2,J))/4.
     9      +(HH/2.)*(RXN(2,J-1)+3.*RXN(2,J))/4.
      B(6,6)=DIFF(3)/H+DIFF(3)/HH-(H/2.)*3./4.*(VELH12-VEL(J))/(H/2.)
     1          -(HH/2.)*3./4.*(VEL(J)-VELHH12)/(HH/2.)
      D(6,6)=-DIFF(3)/H+(VEL(J)+VEL(J+1))/4.
     1          -(H/2.)/4.*(VELH12-VEL(J))/(H/2.)
      A(6,6)=-DIFF(3)/HH-(VEL(J)+VEL(J-1))/4.
     1          -(HH/2.)/4.*(VEL(J)-VELHH12)/(HH/2.)
      B(6,5)=+HH/2*omega*(3./4.)+H/2*omega*(3./4.)
      D(6,5)=+HH/2*omega*(1./4.)
      A(6,5)=+H/2*omega*(1./4.)
      B(6,8)=-(H/2.)*3./4.-(HH/2.)*(1./4.)
      D(6,8)=-(H/2.)*3./4.
      A(6,8)=-(HH/2.)*(1./4.)

C     For Reaction term
      G(7)=-RXN(1,J)+rateb*equilib*C1(1,J)-rateb*CONCSS(2,J)*C3(1,J)
     1       -rateb*CONCSS(3,J)*C2(1,J)
      B(7,1)=-rateb*equilib
      B(7,3)=rateb*CONCSS(3,J)
      B(7,5)=rateb*CONCSS(2,J)
      B(7,7)=+1.
      G(8)=-RXN(2,J)+rateb*equilib*C1(2,J)-rateb*CONCSS(2,J)*C3(2,J)
     1       -rateb*CONCSS(3,J)*C2(2,J)
      B(8,2)=-rateb*equilib
      B(8,4)=rateb*CONCSS(3,J)
      B(8,6)=rateb*CONCSS(2,J)
      B(8,8)=+1.
  212 WRITE(12,301) J, (G(K),K=1,N)
      RETURN
      END
```

Code D.16. Oscillating Convective Diffusion with Homogeneous Reaction Subroutine for the Electrode Boundary Condition

```
      SUBROUTINE INNER(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(8,8),B(8,8),C(8,100001),D(8,17),G(8),X(8,8),Y(8,8)
      COMMON/NST/ N, NJ
      COMMON/CON/ C1(2,100001),C2(2,100001),C3(2,100001),RXN(2,100001)
      COMMON/RTE/ rateb,equilib,H,EBIG,HH,KJ
      COMMON/BCI/ VEL1, FLUX, omega
      COMMON/CAR/ CONCSS(3,100001),CBULK(3),DIFF(3),Z(3),REF(3)
      COMMON/VAR/ RXNSS(100001),VELNEAR(100001),VELFAR(100001)
      COMMON/FRE/ CB(2010,100001), FREQ(100001),VEL(100001),FVEL(100001)

  301 FORMAT (5x,'J=' I5,  16E15.6)

C     For AB
      G(1)=omega*C1(2,J)
     1     +DIFF(1)*(C1(1,J+1)-2.*C1(1,J)+C1(1,J-1))/H**2.
     2     -VEL(J)*(C1(1,J+1)-C1(1,J-1))/(2.*H)
     3     -RXN(1,J)
      B(1,1)=2.*DIFF(1)/H**2.
      A(1,1)=-DIFF(1)/H**2.-VEL(J)/(2.*H)
      D(1,1)=-DIFF(1)/H**2.+VEL(J)/(2.*H)
      B(1,2)=-omega
      B(1,7)=+1.

      G(2)=-omega*C1(1,J)
     1     +DIFF(1)*(C1(2,J+1)-2.*C1(2,J)+C1(2,J-1))/H**2.
     2     -VEL(J)*(C1(2,J+1)-C1(2,J-1))/(2.*H)
     3     -RXN(2,J)
      B(2,2)=2.*DIFF(1)/H**2.
      A(2,2)=-DIFF(1)/H**2.-VEL(J)/(2.*H)
      D(2,2)=-DIFF(1)/H**2.+VEL(J)/(2.*H)
      B(2,1)=omega
      B(2,8)=+1.

C     For A-
      G(3)=omega*C2(2,J)+DIFF(2)*(C2(1,J+1)-2.*C2(1,J)+C2(1,J-1))/H**2.
     1     -VEL(J)*(C2(1,J+1)-C2(1,J-1))/(2.*H)+RXN(1,J)
      B(3,3)=2.*DIFF(2)/H**2.
      A(3,3)=-DIFF(2)/H**2.-VEL(J)/(2.*H)
      D(3,3)=-DIFF(2)/H**2.+VEL(J)/(2.*H)
      B(3,4)=-omega
      B(3,7)=-1.

      G(4)=-omega*C2(1,J)+DIFF(2)*(C2(2,J+1)-2.*C2(2,J)+C2(2,J-1))/H**2.
     1     -VEL(J)*(C2(2,J+1)-C2(2,J-1))/(2.*H)+RXN(2,J)
      B(4,4)=2.*DIFF(2)/H**2.
      A(4,4)=-DIFF(2)/H**2.-VEL(J)/(2.*H)
      D(4,4)=-DIFF(2)/H**2.+VEL(J)/(2.*H)
      B(4,3)=omega
      B(4,8)=-1.

C     For B+
      G(5)=omega*C3(2,J)-DIFF(3)*(C3(1,J+1)-2.*C3(1,J)+C3(1,J-1))/H**2.
     1     -VEL(J)*(C3(1,J+1)-C3(1,J-1))/(2.*H)+RXN(1,J)
```

```
55          B(5,5)=2.*DIFF(3)/H**2.
56          A(5,5)=-DIFF(3)/H**2.-VEL(J)/(2.*H)
57          D(5,5)=-DIFF(3)/H**2.+VEL(J)/(2.*H)
58          B(5,6)=-omega
59          B(5,7)=-1.
60
61          G(6)=-omega*C3(1,J)-DIFF(3)*(C3(2,J+1)-2.*C3(2,J)+C3(2,J-1))/H**2.
62         1      -VEL(J)*(C3(2,J+1)-C3(2,J-1))/(2.*H)+RXN(2,J)
63          B(6,6)=2.*DIFF(3)/H**2.
64          A(6,6)=-DIFF(3)/H**2.-VEL(J)/(2.*H)
65          D(6,6)=-DIFF(3)/H**2.+VEL(J)/(2.*H)
66          B(6,5)=omega
67          B(6,8)=-1.
68
69 C        For Reaction term
70          G(7)=-RXN(1,J)+rateb*equilib*C1(1,J)-rateb*CONCSS(2,J)*C3(1,J)
71         1         -rateb*CONCSS(3,J)*C2(1,J)
72          B(7,1)=-rateb*equilib
73          B(7,3)=rateb*CONCSS(3,J)
74          B(7,5)=rateb*CONCSS(2,J)
75          B(7,7)=+1.
76
77          G(8)=-RXN(2,J)+rateb*equilib*C1(2,J)-rateb*CONCSS(2,J)*C3(2,J)
78         1         -rateb*CONCSS(3,J)*C2(2,J)
79          B(8,2)=-rateb*equilib
80          B(8,4)=rateb*CONCSS(3,J)
81          B(8,6)=rateb*CONCSS(2,J)
82          B(8,8)=+1.
83
84          IF (J.EQ.KJ/2) THEN
85          WRITE(14,301) J, (G(K),K=1,N)
86          ELSE IF (J.EQ.(NJ-2)) THEN
87          WRITE(14,301) J, (G(K),K=1,N)
88          ELSE IF (J.EQ.(NJ-1)) THEN
89          WRITE(14,301) J, (G(K),K=1,N)
90          END IF
91
92          RETURN
93          END
```

Code D.17. Oscillating Convective Diffusion with Homogeneous Reaction Subroutine for the Electrode Boundary Condition

```
      SUBROUTINE BCNJ(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(8,8),B(8,8),C(8,100001),D(8,17),G(8),X(8,8),Y(8,8)
      COMMON/NST/ N, NJ
      COMMON/CON/ C1(2,100001),C2(2,100001),C3(2,100001),RXN(2,100001)
      COMMON/RTE/ rateb,equilib,H,EBIG,HH,KJ
      COMMON/BCI/ VEL1, FLUX, omega
      COMMON/CAR/ CONCSS(3,100001),CBULK(3),DIFF(3),Z(3),REF(3)
      COMMON/VAR/ RXNSS(100001),VELNEAR(100001),VELFAR(100001)
      COMMON/FRE/ CB(2010,100001), FREQ(100001),VEL(100001),FVEL(100001)

  301 FORMAT (5x,'J=' I5,  16E15.6)

C     For AB  non reacting species
   29 G(1)=C1(1,J)
      B(1,1)=-1.

      G(2)=C1(2,J)
      B(2,2)=-1.

C     For A+  non reacting species
      G(3)=C2(1,J)
      B(3,3)=-1.

      G(4)=C2(2,J)
      B(4,4)=-1.

C     For B- reacting species
      G(5)=C3(1,J)
      B(5,5)=-1.

      G(6)=C3(2,J)
      B(6,6)=-1.

C     For Reaction term
      G(7)=-RXN(1,J)+rateb*equilib*C1(1,J)-rateb*CONCSS(2,J)*C3(1,J)
     1       -rateb*CONCSS(3,J)*C2(1,J)
      B(7,1)=-rateb*equilib
      B(7,3)=rateb*CONCSS(3,J)
      B(7,5)=rateb*CONCSS(2,J)
      B(7,7)=+1.

      G(8)=-RXN(2,J)+rateb*equilib*C1(2,J)-rateb*CONCSS(2,J)*C3(2,J)
     1       -rateb*CONCSS(3,J)*C2(2,J)
      B(8,2)=-rateb*equilib
      B(8,4)=rateb*CONCSS(3,J)
      B(8,6)=rateb*CONCSS(2,J)
      B(8,8)=+1.

      WRITE(12,301) J, (G(K),K=1,N)

      RETURN
      END
```

Code D.18. Matlab code to create and plot polarization curve

```
%Inserting concentration data from Fortran
clc; close all;clear all;
format longE;

%Read the unsteady state data at each frequency
B = dlmread('cdh_B_out.txt');

%Read constant values used in the Fortran code
M = dlmread('cdh_values_out.txt');

N=M(1);
NJ=M(2);
KJ=M(3);
H=M(4);
HH=M(5);
V=M(6);
AKB=M(7);
BB=M(8);
RTB=M(9);
DiffB=M(10);
delta=M(11);
rot=M(12);
anu=M(13);
nf=M(14);

%Read frequency points, Kw=omega, KK=K
K = dlmread('k_values_out.txt');
K=K';
for n=1:nf
    Kw(n)=K(n+nf);
    KK(n)=K(n);
end

deltan=gamma(4/3)*delta;

%Read the steady state values for CB
Bss1 = dlmread('cdh_out.txt');
Bss=Bss1(:,3);

%Other constants
F=96487;

%Create y values for plotting
y=zeros(NJ,1);

far=HH*(KJ-1);
y1=0:HH:far;

far1=H*(NJ-KJ);
y2=y1(KJ):H:y1(KJ)+far1;

for i=1:KJ-1
    y(i)=y1(i);
end
for i=KJ:NJ
    y(i)=y2(i-KJ+1);
```

```
end

%Create complex numbers from unsteady state data
CB=zeros(NJ,nf);
for n=1:nf
    for i=1:NJ
    CB(i,n)=complex(B(i,2*n-1),B(i,2*n));
    end
end

%Calculate the impedance
Zdfront=(RTB*AKB*exp(-BB*V))/(F*DiffB);

Zd=zeros(1,nf);
for i=1:nf
Zd(i)=(-CB(1,i)/((-CB(3,i)+4*CB(2,i)-3*CB(1,i))/(2*H)))/deltan;
end

figure(1)
plot(real(Zd),-imag(Zd),'-k');hold on; axis equal;
title('Nyquist plot');
xlabel('Real part of Impedance');
ylabel('Imaginary part of Impedance');
real=real(Zd);
imag=imag(Zd);

figure(2)
loglog(Kw,-imag,'-k');hold on;

figure(3)
loglog(Kw,real,'-k');hold on;

impedance=zeros(nf,2);
impedance(:,1)=real';
impedance(:,2)=imag';
```

APPENDIX E
CODES FOR CONTINUOUS GLUCOSE MONITOR

This appendix contains the different FORTRAN codes that produced the results for the solution of the convective-diffusion equation with homogeneous reaction. This appendix also has the Matlab codes to plot the steady-state results, to plot a polarization curve from the steady state results and a code to create the impedance from the results of the oscillating FORTRAN code.

E.1 Input files for the Continuous Glucose Monitor

The following codes are the input files for the continuous glucose monitor. The input code has the number of species being solved, the total number of points, the number of points until the first coupler, the number of points until the second coupler, the distance of the reaction region in cm, the distance of the inner layer in cm, and the distance in the GLM layer in cm. The rate constants in the input file are the equilibrium rates of two reactions and the forward rate of reaction for two reversible reactions and two irreversible reactions and the rate constant for the heterogeneous reaction of the reacting species, and the tafel kinetics value for the heterogeneous reaction. The input file includes the error allowed for the BIG values, which is discussed in section 2.2.2. And the end of the input file has the specific values to describe each species in the system, including diffusion coefficients in cm^2/s, the charge, a character name, and the concentration value in the bulk in mol/cm^3.

Code E.1. Input file for the Continuous Glucose Monitor Code

```
12
40001
39001
29001
0.0004
0.0003
0.0015
0.8
0.42
0.169
0.32
0.11
0.0176
1.E8
10.
1.E9
1.E8
10.
1.E9
1.
37.42
1.E-14
7.2E-6 GL 5.55075E-6
0. GOx .5E-3
7.2E-6 GA 1.E-20
0. GOx2 .5E-3
2.46E-5 O2 3.125E-9
1.83E-5 H2O2 1.E-20
0. CX-GOX2 .5E-3
0. CX-GOX .5E-3

C     line 1 is the number of species
C     line 2 is the number of points, NJ
C     line 3 is the point where the domains split, value of IJ
C     line 4 is the point where the reaction layer is, value of KJ
C     line 5 is the distance of the inner reaction later in cm (1um)
C     line 6 is the distance of the inner GOx layer in cm (6 um)
C     line 7 is the distance of the outer GLM layer in cm (15 um)
C     line 8 is the porosity factor of the inner layer and reaction layer
C     line 9 is the porosity factor of the outer layer for small species
C     line 10 is the porosity factor of the outer layer for large species
C     line 11 is the solubility coefficient of H2O2
C     line 12 is the solubility coefficient of O2
c     line 13 is the solubility coefficient of Glucose
C     line 14 is the ratef1 of rxn1, mol/cm^3
C     line 15 is the equilib1 of rxn1, cm^3/(mol*s)
C     line 16 is the ratef2 of rxn2, mol/cm^3
C     line 17 is the ratef3 of rxn3, mol/cm^3
C     line 18 is the equilib3 of rxn3, cm^3/(mol*s)
C     line 19 is the ratef3 of rxn4, mol/cm^3
C     line 20 is potential
C     line 21 is the rate constant (K) for the flux of the reacting species, A
    /cm2 cm3/mol
C     line 22 is the tafel b value for the flux of the reacting species
C     line 23 is the error allowed for the BIGs
C     lines 24-31 specify values used to describe each species in the system
```

E.2 Steady-State Continuous Glucose Monitor Code

This section contains the steady-state FORTRAN codes used to solve the continuous glucose monitor. The mathematical workup for these codes are in Chapter 5. The FORTRAN codes are followed by a code. The Matlab code takes the output from the steady-state FORTRAN code and plots the data.

Code E.2. Steady-State Continuous Glucose Monitor Main Program

```
C     Convective Diffusion Equation with Homogeneous Reaction
C     Enzyme kinetics added
C     8 species system
C     SPECIES 1 = glucose, SPECIES 2 = GOx-FAD, SPECIES 3 = Gluconic acid
C     SPECIES 4 = GOx-FADH2, SPECIES 5 = O2, SPECIES 6 = H2O2
C     SPECIES 7 = GOx-FADH2-GA, SPECIES 8 = GOx-FAD-H2O2
C     Species 6 is the reacting species
C     This is the steady state solution only
C     It should be ran prior to cdhgox_os.for
C     The input file is the same for both
C     This version of the code is reversible normal kinetics for reactions 1
   and 3
C     Reactions 2 and 4 are irreversible

      PROGRAM CONVDIFF
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(12,12),B(12,12),C(12,80001),D(12,25),G(12),X(12,12)
     1   ,Y(12,12)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(8,80001),RXN(4,80001),DIFF(8),H,EBIG,HH,IJ
      COMMON/VARR/ HHH, KJ
      COMMON/POR/ POR1,POR2,PORGLU
      COMMON/BCI/ FLUX
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT
      COMMON/EXTRA/ REF(8)
      CHARACTER REF*8

  102 FORMAT (/30H THE NEXT RUN DID NOT CONVERGE)
  103 FORMAT ('Error=',E16.6/(1X,'Species=',A6,2X,'C at Electrode=',
     1 E12.5,2X,'C at Bulk=',E12.5))
  300 FORMAT (18x,'Glucose'14x,'GOx',14x,'GA',14x,'GOx2',14x,'O2',14x,
     1     'H2O2',14x,'CX-GOx2',14x,'CX-GOx',14x,'RXN1',14x,'RXN2',14x,
     1     'RXN3',14x,'RXN4')
  301 FORMAT (5x,'J=' I5,  12E18.9)
  334 FORMAT (12(E25.15,5X))
  302 FORMAT ('Iteration='I4)

      OPEN(UNIT=13, FILE='cdhgox_out.txt')
      CLOSE(UNIT=13, STATUS='DELETE')
      OPEN(UNIT=13, FILE='cdhgox_out.txt')

      OPEN(12,FILE='cdhgox_G_out.txt')
      CLOSE(12, STATUS='DELETE')
      OPEN(12, FILE='cdhgox_G_out.txt')
      WRITE(12,300)

      open(14,file='cdhgox_in.txt',status='old')
  106 FORMAT (I2/I7/I7/E15.4/E15.4/E15.4/E15.4/E15.4/E15.4/E15.4/E15.4
     1    /E15.4/E15.4/E15.4/E15.4/E15.4/E15.4)
      print *, 'does this work'
      read(14,*) N,NJ,IJ,KJ,Y1,Y2,Y3,POR1,POR2,POR3,PARH2O2,PAR02,
     1     PARGLUCOSE,ratef1,equilib1,ratef2,ratef3,equilib3,ratef4,
     2     AKB,BB,EBIG
      read(14,*) (DIFF(I),REF(I),CBULK(I),I=1,(N-4))
      POR1=POR1**(1.5)
```

```
      POR2=POR2**(1.5)
      PORGLU=POR3**(1.5)
      open(16,file='pot_in.txt',status='old')
      read(16,*) V

      open(16,file='O2_in.txt',status='old')
  305 FORMAT (E25.5)
      read(16,305) CBULK(5)

C     Constants
      F=96487.

c     THIS IS SPACING FOR OUTER LAYER, BCNJ
      H=Y3/(NJ-IJ)
      PRINT *, 'H=', H
      PRINT *, 'Y3=', Y3
      PRINT *, 'NJ-IJ=', NJ-IJ

c     THIS IS SPACING FOR INNER LAYER
      HH=(Y2)/(IJ-KJ)
      PRINT *, 'HH=', HH
      PRINT *, 'Y2=', Y2
      PRINT *, 'IJ-KJ=', IJ-KJ

c     THIS IS SPACING FOR REACTION LAYER
      HHH=(Y1)/(KJ-1)
      PRINT *, 'Y1=', Y1
      PRINT *, 'KJ-1=', KJ-1
      PRINT *, 'HHH=', HHH

      OPEN(15,FILE='cdhgox_ssvalues_out.txt')
      CLOSE(15, STATUS='DELETE')
      OPEN(15, FILE='cdhgox_ssvalues_out.txt')
  337 FORMAT (I2/I7/I7/I7/E25.15/E25.15/E25.15/E15.8/E15.8/E15.4/E15.4
     1    /E15.4/E15.4/E25.15)
      WRITE (15,337) N,NJ,IJ,KJ,H,HH,HHH,DIFF(6),AKB,BB,V,POR1

C     Create flux of the reacting species constants
      FLUX=-AKB*exp(BB*V)/F/2.
      PRINT *, 'FLUX=', FLUX

c     THIS IS THE MAIN PART OF THE PROGRAM
      DO 21 J=1,NJ
      RXN(1,J)=0.00001
      RXN(2,J)=0.00001
      RXN(3,J)=0.00001
      RXN(4,J)=0.00001
      DO 21 I=1,N-4
      C(I,J)=0.0
   21 CONC(I,J)=CBULK(I)
      JCOUNT=0
      TOL=1.E-10*N*NJ/1.E8
      PRINT *, 'TOL=', TOL
   22 JCOUNT=JCOUNT+1
      AMP=0.0
      J=0
      DO 23 I=1,N
      DO 23 K=1,N
```

```
      Y(I,K)=0.0
   23 X(I,K)=0.0
   24 J=J+1
      DO 25   I=1,N
      G(I)=0.0
      DO 25 K=1,N
      A(I,K)=0.0
      B(I,K)=0.0
   25 D(I,K)=0.0

      IF (J.EQ.1) CALL BC1(J)
      IF (J.GT.1 .AND. J.LT.KJ) CALL REACTION(J)
      IF (J.EQ.KJ) CALL COUPLER1(J)
      IF (J.GT.KJ .AND. J.LT.IJ) CALL INNER(J)
      IF (J.EQ.IJ) CALL COUPLER2(J)
      IF (J.GT.IJ .AND. J.LT.NJ) CALL OUTER(J)
      IF (J.EQ.NJ)  CALL BCNJ(J)
      CALL BAND(J)

      AMP=AMP+DABS(G(1))+DABS(G(2))+DABS(G(3))+DABS(G(4))+DABS(G(5))
     1    +DABS(G(6))+DABS(G(7))+DABS(G(8))+DABS(G(9))+DABS(G(10))
     2    +DABS(G(11))+DABS(G(12))

      IF (J.LT.NJ)  GO TO 24

      PRINT *, 'ERROR=', AMP

      DO 16 K=1,NJ
      RXN(1,K)=RXN(1,K)+C(9,K)
      RXN(2,K)=RXN(2,K)+C(10,K)
      RXN(3,K)=RXN(3,K)+C(11,K)
      RXN(4,K)=RXN(4,K)+C(12,K)
      DO 16 I=1,N-4
      IF (C(I,K).LT.-0.999*CONC(I,K)) C(I,K)=-0.999*CONC(I,K)
      IF (C(I,K).GT. 999.*CONC(I,K)) C(I,K)= 999.*CONC(I,K)
      CONC(I,K)=CONC(I,K)+C(I,K)
   16 CONTINUE

      WRITE(12,302) (JCOUNT)

c     If the error is less then the tolerance, finish program
      IF (DABS(AMP).LT.DABS(TOL)) GO TO  15

c     If the error is greater then tolerance, do another iteration
   33 IF (JCOUNT.LE.19) GO TO 22
      print 102

   15 PRINT 103, AMP,(REF(I),CONC(I,1),CONC(I,NJ),I=1,N-4)

      PRINT *,'JCOUNT=',JCOUNT

      WRITE(13,334) (CONC(1,J),CONC(2,J),CONC(3,J),CONC(4,J),CONC(5,J),
     1     CONC(6,J),CONC(7,J),CONC(8,J),RXN(1,J),RXN(2,J),RXN(3,J),
     2     RXN(4,J),J=1,NJ)

C       WRITE(13,334) (CONC(1,J),CONC(2,J),CONC(3,J),CONC(4,J),CONC(5,J),
C      1     CONC(6,J),CONC(7,J),CONC(8,J),J=1,NJ)

```

```
172         END PROGRAM CONVDIFF
```

Code E.3. Steady-State Continuous Glucose Monitor for the Electrode Boundary Condition

```
      SUBROUTINE BC1(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(12,12),B(12,12),C(12,80001),D(12,25),G(12),X(12,12)
     1   ,Y(12,12)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(8,80001),RXN(4,80001),DIFF(8),H,EBIG,HH,IJ
      COMMON/VARR/ HHH, KJ
      COMMON/POR/ POR1,POR2,PORGLU
      COMMON/BCI/ FLUX
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT

  301 FORMAT (5x,'J=' I5,  12E18.9)

C     For Glucose, being consumed only
      G(1)=2.*POR1*DIFF(1)*(CONC(1,J+1)-CONC(1,J))/HHH**2.
     2    -(3.*RXN(1,J)+RXN(1,J+1))/4.
      B(1,1)=2.*POR1*DIFF(1)/HHH**2.
      D(1,1)=-2.*POR1*DIFF(1)/HHH**2.
      B(1,9)=+0.75
      D(1,9)=+0.25

      BIG=ABS(2.*POR1*DIFF(1)*(CONC(1,J+1))/HHH**2.)
      BIG2=ABS(2.*POR1*DIFF(1)*(-CONC(1,J))/HHH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      IF (ABS(-3.*RXN(1,J)/4.).GT.BIG) BIG=ABS(-3.*RXN(1,J)/4.)
      IF (ABS(-RXN(1,J+1)/4.).GT.BIG) BIG=ABS(-RXN(1,J+1)/4.)
      IF (ABS(G(1)).LT.BIG*EBIG) G(1)=0

C     For GOx, enzyme
      G(2)=-RXN(1,J)+RXN(4,J)
      B(2,9)=+1
      B(2,12)=-1

      IF (ABS(RXN(1,J)).GT.BIG) BIG=ABS(RXN(1,J))
      IF (ABS(RXN(4,J)).GT.BIG) BIG=ABS(RXN(4,J))
      IF (ABS(G(2)).LT.BIG*EBIG) G(2)=0

C     For Gluconic Acid, being produced only
      G(3)=2.*POR1*DIFF(3)*(CONC(3,J+1)-CONC(3,J))/HHH**2.
     2    +(3*RXN(2,J)+RXN(2,J+1))/4.
      B(3,3)=2.*POR1*DIFF(3)/HHH**2.
      D(3,3)=-2.*POR1*DIFF(3)/HHH**2.
      B(3,10)=-0.75
      D(3,10)=-0.25

      BIG=ABS(2.*POR1*DIFF(3)*(CONC(3,J+1))/HHH**2.)
      BIG2=ABS(2.*POR1*DIFF(3)*(-CONC(3,J))/HHH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      IF (ABS(3.*RXN(2,J)/4.).GT.BIG) BIG=ABS(3.*RXN(2,J)/4.)
      IF (ABS(RXN(2,J+1)/4.).GT.BIG) BIG=ABS(RXN(2,J+1)/4.)
      IF (ABS(G(3)).LT.BIG*EBIG) G(3)=0

C     For GOx2, enzyme
```

```
      G(4)=CBULK(2)+CBULK(4)+CBULK(7)+CBULK(8)-CONC(2,J)-CONC(4,J)
     1    -CONC(7,J)-CONC(8,J)
      B(4,2)=+1.
      B(4,4)=+1.
      B(4,7)=+1.
      B(4,8)=+1.

      BIG=ABS(CBULK(2))
      IF (ABS(CBULK(4)).GT.BIG) BIG=ABS(CBULK(4))
      IF (ABS(CBULK(7)).GT.BIG) BIG=ABS(CBULK(7))
      IF (ABS(CBULK(8)).GT.BIG) BIG=ABS(CBULK(8))
      IF (ABS(CONC(2,J)).GT.BIG) BIG=ABS(CONC(2,J))
      IF (ABS(CONC(4,J)).GT.BIG) BIG=ABS(CONC(4,J))
      IF (ABS(CONC(7,J)).GT.BIG) BIG=ABS(CONC(7,J))
      IF (ABS(CONC(8,J)).GT.BIG) BIG=ABS(CONC(8,J))
      IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0

C     For O2, being consumed only
      G(5)=2.*POR1*DIFF(5)*(CONC(5,J+1)-CONC(5,J))/HHH**2.
     1      -FLUX*CONC(6,J)/(HHH/2.)
     2      -(3.*RXN(3,J)+RXN(3,J+1))/4.
      B(5,5)=2.*POR1*DIFF(5)/HHH**2.
      D(5,5)=-2.*POR1*DIFF(5)/HHH**2.
      B(5,11)=+0.75
      D(5,11)=+0.25
      B(5,6)=+FLUX/(HHH/2.)

      BIG=ABS(2.*POR1*DIFF(5)*(CONC(5,J+1))/HHH**2.)
      BIG2=ABS(2.*POR1*DIFF(5)*(-CONC(5,J))/HHH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      IF (ABS(-3.*RXN(3,J)/4.).GT.BIG) BIG=ABS(-3.*RXN(3,J)/4.)
      IF (ABS(-RXN(3,J+1)/4.).GT.BIG) BIG=ABS(-RXN(3,J+1)/4.)
      IF (ABS(G(5)).LT.BIG*EBIG) G(5)=0

C     For H2O2, reacting species
      G(6)=2.*POR1*DIFF(6)*(CONC(6,J+1)-CONC(6,J))/HHH**2.
     2     +FLUX*CONC(6,J)/(HHH/2.)
     3     +(3.*RXN(4,J)+RXN(4,J+1))/4.
      B(6,6)=2.*POR1*DIFF(6)/HHH**2-FLUX/(HHH/2.)
      D(6,6)=-2.*POR1*DIFF(6)/HHH**2
      B(6,12)=-0.75
      D(6,12)=-0.25

      BIG=ABS(2.*POR1*DIFF(6)*(CONC(6,J+1))/HHH**2.)
      BIG2=ABS(2.*POR1*DIFF(6)*(-CONC(6,J))/HHH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      IF (ABS(3.*RXN(4,J)/4.).GT.BIG) BIG=ABS(3.*RXN(4,J)/4.)
      IF (ABS(RXN(4,J+1)/4.).GT.BIG) BIG=ABS(RXN(4,J+1)/4.)
      IF (ABS(G(6)).LT.BIG*EBIG) G(6)=0

C     For CX-GOx2, enzyme
      G(7)=RXN(1,J)-RXN(2,J)
      B(7,9)=-1.
      B(7,10)=1.

      IF (ABS(RXN(1,J)).GT.BIG) BIG=ABS(RXN(1,J))
      IF (ABS(RXN(2,J)).GT.BIG) BIG=ABS(RXN(2,J))
      IF (ABS(G(7)).LT.BIG*EBIG) G(7)=0
```

```
113
114 C       For CX-GOx, enzyme
115         G(8)=-RXN(3,J)-RXN(4,J)
116         B(8,11)=-1.
117         B(8,12)=1.
118
119         IF (ABS(RXN(3,J)).GT.BIG) BIG=ABS(RXN(3,J))
120         IF (ABS(RXN(4,J)).GT.BIG) BIG=ABS(RXN(4,J))
121         IF (ABS(G(8)).LT.BIG*EBIG) G(8)=0
122
123 C       REACTION1
124    214  G(9)=-RXN(1,J)+ratef1*(CONC(1,J)*CONC(2,J)-(CONC(7,J)/equilib1))
125         B(9,1)=-ratef1*CONC(2,J)
126         B(9,2)=-ratef1*CONC(1,J)
127         B(9,7)=ratef1/equilib1
128         B(9,9)=+1.
129
130         BIG=ABS(RXN(1,J))
131         BIG2=ABS(ratef1*CONC(1,J)*CONC(2,J))
132         IF (BIG2.GT.BIG) BIG=BIG2
133         BIG3=ABS(ratef1*(CONC(7,J)/equilib1))
134         IF (BIG3.GT.BIG) BIG=BIG3
135         IF (ABS(G(9)).LT.BIG*EBIG) G(9)=0
136
137 C       REACTION2
138    215  G(10)=-RXN(2,J)+ratef2*CONC(7,J)
139         B(10,7)=-ratef2
140         B(10,10)=+1.
141
142         BIG=ABS(RXN(2,J))
143         BIG2=ABS(ratef2*CONC(7,J))
144         IF (BIG2.GT.BIG) BIG=BIG2
145         IF (ABS(G(10)).LT.BIG*EBIG) G(10)=0
146
147 C       REACTION3
148    216  G(11)=-RXN(3,J)+ratef3*(CONC(4,J)*CONC(5,J)-(CONC(8,J)/equilib3))
149         B(11,4)=-ratef3*CONC(5,J)
150         B(11,5)=-ratef3*CONC(4,J)
151         B(11,8)=ratef3/equilib3
152         B(11,11)=+1.
153
154         BIG=ABS(RXN(3,J))
155         BIG2=ABS(ratef3*CONC(4,J)*CONC(5,J))
156         IF (BIG2.GT.BIG) BIG=BIG2
157         BIG3=ABS(ratef3*(CONC(8,J)/equilib3))
158         IF (BIG3.GT.BIG) BIG=BIG3
159         IF (ABS(G(11)).LT.BIG*EBIG) G(11)=0
160
161 C       REACTION4
162    217  G(12)=-RXN(4,J)+ratef4*CONC(8,J)
163         B(12,8)=-ratef4
164         B(12,12)=+1.
165         BIG=ABS(RXN(4,J))
166         BIG2=ABS(ratef4*CONC(8,J))
167         IF (BIG2.GT.BIG) BIG=BIG2
168         IF (ABS(G(12)).LT.BIG*EBIG) G(12)=0
169    212  WRITE(12,301) J, (G(K),K=1,N)
170         RETURN
```

```
171         END
```

Code E.4. Steady-State Continuous Glucose Monitor Subroutine for the Reaction Region

```
      SUBROUTINE REACTION(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(12,12),B(12,12),C(12,80001),D(12,25),G(12),X(12,12)
     1   ,Y(12,12)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(8,80001),RXN(4,80001),DIFF(8),H,EBIG,HH,IJ
      COMMON/VARR/ HHH, KJ
      COMMON/POR/ POR1,POR2,PORGLU
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT

  301 FORMAT (5x,'J=' I5,  12E18.9)

C     For Glucose, being consumed only
      G(1)=POR1*DIFF(1)*(CONC(1,J+1)-2.*CONC(1,J)+CONC(1,J-1))/HHH**2.
     2    -RXN(1,J)
      B(1,1)=2.*POR1*DIFF(1)/HHH**2.
      D(1,1)=-POR1*DIFF(1)/HHH**2.
      A(1,1)=-POR1*DIFF(1)/HHH**2.
      B(1,9)=+1.

      BIG=ABS(POR1*DIFF(1)*(CONC(1,J+1))/HHH**2.)
      BIG2=ABS(POR1*DIFF(1)*(-2.*CONC(1,J))/HHH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(POR1*DIFF(1)*(CONC(1,J-1))/HHH**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(-RXN(1,J)).GT.BIG) BIG=ABS(-RXN(1,J))
      IF (ABS(G(1)).LT.BIG*EBIG) G(1)=0

C     For GOx, enzyme
      G(2)=-RXN(1,J)+RXN(4,J)
      B(2,9)=+1
      B(2,12)=-1

      IF (ABS(RXN(1,J)).GT.BIG) BIG=ABS(RXN(1,J))
      IF (ABS(RXN(4,J)).GT.BIG) BIG=ABS(RXN(4,J))
      IF (ABS(G(2)).LT.BIG*EBIG) G(2)=0

C     For Gluconic Acid, being produced only
      G(3)=POR1*DIFF(3)*(CONC(3,J+1)-2.*CONC(3,J)+CONC(3,J-1))/HHH**2.
     2    +RXN(2,J)
      B(3,3)=2.*POR1*DIFF(3)/HHH**2.
      D(3,3)=-POR1*DIFF(3)/HHH**2.
      A(3,3)=-POR1*DIFF(3)/HHH**2.
      B(3,10)=-1.

      BIG=ABS(POR1*DIFF(3)*(CONC(3,J+1))/HHH**2.)
      BIG2=ABS(POR1*DIFF(3)*(-2.*CONC(3,J))/HHH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(POR1*DIFF(3)*(CONC(3,J-1))/HHH**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(RXN(2,J)).GT.BIG) BIG=ABS(RXN(2,J))
      IF (ABS(G(3)).LT.BIG*EBIG) G(3)=0

C     For GOx2, enzyme
      G(4)=CBULK(2)+CBULK(4)+CBULK(7)+CBULK(8)-CONC(2,J)-CONC(4,J)
```

```
     1    -CONC(7,J)-CONC(8,J)
      B(4,2)=+1.
      B(4,4)=+1.
      B(4,7)=+1.
      B(4,8)=+1.

      BIG=ABS(CBULK(2))
      IF (ABS(CBULK(4)).GT.BIG) BIG=ABS(CBULK(4))
      IF (ABS(CBULK(7)).GT.BIG) BIG=ABS(CBULK(7))
      IF (ABS(CBULK(8)).GT.BIG) BIG=ABS(CBULK(8))
      IF (ABS(CONC(2,J)).GT.BIG) BIG=ABS(CONC(2,J))
      IF (ABS(CONC(4,J)).GT.BIG) BIG=ABS(CONC(4,J))
      IF (ABS(CONC(7,J)).GT.BIG) BIG=ABS(CONC(7,J))
      IF (ABS(CONC(8,J)).GT.BIG) BIG=ABS(CONC(8,J))
      IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0

C     For O2, being consumed only
      G(5)=POR1*DIFF(5)*(CONC(5,J+1)-2.*CONC(5,J)+CONC(5,J-1))/HHH**2.
     2    -RXN(3,J)
      B(5,5)=2.*POR1*DIFF(5)/HHH**2.
      D(5,5)=-POR1*DIFF(5)/HHH**2.
      A(5,5)=-POR1*DIFF(5)/HHH**2.
      B(5,11)=+1.

      BIG=ABS(POR1*DIFF(5)*(CONC(5,J+1))/HHH**2.)
      BIG2=ABS(POR1*DIFF(5)*(-2.*CONC(5,J))/HHH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(POR1*DIFF(5)*(CONC(5,J-1))/HHH**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(-RXN(3,J)).GT.BIG) BIG=ABS(-RXN(3,J))
      IF (ABS(G(5)).LT.BIG*EBIG) G(5)=0

C     For H2O2, reacting species
      G(6)=POR1*DIFF(6)*(CONC(6,J+1)-2.*CONC(6,J)+CONC(6,J-1))/HHH**2.
     2    +RXN(4,J)
      B(6,6)=2.*POR1*DIFF(6)/HHH**2.
      D(6,6)=-POR1*DIFF(6)/HHH**2.
      A(6,6)=-POR1*DIFF(6)/HHH**2.
      B(6,12)=-1.

      BIG=ABS(POR1*DIFF(6)*(CONC(6,J+1))/HHH**2.)
      BIG2=ABS(POR1*DIFF(6)*(-2.*CONC(6,J))/HHH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(POR1*DIFF(6)*(CONC(6,J-1))/HHH**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(RXN(4,J)).GT.BIG) BIG=ABS(RXN(4,J))
      IF (ABS(G(6)).LT.BIG*EBIG) G(6)=0

C     For CX-GOx2, enzyme
      G(7)=RXN(1,J)-RXN(2,J)
      B(7,9)=-1.
      B(7,10)=1.

      IF (ABS(RXN(1,J)).GT.BIG) BIG=ABS(RXN(1,J))
      IF (ABS(RXN(2,J)).GT.BIG) BIG=ABS(RXN(2,J))
      IF (ABS(G(7)).LT.BIG*EBIG) G(7)=0

C     For CX-GOx, enzyme
```

```
115         G(8)=RXN(3,J)-RXN(4,J)
116         B(8,11)=-1.
117         B(8,12)=1.
118
119         IF (ABS(RXN(3,J)).GT.BIG) BIG=ABS(RXN(3,J))
120         IF (ABS(RXN(4,J)).GT.BIG) BIG=ABS(RXN(4,J))
121         IF (ABS(G(8)).LT.BIG*EBIG) G(8)=0
122
123
124 C       REACTION1
125    214  G(9)=-RXN(1,J)+ratef1*(CONC(1,J)*CONC(2,J)-(CONC(7,J)/equilib1))
126         B(9,1)=-ratef1*CONC(2,J)
127         B(9,2)=-ratef1*CONC(1,J)
128         B(9,7)=ratef1/equilib1
129         B(9,9)=+1.
130
131         BIG=ABS(RXN(1,J))
132         BIG2=ABS(ratef1*CONC(1,J)*CONC(2,J))
133         IF (BIG2.GT.BIG) BIG=BIG2
134         BIG3=ABS(ratef1*(CONC(7,J)/equilib1))
135         IF (BIG3.GT.BIG) BIG=BIG3
136         IF (ABS(G(9)).LT.BIG*EBIG) G(9)=0
137
138 C       REACTION2
139    215  G(10)=-RXN(2,J)+ratef2*CONC(7,J)
140         B(10,7)=-ratef2
141         B(10,10)=+1.
142
143         BIG=ABS(RXN(2,J))
144         BIG2=ABS(ratef2*CONC(7,J))
145         IF (BIG2.GT.BIG) BIG=BIG2
146         IF (ABS(G(10)).LT.BIG*EBIG) G(10)=0
147
148 C       REACTION3
149    216  G(11)=-RXN(3,J)+ratef3*(CONC(4,J)*CONC(5,J)-(CONC(8,J)/equilib3))
150         B(11,4)=-ratef3*CONC(5,J)
151         B(11,5)=-ratef3*CONC(4,J)
152         B(11,8)=ratef3/equilib3
153         B(11,11)=+1.
154
155         BIG=ABS(RXN(3,J))
156         BIG2=ABS(ratef3*CONC(4,J)*CONC(5,J))
157         IF (BIG2.GT.BIG) BIG=BIG2
158         BIG3=ABS(ratef3*(CONC(8,J)/equilib3))
159         IF (BIG3.GT.BIG) BIG=BIG3
160         IF (ABS(G(11)).LT.BIG*EBIG) G(11)=0
161
162 C       REACTION4
163    217  G(12)=-RXN(4,J)+ratef4*CONC(8,J)
164         B(12,8)=-ratef4
165         B(12,12)=+1.
166
167         BIG=ABS(RXN(4,J))
168         BIG2=ABS(ratef4*CONC(8,J))
169         IF (BIG2.GT.BIG) BIG=BIG2
170         IF (ABS(G(12)).LT.BIG*EBIG) G(12)=0
171
172 c       SAVE G OUT DATA
```

```
  212 DO 11 I=2,13
   11 If (I.EQ.J) WRITE(12,301) J, (G(K),K=1,N)
      IF (J.EQ.KJ/2) THEN
      WRITE(12,301) J, (G(K),K=1,N)
      ELSE IF (J.EQ.(KJ-1)) THEN
      WRITE(12,301) J, (G(K),K=1,N)
      ELSE IF (J.EQ.(KJ-2)) THEN
      WRITE(12,301) J, (G(K),K=1,N)
      ELSE IF (J.EQ.(KJ-3)) THEN
      WRITE(12,301) J, (G(K),K=1,N)
      END IF

      RETURN
      END
```

Code E.5. Steady-State Continuous Glucose Monitor Subroutine for the First Coupler

```
      SUBROUTINE COUPLER1(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(12,12),B(12,12),C(12,80001),D(12,25),G(12),X(12,12)
     1   ,Y(12,12)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(8,80001),RXN(4,80001),DIFF(8),H,EBIG,HH,IJ
      COMMON/VARR/ HHH, KJ
      COMMON/POR/ POR1,POR2,PORGLU
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT

C       DIMENSION COEFF1, COEFF3, COEFF5, COEFF6

  301 FORMAT (5x,'J=' I5,  12E18.9)

      COEFF1HH=POR1*DIFF(1)/(HH)
      COEFF1HHH=POR1*DIFF(1)/(HHH)
      COEFF3HH=POR1*DIFF(3)/(HH)
      COEFF3HHH=POR1*DIFF(3)/(HHH)
      COEFF5HH=POR1*DIFF(5)/(HH)
      COEFF5HHH=POR1*DIFF(5)/(HHH)
      COEFF6HH=POR1*DIFF(6)/(HH)
      COEFF6HHH=POR1*DIFF(6)/(HHH)

C     For Glucose, being consumed only
      G(1)=COEFF1HH*(CONC(1,KJ+1)-CONC(1,KJ))
     1     -COEFF1HHH*(CONC(1,KJ)-CONC(1,KJ-1))
     2     -(HH/2.)*(RXN(1,J+1)+3.*RXN(1,J))/4.
     3     -(HHH/2.)*(RXN(1,J-1)+3.*RXN(1,J))/4.
      B(1,1)=COEFF1HH+COEFF1HHH
      D(1,1)=-COEFF1HH
      A(1,1)=-COEFF1HHH
      B(1,9)=+(HH/2.)*(3./4.)+(HHH/2.)*(3./4.)
      D(1,9)=+(HH/2.)*(1./4.)
      A(1,9)=+(HHH/2.)*(1./4.)

      BIG=ABS(COEFF1HH*CONC(1,KJ+1))
      BIG2=ABS(COEFF1HH*CONC(1,KJ))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(-COEFF1HHH*CONC(1,KJ))
      IF (BIG3.GT.BIG) BIG=BIG3
      BIG4=ABS(-COEFF1HHH*CONC(1,KJ-1))
      IF (BIG4.GT.BIG) BIG=BIG4
      BIG5=ABS((HH/2.)*(RXN(1,J+1)/4.))
      IF (BIG5.GT.BIG) BIG=BIG5
      BIG6=ABS((HH/2.)*(3.*RXN(1,J))/4.)
      IF (BIG6.GT.BIG) BIG=BIG6
      BIG7=ABS((HHH/2.)*(RXN(1,J-1)/4.))
      IF (BIG7.GT.BIG) BIG=BIG7
      BIG8=ABS((HHH/2.)*(3.*RXN(1,J))/4.)
      IF (BIG8.GT.BIG) BIG=BIG8
      IF (ABS(G(1)).LT.BIG*EBIG) G(1)=0

C     For GOx, enzyme
      G(2)=-RXN(1,KJ)+RXN(4,KJ)
      B(2,9)=+1.
```

```
      B(2,12)=-1.

      BIG=ABS(RXN(1,J))
      BIG2=ABS(RXN(4,J))
      IF (BIG2.GT.BIG) BIG=BIG2
      IF (ABS(G(2)).LT.BIG*EBIG) G(2)=0

C     For Gluconic Acid, being produced only
      G(3)=COEFF3HH*(CONC(3,KJ+1)-CONC(3,KJ))
     1     -COEFF3HHH*(CONC(3,KJ)-CONC(3,KJ-1))
     2     +(HH/2.)*(RXN(2,J+1)+3.*RXN(2,J))/4.
     3     +(HHH/2.)*(RXN(2,J-1)+3.*RXN(2,J))/4.
      B(3,3)=COEFF3HH+COEFF3HHH
      D(3,3)=-COEFF3HH
      A(3,3)=-COEFF3HHH
      B(3,10)=-(HH/2.)*(3./4.)+(HHH/2.)*(3./4.)
      D(3,10)=-(HH/2.)*(1./4.)
      A(3,10)=-(HHH/2.)*(1./4.)

      BIG=ABS(COEFF3HH*CONC(3,KJ+1))
      BIG2=ABS(COEFF3HH*CONC(3,KJ))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(-COEFF3HHH*CONC(3,KJ))
      IF (BIG3.GT.BIG) BIG=BIG3
      BIG4=ABS(-COEFF3HHH*CONC(3,KJ-1))
      IF (BIG4.GT.BIG) BIG=BIG4
      BIG5=ABS((HH/2.)*(RXN(2,J+1)/4.))
      IF (BIG5.GT.BIG) BIG=BIG5
      BIG6=ABS((HH/2.)*(3.*RXN(2,J))/4.)
      IF (BIG6.GT.BIG) BIG=BIG6
      BIG7=ABS((HHH/2.)*(RXN(2,J-1)/4.))
      IF (BIG7.GT.BIG) BIG=BIG7
      BIG8=ABS((HHH/2.)*(3.*RXN(2,J))/4.)
      IF (BIG8.GT.BIG) BIG=BIG8
      IF (ABS(G(3)).LT.BIG*EBIG) G(3)=0

C     For GOx2, enzyme
      G(4)=CBULK(2)+CBULK(4)+CBULK(7)+CBULK(8)-CONC(2,J)-CONC(4,J)
     1    -CONC(7,J)-CONC(8,J)
      B(4,2)=+1.
      B(4,4)=+1.
      B(4,7)=+1.
      B(4,8)=+1.

      BIG=ABS(CBULK(2))
      IF (ABS(CBULK(4)).GT.BIG) BIG=ABS(CBULK(4))
      IF (ABS(CBULK(7)).GT.BIG) BIG=ABS(CBULK(7))
      IF (ABS(CBULK(8)).GT.BIG) BIG=ABS(CBULK(8))
      IF (ABS(CONC(2,J)).GT.BIG) BIG=ABS(CONC(2,J))
      IF (ABS(CONC(4,J)).GT.BIG) BIG=ABS(CONC(4,J))
      IF (ABS(CONC(7,J)).GT.BIG) BIG=ABS(CONC(7,J))
      IF (ABS(CONC(8,J)).GT.BIG) BIG=ABS(CONC(8,J))
      IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0

C     For O2, being consumed only
      G(5)=COEFF5HH*(CONC(5,KJ+1)-CONC(5,KJ))
     1     -COEFF5HHH*(CONC(5,KJ)-CONC(5,KJ-1))
     2     -(HH/2.)*(RXN(3,J+1)+3.*RXN(3,J))/4.
```

```
     3      -(HHH/2.)*(RXN(3,J-1)+3.*RXN(3,J))/4.
      B(5,5)=COEFF5HH+COEFF5HHH
      D(5,5)=-COEFF5HH
      A(5,5)=-COEFF5HHH
      B(5,11)=+(HH/2.)*(3./4.)+(HHH/2.)*(3./4.)
      D(5,11)=+(HH/2.)*(1./4.)
      A(5,11)=+(HHH/2.)*(1./4.)

      BIG=ABS(COEFF5HH*CONC(5,KJ+1))
      BIG2=ABS(COEFF5HH*CONC(5,KJ))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(-COEFF5HHH*CONC(5,KJ))
      IF (BIG3.GT.BIG) BIG=BIG3
      BIG4=ABS(-COEFF5HHH*CONC(5,KJ-1))
      IF (BIG4.GT.BIG) BIG=BIG4
      BIG5=ABS((HH/2.)*(RXN(3,J+1)/4.))
      IF (BIG5.GT.BIG) BIG=BIG5
      BIG6=ABS((HH/2.)*(3.*RXN(3,J))/4.)
      IF (BIG6.GT.BIG) BIG=BIG6
      BIG7=ABS((HHH/2.)*(RXN(3,J-1)/4.))
      IF (BIG7.GT.BIG) BIG=BIG7
      BIG8=ABS((HHH/2.)*(3.*RXN(3,J))/4.)
      IF (BIG8.GT.BIG) BIG=BIG8
      IF (ABS(G(5)).LT.BIG*EBIG) G(5)=0

C     For H2O2, reacting species
      G(6)=COEFF6HH*(CONC(6,KJ+1)-CONC(6,KJ))
     1    -COEFF6HHH*(CONC(6,KJ)-CONC(6,KJ-1))
     2    +(HH/2.)*(RXN(4,J+1)+3.*RXN(4,J))/4.
     3    +(HHH/2.)*(RXN(4,J-1)+3.*RXN(4,J))/4.
      B(6,6)=COEFF6HH+COEFF6HHH
      D(6,6)=-COEFF6HH
      A(6,6)=-COEFF6HHH
      B(6,12)=-(HH/2.)*(3./4.)+(HHH/2.)*(3./4.)
      D(6,12)=-(HH/2.)*(1./4.)
      A(6,12)=-(HHH/2.)*(1./4.)

      BIG=ABS(COEFF6HH*CONC(6,KJ+1))
      BIG2=ABS(COEFF16HH*CONC(6,KJ))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(-COEFF6HHH*CONC(6,KJ))
      IF (BIG3.GT.BIG) BIG=BIG3
      BIG4=ABS(-COEFF6HHH*CONC(6,KJ-1))
      IF (BIG4.GT.BIG) BIG=BIG4
      BIG5=ABS((HH/2.)*(RXN(4,J+1)/4.))
      IF (BIG5.GT.BIG) BIG=BIG5
      BIG6=ABS((HH/2.)*(3.*RXN(4,J))/4.)
      IF (BIG6.GT.BIG) BIG=BIG6
      BIG7=ABS((HHH/2.)*(RXN(4,J-1)/4.))
      IF (BIG7.GT.BIG) BIG=BIG7
      BIG8=ABS((HHH/2.)*(3.*RXN(4,J))/4.)
      IF (BIG8.GT.BIG) BIG=BIG8
      IF (ABS(G(6)).LT.BIG*EBIG) G(6)=0

C     For CX-GOx2, enzyme
      G(7)=RXN(1,J)-RXN(2,J)
      B(7,9)=-1.
      B(7,10)=1.
```

```
173
174         IF (ABS(RXN(1,J)).GT.BIG) BIG=ABS(RXN(1,J))
175         IF (ABS(RXN(2,J)).GT.BIG) BIG=ABS(RXN(2,J))
176         IF (ABS(G(7)).LT.BIG*EBIG) G(7)=0
177
178 C       For CX-GOx, enzyme
179         G(8)=RXN(3,J)-RXN(4,J)
180         B(8,11)=-1.
181         B(8,12)=1.
182
183         IF (ABS(RXN(3,J)).GT.BIG) BIG=ABS(RXN(3,J))
184         IF (ABS(RXN(4,J)).GT.BIG) BIG=ABS(RXN(4,J))
185         IF (ABS(G(8)).LT.BIG*EBIG) G(8)=0
186
187
188 C       REACTION1
189    214  G(9)=-RXN(1,J)+ratef1*(CONC(1,J)*CONC(2,J)-(CONC(7,J)/equilib1))
190         B(9,1)=-ratef1*CONC(2,J)
191         B(9,2)=-ratef1*CONC(1,J)
192         B(9,7)=ratef1/equilib1
193         B(9,9)=+1.
194
195         BIG=ABS(RXN(1,J))
196         BIG2=ABS(ratef1*CONC(1,J)*CONC(2,J))
197         IF (BIG2.GT.BIG) BIG=BIG2
198         BIG3=ABS(ratef1*(CONC(7,J)/equilib1))
199         IF (BIG3.GT.BIG) BIG=BIG3
200         IF (ABS(G(9)).LT.BIG*EBIG) G(9)=0
201
202 C       REACTION2
203    215  G(10)=-RXN(2,J)+ratef2*CONC(7,J)
204         B(10,7)=-ratef2
205         B(10,10)=+1.
206
207         BIG=ABS(RXN(2,J))
208         BIG2=ABS(ratef2*CONC(7,J))
209         IF (BIG2.GT.BIG) BIG=BIG2
210         IF (ABS(G(10)).LT.BIG*EBIG) G(10)=0
211
212 C       REACTION3
213    216  G(11)=-RXN(3,J)+ratef3*(CONC(4,J)*CONC(5,J)-(CONC(8,J)/equilib3))
214         B(11,4)=-ratef3*CONC(5,J)
215         B(11,5)=-ratef3*CONC(4,J)
216         B(11,8)=ratef3/equilib3
217         B(11,11)=+1.
218
219         BIG=ABS(RXN(3,J))
220         BIG2=ABS(ratef3*CONC(4,J)*CONC(5,J))
221         IF (BIG2.GT.BIG) BIG=BIG2
222         BIG3=ABS(ratef3*(CONC(8,J)/equilib3))
223         IF (BIG3.GT.BIG) BIG=BIG3
224         IF (ABS(G(11)).LT.BIG*EBIG) G(11)=0
225
226 C       REACTION4
227    217  G(12)=-RXN(4,J)+ratef4*CONC(8,J)
228         B(12,8)=-ratef4
229         B(12,12)=+1.
230
```

```
231         BIG=ABS(RXN(4,J))
232         BIG2=ABS(ratef4*CONC(8,J))
233         IF (BIG2.GT.BIG) BIG=BIG2
234         IF (ABS(G(12)).LT.BIG*EBIG) G(12)=0
235
236   212   WRITE(12,301) J, (G(K),K=1,N)
237         RETURN
238         END
```

Code E.6. Steady-State Continuous Glucose Monitor Subroutine for the Inner Region

```
      SUBROUTINE INNER(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(12,12),B(12,12),C(12,80001),D(12,25),G(12),X(12,12)
     1   ,Y(12,12)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(8,80001),RXN(4,80001),DIFF(8),H,EBIG,HH,IJ
      COMMON/POR/ POR1,POR2,PORGLU
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT
      COMMON/VARR/ HHH, KJ

  301 FORMAT (5x,'J=' I5,  12E18.9)

C     For Glucose, being consumed only
      G(1)=POR1*DIFF(1)*(CONC(1,J+1)-2.*CONC(1,J)+CONC(1,J-1))/HH**2.
     2    -RXN(1,J)
      B(1,1)=2.*POR1*DIFF(1)/HH**2.
      D(1,1)=-POR1*DIFF(1)/HH**2.
      A(1,1)=-POR1*DIFF(1)/HH**2.
      B(1,9)=+1.

      BIG=ABS(POR1*DIFF(1)*(CONC(1,J+1))/HH**2.)
      BIG2=ABS(POR1*DIFF(1)*(-2.*CONC(1,J))/HH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(POR1*DIFF(1)*(CONC(1,J-1))/HH**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(-RXN(1,J)).GT.BIG) BIG=ABS(-RXN(1,J))
      IF (ABS(G(1)).LT.BIG*EBIG) G(1)=0

C     For GOx, enzyme
      G(2)=-RXN(1,J)+RXN(4,J)
      B(2,9)=+1.
      B(2,12)=-1.

      BIG=ABS(RXN(1,J))
      BIG2=ABS(RXN(4,J))
      IF (BIG2.GT.BIG) BIG=BIG2
      IF (ABS(G(2)).LT.BIG*EBIG) G(2)=0

C     For Gluconic Acid, being produced only
      G(3)=POR1*DIFF(3)*(CONC(3,J+1)-2.*CONC(3,J)+CONC(3,J-1))/HH**2.
     2    +RXN(2,J)
      B(3,3)=2.*POR1*DIFF(3)/HH**2.
      D(3,3)=-POR1*DIFF(3)/HH**2.
      A(3,3)=-POR1*DIFF(3)/HH**2.
      B(3,10)=-1.

      BIG=ABS(POR1*DIFF(3)*(CONC(3,J+1))/HH**2.)
      BIG2=ABS(POR1*DIFF(3)*(-2.*CONC(3,J))/HH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(POR1*DIFF(3)*(CONC(3,J-1))/HH**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(RXN(2,J)).GT.BIG) BIG=ABS(RXN(2,J))
      IF (ABS(G(3)).LT.BIG*EBIG) G(3)=0

C     For GOx2, enzyme
```

```
      G(4)=CBULK(2)+CBULK(4)+CBULK(7)+CBULK(8)-CONC(2,J)-CONC(4,J)
     1    -CONC(7,J)-CONC(8,J)
      B(4,2)=+1.
      B(4,4)=+1.
      B(4,7)=+1.
      B(4,8)=+1.

      BIG=ABS(CBULK(2))
      IF (ABS(CBULK(4)).GT.BIG) BIG=ABS(CBULK(4))
      IF (ABS(CBULK(7)).GT.BIG) BIG=ABS(CBULK(7))
      IF (ABS(CBULK(8)).GT.BIG) BIG=ABS(CBULK(8))
      IF (ABS(CONC(2,J)).GT.BIG) BIG=ABS(CONC(2,J))
      IF (ABS(CONC(4,J)).GT.BIG) BIG=ABS(CONC(4,J))
      IF (ABS(CONC(7,J)).GT.BIG) BIG=ABS(CONC(7,J))
      IF (ABS(CONC(8,J)).GT.BIG) BIG=ABS(CONC(8,J))
      IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0

C     For O2, being consumed only
      G(5)=POR1*DIFF(5)*(CONC(5,J+1)-2.*CONC(5,J)+CONC(5,J-1))/HH**2.
     2    -RXN(3,J)
      B(5,5)=2.*POR1*DIFF(5)/HH**2.
      D(5,5)=-POR1*DIFF(5)/HH**2.
      A(5,5)=-POR1*DIFF(5)/HH**2.
      B(5,11)=+1.

      BIG=ABS(POR1*DIFF(5)*(CONC(5,J+1))/HH**2.)
      BIG2=ABS(POR1*DIFF(5)*(-2.*CONC(5,J))/HH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(POR1*DIFF(5)*(CONC(5,J-1))/HH**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(-RXN(3,J)).GT.BIG) BIG=ABS(-RXN(3,J))
      IF (ABS(G(5)).LT.BIG*EBIG) G(5)=0

C     For H2O2, reacting species
      G(6)=POR1*DIFF(6)*(CONC(6,J+1)-2.*CONC(6,J)+CONC(6,J-1))/HH**2.
     2    +RXN(4,J)
      B(6,6)=2.*POR1*DIFF(6)/HH**2.
      D(6,6)=-POR1*DIFF(6)/HH**2.
      A(6,6)=-POR1*DIFF(6)/HH**2.
      B(6,12)=-1.

      BIG=ABS(POR1*DIFF(6)*(CONC(6,J+1))/HH**2.)
      BIG2=ABS(POR1*DIFF(6)*(-2.*CONC(6,J))/HH**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(POR1*DIFF(6)*(CONC(6,J-1))/HH**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(RXN(4,J)).GT.BIG) BIG=ABS(RXN(4,J))
      IF (ABS(G(6)).LT.BIG*EBIG) G(6)=0

C     For CX-GOx2, enzyme
      G(7)=RXN(1,J)-RXN(2,J)
      B(7,9)=-1.
      B(7,10)=1.

      IF (ABS(RXN(1,J)).GT.BIG) BIG=ABS(RXN(1,J))
      IF (ABS(RXN(2,J)).GT.BIG) BIG=ABS(RXN(2,J))
      IF (ABS(G(7)).LT.BIG*EBIG) G(7)=0

```

```
C       For CX-GOx, enzyme
        G(8)=RXN(3,J)-RXN(4,J)
        B(8,11)=-1.
        B(8,12)=1.

        IF (ABS(RXN(3,J)).GT.BIG) BIG=ABS(RXN(3,J))
        IF (ABS(RXN(4,J)).GT.BIG) BIG=ABS(RXN(4,J))
        IF (ABS(G(8)).LT.BIG*EBIG) G(8)=0


C       REACTION1
  214   G(9)=-RXN(1,J)+ratef1*(CONC(1,J)*CONC(2,J)-(CONC(7,J)/equilib1))
        B(9,1)=-ratef1*CONC(2,J)
        B(9,2)=-ratef1*CONC(1,J)
        B(9,7)=ratef1/equilib1
        B(9,9)=+1.

        BIG=ABS(RXN(1,J))
        BIG2=ABS(ratef1*CONC(1,J)*CONC(2,J))
        IF (BIG2.GT.BIG) BIG=BIG2
        BIG3=ABS(ratef1*(CONC(7,J)/equilib1))
        IF (BIG3.GT.BIG) BIG=BIG3
        IF (ABS(G(9)).LT.BIG*EBIG) G(9)=0

C       REACTION2
  215   G(10)=-RXN(2,J)+ratef2*CONC(7,J)
        B(10,7)=-ratef2
        B(10,10)=+1.

        BIG=ABS(RXN(2,J))
        BIG2=ABS(ratef2*CONC(7,J))
        IF (BIG2.GT.BIG) BIG=BIG2
        IF (ABS(G(10)).LT.BIG*EBIG) G(10)=0

C       REACTION3
  216   G(11)=-RXN(3,J)+ratef3*(CONC(4,J)*CONC(5,J)-(CONC(8,J)/equilib3))
        B(11,4)=-ratef3*CONC(5,J)
        B(11,5)=-ratef3*CONC(4,J)
        B(11,8)=ratef3/equilib3
        B(11,11)=+1.

        BIG=ABS(RXN(3,J))
        BIG2=ABS(ratef3*CONC(4,J)*CONC(5,J))
        IF (BIG2.GT.BIG) BIG=BIG2
        BIG3=ABS(ratef3*(CONC(8,J)/equilib3))
        IF (BIG3.GT.BIG) BIG=BIG3
        IF (ABS(G(11)).LT.BIG*EBIG) G(11)=0

C       REACTION4
  217   G(12)=-RXN(4,J)+ratef4*CONC(8,J)
        B(12,8)=-ratef4
        B(12,12)=+1.

        BIG=ABS(RXN(4,J))
        BIG2=ABS(ratef4*CONC(8,J))
        IF (BIG2.GT.BIG) BIG=BIG2
        IF (ABS(G(12)).LT.BIG*EBIG) G(12)=0

```

```
173
174 c      SAVE G OUT DATA
175   212 DO 11 I=2,13
176    11 If (I.EQ.J) WRITE(12,301) J, (G(K),K=1,N)
177       IF (J.EQ.IJ/2) THEN
178       WRITE(12,301) J, (G(K),K=1,N)
179       ELSE IF (J.EQ.(KJ+1)) THEN
180       WRITE(12,301) J, (G(K),K=1,N)
181       ELSE IF (J.EQ.(KJ+2)) THEN
182       WRITE(12,301) J, (G(K),K=1,N)
183       ELSE IF (J.EQ.(KJ+3)) THEN
184       WRITE(12,301) J, (G(K),K=1,N)
185       ELSE IF (J.EQ.(KJ+4)) THEN
186       WRITE(12,301) J, (G(K),K=1,N)
187       ELSE IF (J.EQ.(IJ-1)) THEN
188       WRITE(12,301) J, (G(K),K=1,N)
189       ELSE IF (J.EQ.(IJ-2)) THEN
190       WRITE(12,301) J, (G(K),K=1,N)
191       ELSE IF (J.EQ.(IJ-3)) THEN
192       WRITE(12,301) J, (G(K),K=1,N)
193       END IF
194
195       RETURN
196       END
```

Code E.7. Steady-State Continuous Glucose Monitor Subroutine for the Second Coupler

```
      SUBROUTINE COUPLER2(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(12,12),B(12,12),C(12,80001),D(12,25),G(12),X(12,12)
     1   ,Y(12,12)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(8,80001),RXN(4,80001),DIFF(8),H,EBIG,HH,IJ
      COMMON/POR/ POR1,POR2,PORGLU
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT

  301 FORMAT (5x,'J=' I5,  12E18.9)

      COEFF1H=2.*PORGLU*DIFF(1)/H/HH
      COEFF1HH=2.*POR1*DIFF(1)/HH/HH
      COEFF3H=2.*PORGLU*DIFF(3)/H/HH
      COEFF3HH=2.*POR1*DIFF(3)/HH/HH
      COEFF5H=2.*POR2*DIFF(5)/H/HH
      COEFF5HH=2.*POR1*DIFF(5)/HH/HH
      COEFF6H=2.*POR2*DIFF(6)/H/HH
      COEFF6HH=2.*POR1*DIFF(6)/HH/HH

C     For Glucose, being consumed only
      G(1)=COEFF1H*(CONC(1,J+1)-CONC(1,J))
     1     -COEFF1HH*(CONC(1,J)-CONC(1,J-1))
     2     -(RXN(1,J-1)+3.*RXN(1,J))/4.
      B(1,1)=COEFF1H+COEFF1HH
      D(1,1)=-COEFF1H
      A(1,1)=-COEFF1HH
      B(1,9)=3./4.
      A(1,9)=1./4.

      BIG=ABS(COEFF1H*CONC(1,IJ+1))
      BIG2=ABS(COEFF1H*CONC(1,IJ))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG5=ABS(COEFF1HH*CONC(1,IJ))
      IF (BIG5.GT.BIG) BIG=BIG5
      BIG6=ABS(COEFF1HH*CONC(1,IJ-1))
      IF (BIG6.GT.BIG) BIG=BIG6
      BIG7=ABS(3*RXN(1,J)/4)
      IF (BIG7.GT.BIG) BIG=BIG7
      BIG8=ABS(RXN(1,J-1)/4)
      IF (BIG8.GT.BIG) BIG=BIG8
      IF (ABS(G(1)).LT.BIG*EBIG) G(1)=0

C     For GOx, enzyme
      G(2)=-RXN(1,J)+RXN(4,J)
      B(2,9)=+1.
      B(2,12)=-1.

      BIG=ABS(RXN(1,J))
      BIG2=ABS(RXN(4,J))
      IF (BIG2.GT.BIG) BIG=BIG2
      IF (ABS(G(2)).LT.BIG*EBIG) G(2)=0

C     For Gluconic Acid, being produced only
      G(3)=COEFF3H*(CONC(3,J+1)-CONC(3,J))
```

```
     1      -COEFF3HH*(CONC(3,J)-CONC(3,J-1))
     2      +(RXN(2,J-1)+3.*RXN(2,J))/4.
       B(3,3)=COEFF3H+COEFF3HH
       D(3,3)=-COEFF3H
       A(3,3)=-COEFF3HH
       B(3,10)=-3./4.
       A(3,10)=-1./4.

       BIG=ABS(COEFF3H*CONC(3,IJ+1))
       BIG2=ABS(COEFF3H*CONC(3,IJ))
       IF (BIG2.GT.BIG) BIG=BIG2
       BIG5=ABS(COEFF3HH*CONC(3,IJ))
       IF (BIG5.GT.BIG) BIG=BIG5
       BIG6=ABS(COEFF3HH*CONC(3,IJ-1))
       IF (BIG6.GT.BIG) BIG=BIG6
       BIG7=ABS(3*RXN(2,J)/4)
       IF (BIG7.GT.BIG) BIG=BIG7
       BIG8=ABS(RXN(2,J-1)/4)
       IF (BIG8.GT.BIG) BIG=BIG8
       IF (ABS(G(3)).LT.BIG*EBIG) G(3)=0

C      For GOx2, enzyme
       G(4)=CBULK(2)+CBULK(4)+CBULK(7)+CBULK(8)-CONC(2,J)-CONC(4,J)
     1   -CONC(7,J)-CONC(8,J)
       B(4,2)=+1.
       B(4,4)=+1.
       B(4,7)=+1.
       B(4,8)=+1.

       BIG=ABS(CBULK(2))
       IF (ABS(CBULK(4)).GT.BIG) BIG=ABS(CBULK(4))
       IF (ABS(CBULK(7)).GT.BIG) BIG=ABS(CBULK(7))
       IF (ABS(CBULK(8)).GT.BIG) BIG=ABS(CBULK(8))
       IF (ABS(CONC(2,J)).GT.BIG) BIG=ABS(CONC(2,J))
       IF (ABS(CONC(4,J)).GT.BIG) BIG=ABS(CONC(4,J))
       IF (ABS(CONC(7,J)).GT.BIG) BIG=ABS(CONC(7,J))
       IF (ABS(CONC(8,J)).GT.BIG) BIG=ABS(CONC(8,J))
       IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0

C      For O2, being consumed only
       G(5)=COEFF5H*(CONC(5,J+1)-CONC(5,J))
     1      -COEFF5HH*(CONC(5,J)-CONC(5,J-1))
     2      -(RXN(3,J-1)+3.*RXN(3,J))/4.
       B(5,5)=COEFF5H+COEFF5HH
       D(5,5)=-COEFF5H
       A(5,5)=-COEFF5HH
       B(5,11)=3./4.
       A(5,11)=1./4.

       BIG=ABS(COEFF5H*CONC(5,IJ+1))
       BIG2=ABS(COEFF5H*CONC(5,IJ))
       IF (BIG2.GT.BIG) BIG=BIG2
       BIG5=ABS(COEFF5HH*CONC(5,IJ))
       IF (BIG5.GT.BIG) BIG=BIG5
       BIG6=ABS(COEFF5HH*CONC(5,IJ-1))
       IF (BIG6.GT.BIG) BIG=BIG6
       BIG7=ABS(3*RXN(3,J)/4)
       IF (BIG7.GT.BIG) BIG=BIG7
```

```
      BIG8=ABS(RXN(3,J-1)/4)
      IF (BIG8.GT.BIG) BIG=BIG8
      IF (ABS(G(5)).LT.BIG*EBIG) G(5)=0

C     For H2O2, reacting species
      G(6)=COEFF6H*(CONC(6,J+1)-CONC(6,J))
     1     -COEFF6HH*(CONC(6,J)-CONC(6,J-1))
     2     +(RXN(4,J-1)+3.*RXN(4,J))/4.
      B(6,6)=COEFF6H+COEFF6HH
      D(6,6)=-COEFF6H
      A(6,6)=-COEFF6HH
      B(6,12)=-3./4.
      A(6,12)=-1./4.

      BIG=ABS(COEFF6H*CONC(6,IJ+1))
      BIG2=ABS(COEFF6H*CONC(6,IJ))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG5=ABS(COEFF6HH*CONC(6,IJ))
      IF (BIG5.GT.BIG) BIG=BIG5
      BIG6=ABS(COEFF6HH*CONC(6,IJ-1))
      IF (BIG6.GT.BIG) BIG=BIG6
      BIG7=ABS(3*RXN(4,J)/4)
      IF (BIG7.GT.BIG) BIG=BIG7
      BIG8=ABS(RXN(4,J-1)/4)
      IF (BIG8.GT.BIG) BIG=BIG8
      IF (ABS(G(6)).LT.BIG*EBIG) G(6)=0

C     For CX-GOx2, enzyme
      G(7)=RXN(1,J)-RXN(2,J)
      B(7,9)=-1.
      B(7,10)=1.

      IF (ABS(RXN(1,J)).GT.BIG) BIG=ABS(RXN(1,J))
      IF (ABS(RXN(2,J)).GT.BIG) BIG=ABS(RXN(2,J))
      IF (ABS(G(7)).LT.BIG*EBIG) G(7)=0

C     For CX-GOx, enzyme
      G(8)=RXN(3,J)-RXN(4,J)
      B(8,11)=-1.
      B(8,12)=1.

      IF (ABS(RXN(3,J)).GT.BIG) BIG=ABS(RXN(3,J))
      IF (ABS(RXN(4,J)).GT.BIG) BIG=ABS(RXN(4,J))
      IF (ABS(G(8)).LT.BIG*EBIG) G(8)=0


C     REACTION1
  214 G(9)=-RXN(1,J)+ratef1*(CONC(1,J)*CONC(2,J)-(CONC(7,J)/equilib1))
      B(9,1)=-ratef1*CONC(2,J)
      B(9,2)=-ratef1*CONC(1,J)
      B(9,7)=ratef1/equilib1
      B(9,9)=+1.

      BIG=ABS(RXN(1,J))
      BIG2=ABS(ratef1*CONC(1,J)*CONC(2,J))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(ratef1*(CONC(7,J)/equilib1))
      IF (BIG3.GT.BIG) BIG=BIG3
```

```
      IF (ABS(G(9)).LT.BIG*EBIG) G(9)=0

C     REACTION2
  215 G(10)=-RXN(2,J)+ratef2*CONC(7,J)
      B(10,7)=-ratef2
      B(10,10)=+1.

      BIG=ABS(RXN(2,J))
      BIG2=ABS(ratef2*CONC(7,J))
      IF (BIG2.GT.BIG) BIG=BIG2
      IF (ABS(G(10)).LT.BIG*EBIG) G(10)=0

C     REACTION3
  216 G(11)=-RXN(3,J)+ratef3*(CONC(4,J)*CONC(5,J)-(CONC(8,J)/equilib3))
      B(11,4)=-ratef3*CONC(5,J)
      B(11,5)=-ratef3*CONC(4,J)
      B(11,8)=ratef3/equilib3
      B(11,11)=+1.

      BIG=ABS(RXN(3,J))
      BIG2=ABS(ratef3*CONC(4,J)*CONC(5,J))
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(ratef3*(CONC(8,J)/equilib3))
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(G(11)).LT.BIG*EBIG) G(11)=0

C     REACTION4
  217 G(12)=-RXN(4,J)+ratef4*CONC(8,J)
      B(12,8)=-ratef4
      B(12,12)=+1.

      BIG=ABS(RXN(4,J))
      BIG2=ABS(ratef4*CONC(8,J))
      IF (BIG2.GT.BIG) BIG=BIG2
      IF (ABS(G(12)).LT.BIG*EBIG) G(12)=0

  212 WRITE(12,301) J, (G(K),K=1,N)
      RETURN
      END
```

Code E.8. Steady-State Continuous Glucose Monitor Subroutine for GLM Region

```
      SUBROUTINE OUTER(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(12,12),B(12,12),C(12,80001),D(12,25),G(12),X(12,12)
     1   ,Y(12,12)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(8,80001),RXN(4,80001),DIFF(8),H,EBIG,HH,IJ
      COMMON/POR/ POR1,POR2,PORGLU
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4

  301 FORMAT (5x,'J=' I5,  12E18.9)

C     For Glucose, being consumed only
      G(1)=PORGLU*DIFF(1)*(CONC(1,J+1)-2.*CONC(1,J)+CONC(1,J-1))/H**2.
      B(1,1)=2.*PORGLU*DIFF(1)/H**2
      D(1,1)=-PORGLU*DIFF(1)/H**2
      A(1,1)=-PORGLU*DIFF(1)/H**2

      BIG=ABS(PORGLU*DIFF(1)*(CONC(1,J+1))/H**2.)
      BIG2=ABS(PORGLU*DIFF(1)*(-2.*CONC(1,J))/H**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(PORGLU*DIFF(1)*(CONC(1,J-1))/H**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(G(1)).LT.BIG*EBIG) G(1)=0

C     For GOx, enzyme
      G(2)=CONC(2,J)
      B(2,2)=-1.

c      BIG=ABS(CONC(2,J))
c      IF (ABS(G(2)).LT.BIG*EBIG) G(2)=0

C     For Gluconic Acid, being produced only
      G(3)=PORGLU*DIFF(3)*(CONC(3,J+1)-2.*CONC(3,J)+CONC(3,J-1))/H**2.
      B(3,3)=2.*PORGLU*DIFF(3)/H**2
      D(3,3)=-PORGLU*DIFF(3)/H**2
      A(3,3)=-PORGLU*DIFF(3)/H**2

      BIG=ABS(PORGLU*DIFF(3)*(CONC(3,J+1))/H**2.)
      BIG2=ABS(PORGLU*DIFF(3)*(-2.*CONC(3,J))/H**2.)
      IF (BIG2.GT.BIG) BIG=BIG2
      BIG3=ABS(PORGLU*DIFF(3)*(CONC(3,J-1))/H**2.)
      IF (BIG3.GT.BIG) BIG=BIG3
      IF (ABS(G(3)).LT.BIG*EBIG) G(3)=0

C     For GOx2, enzyme
      G(4)=CONC(4,J)
      B(4,4)=-1.

c      BIG=ABS(CONC(4,J))
c      IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0

C     For O2, being consumed only
      G(5)=POR2*DIFF(5)*(CONC(5,J+1)-2.*CONC(5,J)+CONC(5,J-1))/H**2.
      B(5,5)=2.*POR2*DIFF(5)/H**2
      D(5,5)=-POR2*DIFF(5)/H**2
      A(5,5)=-POR2*DIFF(5)/H**2
```

```
57
58          BIG=ABS(POR2*DIFF(5)*(CONC(5,J+1))/H**2.)
59          BIG2=ABS(POR2*DIFF(5)*(-2.*CONC(5,J))/H**2.)
60          IF (BIG2.GT.BIG) BIG=BIG2
61          BIG3=ABS(POR2*DIFF(5)*(CONC(5,J-1))/H**2.)
62          IF (BIG3.GT.BIG) BIG=BIG3
63          IF (ABS(G(5)).LT.BIG*EBIG) G(5)=0
64
65 C        For H2O2, reacting species
66          G(6)=POR2*DIFF(6)*(CONC(6,J+1)-2.*CONC(6,J)+CONC(6,J-1))/H**2.
67          B(6,6)=2.*POR2*DIFF(6)/H**2
68          D(6,6)=-POR2*DIFF(6)/H**2
69          A(6,6)=-POR2*DIFF(6)/H**2
70
71          BIG=ABS(POR2*DIFF(6)*(CONC(6,J+1))/H**2.)
72          BIG2=ABS(POR2*DIFF(6)*(-2.*CONC(6,J))/H**2.)
73          IF (BIG2.GT.BIG) BIG=BIG2
74          BIG3=ABS(POR2*DIFF(6)*(CONC(6,J-1))/H**2.)
75          IF (BIG3.GT.BIG) BIG=BIG3
76          IF (ABS(G(6)).LT.BIG*EBIG) G(6)=0
77
78 C        For CX-GOx2, enzyme complex
79          G(7)=CONC(7,J)
80          B(7,7)=-1.
81
82 c         BIG=ABS(CONC(7,J))
83 c         IF (ABS(G(7)).LT.BIG*EBIG) G(7)=0
84
85 C        For CX-GOx, enzyme complex
86          G(8)=CONC(8,J)
87          B(8,8)=-1.
88
89 c         BIG=ABS(CONC(8,J))
90 c         IF (ABS(G(8)).LT.BIG*EBIG) G(8)=0
91
92 C        For Reaction 1 Enzymatic Catalysis
93          G(9)=RXN(1,J)
94          B(9,9)=-1.
95
96 c         BIG=ABS(RXN(1,J))
97 c         IF (ABS(G(9)).LT.BIG*EBIG) G(9)=0
98
99 C        For Reaction 2
100         G(10)=RXN(2,J)
101         B(10,10)=-1.
102
103 c         BIG=ABS(RXN(2,J))
104 c         IF (ABS(G(10)).LT.BIG*EBIG) G(10)=0
105
106 C        For Reaction 3 Meditation/regeneration
107         G(11)=RXN(3,J)
108         B(11,11)=-1.
109
110 c         BIG=ABS(RXN(3,J))
111 c         IF (ABS(G(11)).LT.BIG*EBIG) G(11)=0
112
113 C        For Reaction 4
114         G(12)=RXN(4,J)
```

```
115         B(12,12)=-1.
116
117 c         BIG=ABS(RXN(4,J))
118 c         IF (ABS(G(12)).LT.BIG*EBIG) G(12)=0
119
120 c       SAVE G OUT DATA
121         IF (J.EQ.(IJ+(NJ-IJ)/2)) THEN
122         WRITE(12,301) J, (G(K),K=1,N)
123         ELSE IF (J.EQ.(NJ-1)) THEN
124         WRITE(12,301) J, (G(K),K=1,N)
125         ELSE IF (J.EQ.(IJ+1)) THEN
126         WRITE(12,301) J, (G(K),K=1,N)
127         END IF
128
129         RETURN
130         END
```

Code E.9. Steady-State Continuous Glucose Monitor Subroutine for the Bulk Boundary Condition

```
      SUBROUTINE BCNJ(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAB/ A(12,12),B(12,12),C(12,80001),D(12,25),G(12),X(12,12)
     1   ,Y(12,12)
      COMMON/NSN/ N, NJ
      COMMON/VAR/ CONC(8,80001),RXN(4,80001),DIFF(8),H,EBIG,HH,IJ
      COMMON/POR/ POR1,POR2,PORGLU
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT

  301 FORMAT (5x,'J=' I5,  12E18.9)

C     For Glucose, being consumed only
      G(1)=PARGLUCOSE*CBULK(1)-CONC(1,J)
      B(1,1)=1.0

      BIG=ABS(PARGLUCOSE*CBULK(1))
      IF (ABS(CONC(1,J)).GT.BIG) BIG=ABS(CONC(1,J))
      IF (ABS(G(1)).LT.BIG*EBIG) G(1)=0

C     For GOx, enzyme
      G(2)=CONC(2,J)
      B(2,2)=-1.

      BIG=ABS(CONC(2,J))
      IF (ABS(G(2)).LT.BIG*EBIG) G(2)=0

C     For Gluconic Acid, being produced only
      G(3)=PARGLUCOSE*CBULK(3)-CONC(3,J)
      B(3,3)=1.0

      BIG=ABS(PARGLUCOSE*CBULK(3))
    IF (ABS(CONC(3,J)).GT.BIG) BIG=ABS(CONC(3,J))
      IF (ABS(G(3)).LT.BIG*EBIG) G(3)=0

C     For GOx2, enzyme
      G(4)=CONC(4,J)
      B(4,4)=-1.

      BIG=ABS(CONC(4,J))
      IF (ABS(G(4)).LT.BIG*EBIG) G(4)=0

C     For O2, being consumed only
      G(5)=PAR02*CBULK(5)-CONC(5,J)
      B(5,5)=1.0

      BIG=ABS(PARO2*CBULK(5))
    IF (ABS(CONC(5,J)).GT.BIG) BIG=ABS(CONC(5,J))
      IF (ABS(G(5)).LT.BIG*EBIG) G(5)=0

C     For H2O2, reacting species
      G(6)=PARH2O2*CBULK(6)-CONC(6,J)
      B(6,6)=1.0

      BIG=ABS(PARH2O2*CBULK(6))
```

```
55       IF (ABS(CONC(6,J)).GT.BIG) BIG=ABS(CONC(6,J))
56         IF (ABS(G(6)).LT.BIG*EBIG) G(6)=0
57
58 C       For CX-GOx2, enzyme complex
59         G(7)=CONC(7,J)
60         B(7,7)=-1.
61
62         BIG=ABS(CONC(7,J))
63         IF (ABS(G(7)).LT.BIG*EBIG) G(7)=0
64
65 C       For CX-GOx, enzyme complex
66         G(8)=CONC(8,J)
67         B(8,8)=-1.
68
69         BIG=ABS(CONC(8,J))
70         IF (ABS(G(8)).LT.BIG*EBIG) G(8)=0
71
72 C       For Reaction 1 Enzymatic Catalysis
73         G(9)=RXN(1,J)
74         B(9,9)=-1.
75
76         BIG=ABS(RXN(1,J))
77         IF (ABS(G(9)).LT.BIG*EBIG) G(9)=0
78
79 C       For Reaction 2
80         G(10)=RXN(2,J)
81         B(10,10)=-1.
82
83         BIG=ABS(RXN(2,J))
84         IF (ABS(G(10)).LT.BIG*EBIG) G(10)=0
85
86 C       For Reaction 3 Meditation/regeneration
87         G(11)=RXN(3,J)
88         B(11,11)=-1.
89
90         BIG=ABS(RXN(3,J))
91         IF (ABS(G(11)).LT.BIG*EBIG) G(11)=0
92
93 C       For Reaction 4
94         G(12)=RXN(4,J)
95         B(12,12)=-1.
96
97         BIG=ABS(RXN(4,J))
98         IF (ABS(G(12)).LT.BIG*EBIG) G(12)=0
99 C        SAVE G OUT DATA
100    206   WRITE(12,301) J, (G(K),K=1,N)
101        PRINT *, 'ITERATION=', JCOUNT
102        RETURN
103        END
```

E.3 Oscillating Continuous Glucose Monitor Code

This section contains the oscillating FORTRAN codes used to solve the convective diffusion equation with a homogeneous reaction. It reads the steady-state input file in order to solve for the oscillating concentrations. The mathematical workup for these codes are in Chapter 5. The FORTRAN codes are followed a Matlab code that reads the oscillating concentration of the reacting species and creates the dimensionless diffusion-impedance.

The first section in the code, called CONVDIFFOSCILLATING, is the main program, which outlines the global variables and sets up calling files to save over as output files as well as calling the input files. Then the subroutines that are called in the main program are all shown. They are the same titled subroutines as the steady state.

Code E.10. Oscillating Continuous Glucose Monitor Main Program

```
C     Convective Diffusion Equation with Homogeneous Reaction
C     Enzyme kinetics added
C     6 species system
C     SPECIES 1 = glucose, SPECIES 2 = GOx-FAD, SPECIES 3 = Gluconic acid
C     SPECIES 4 = GOx-FADH2, SPECIES 5 = O2, SPECIES 6 = H2O2
C     SPECIES 7 = GOx-FADH2-GA, SPECIES 8 = GOx-FAD-H2O2
C     Species 6 is the reacting species
C     This is the unsteady state solution that will eventually lead to
c     the impedance!

C     This should be ran after cdhgox_ss.for
C     The input file is the same for both of these

      PROGRAM CONVDIFFOSCILLATING
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(24,24),B(24,24),C(24,80001),D(24,49),G(24),
     1      X(24,24),Y(24,24)
      COMMON/NST/ N, NJ

      COMMON/VAR/ CONCSS(8,80001),RXNSS(4,80001),DIFF(8)
      COMMON/VARR/ PORGLU, HHH, KJ
      COMMON/CON/ C1(2,80001),C2(2,80001),C3(2,80001),C4(2,80001),
     1    C5(2,80001),C6(2,80001),C7(2,80001),C8(2,80001),
     2    RXN1(2,80001),RXN2(2,80001),RXN3(2,80001),RXN4(2,80001)
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/OTH/ POR1,POR2,H,EBIG,HH,IJ
      COMMON/BCI/ FLUX, omega
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT
      COMMON/DELT/ DELTA1, DELTA2, FREQ(400),CB(2010,80001)
      COMMON/EXTRA/ Z(8),REF(8)
      CHARACTER REF*6

  102 FORMAT (/30H THE NEXT RUN DID NOT CONVERGE)
  103 FORMAT ('Error=',E16.6/(1X,'Species=',A6,2X,'Conc at Electrode=',
     1 E12.5,2X,'Conc at Bulk=',E12.5))
  334 FORMAT (12(E25.15,5X))
  305 FORMAT (E20.12,3X,E20.12,3X,E20.12,3X,E20.12)
  335 FORMAT (16(E25.15,5X))
  336 FORMAT (1000(E25.15,1X))
  339 FORMAT (1000(E16.9,1X))
  301 FORMAT (5x,'J=' I5,  16E15.6)
  302 FORMAT ('Iteration='I4)

C     Read input values used in steady state
      open(10,file='cdhgox_in.txt',status='old')
      read(10,*) N,NJ,IJ,KJ,Y1,Y2,Y3,POR1,POR2,POR3,PARH2O2,PAR02,
     1     PARGLUCOSE,ratef1,equilib1,ratef2,ratef3,equilib3,ratef4,
     2     AKB,BB,EBIG
      read(10,*) (DIFF(I),REF(I),CBULK(I),I=1,(N-4))
      POR1=POR1**(1.5)
      POR2=POR2**(1.5)
      PORGLU=POR3**(1.5)

      open(16,file='pot_in.txt',status='old')
      read(16,*) V

```

```
C     Read steady state values from previous file
      OPEN(UNIT=11, FILE='cdhgox_out.txt')
      READ(11,334) (CONCSS(1,I),CONCSS(2,I),CONCSS(3,I),CONCSS(4,I),
     1    CONCSS(5,I),CONCSS(6,I),CONCSS(7,I),CONCSS(8,I),RXNSS(1,I),
     2    RXNSS(2,I),RXNSS(3,I),RXNSS(4,I),I=1,NJ)

      OPEN(UNIT=13, FILE='cdhgox_os_out.txt')
      CLOSE(UNIT=13, STATUS='DELETE')
      OPEN(UNIT=13, FILE='cdhgox_os_out.txt')

      OPEN(14,FILE='cdhgox_G_out.txt')
      CLOSE(14, STATUS='DELETE')
      OPEN(14, FILE='cdhgox_G_out.txt')

      OPEN(15,FILE='cdhgox_H2O2_out.txt')
      CLOSE(15, STATUS='DELETE')
      OPEN(15, FILE='cdhgox_H2O2_out.txt')

      OPEN(16,FILE='cdhgox_values_out.txt')
      CLOSE(16, STATUS='DELETE')
      OPEN(16, FILE='cdhgox_values_out.txt')

      OPEN(17,FILE='kgox_values_out.txt')
      CLOSE(17, STATUS='DELETE')
      OPEN(17, FILE='kgox_values_out.txt')

C     Constants
      F=96487.

c     THIS IS SPACING FOR OUTER LAYER, BCNJ
      H=Y3/(NJ-IJ)

c     THIS IS SPACING FOR INNER LAYER, BC1
      HH=Y2/(IJ-KJ)

c     THIS IS SPACING FOR REACTION LAYER
      HHH=Y1/(KJ-1)

C     Create flux of the reacting species constants
      FLUX=-AKB*exp(BB*V)/F/2.
      PRINT *, 'FLUX=', FLUX

C     Create charge transfer resistance
      RTB=1./(AKB*BB*CONCSS(6,1)*EXP(BB*V))
      PRINT *, 'Charge Transfer Resistance=', RTB

      N=2*N
      PRINT *, 'N=', N

  337 FORMAT (I2/I7/I7/I7/10(E15.8/)E15.8)
      write (16,337) N,NJ,IJ,KJ,H,HH,HHH,V,AKB,BB,DIFF(6),RTB,POR1

C     The number of points for frequency
      NPTS=241
      PRINT *, 'NPTS=', NPTS

c     Create range for the dimensionless frequency
      DO 261 I=1,NPTS
```

```
115       FREQ(I)=10.**(-5.+0.05*(I-1.))
116     261 WRITE (17,339), FREQ(I)
117
118
119 C     The number of points for frequency
120 C      NPTS=13
121 C      PRINT *, 'NPTS=', NPTS
122
123 C     Create range for the dimensionless frequency
124 C      DO 261 I=1,NPTS
125 C      FREQ(I)=10.**(-3.+0.5*(I-1.))
126 C  261 WRITE (17,339), FREQ(I)
127
128       DO 19 nf=1,NPTS
129 c      DO 19 nf=1,3
130
131       PRINT *, 'FREQ(NF)=', FREQ(NF)
132       omega=FREQ(NF)*DIFF(6)*POR1/(Y1+Y2)**2
133
134       IF (rateb1.LT.1E-3) omega=FREQ(NF)*DIFF(6)*POR1/(Y1+Y2+Y3)**2
135
136       PRINT *, 'omega=', omega
137     340 FORMAT (E12.6)
138       write (17,340), omega
139
140 C     Start actual code
141       DO 20 J=1,NJ
142       DO 20 I=1,N
143      20 C(I,J)=0.0
144       DO 21 J=1,NJ
145       DO 21 K=1,2
146       C1(K,J)=0.0
147       C2(K,J)=0.0
148       C3(K,J)=0.0
149       C4(K,J)=0.0
150       C5(K,J)=0.0
151       C6(K,J)=0.0
152       C7(K,J)=0.0
153       C8(K,J)=0.0
154       RXN1(K,J)=0.0
155       RXN2(K,J)=0.0
156       RXN3(K,J)=0.0
157      21 RXN4(K,J)=0.0
158       JCOUNT=0
159       TOL=1.E-10*N*NJ/1000000
160      22 JCOUNT=JCOUNT+1
161       AMP=0.0
162       J=0
163       DO 23 I=1,N
164       DO 23 K=1,N
165       Y(I,K)=0.0
166      23 X(I,K)=0.0
167      24 J=J+1
168       DO 25  I=1,N
169       G(I)=0.0
170       DO 25 K=1,N
171       A(I,K)=0.0
172       B(I,K)=0.0
```

```
   25 D(I,K)=0.0

      IF (J.EQ.1) CALL BC1(J)
      IF (J.GT.1 .AND. J.LT.KJ) CALL REACTION(J)
      IF (J.EQ.KJ) CALL COUPLER1(J)
      IF (J.GT.KJ .AND. J.LT.IJ) CALL INNER(J)
      IF (J.EQ.IJ) CALL COUPLER2(J)
      IF (J.GT.IJ .AND. J.LT.NJ) CALL OUTER(J)
      IF (J.EQ.NJ)  CALL BCNJ(J)
      CALL BAND(J)

      AMP=DABS(G(1))+DABS(G(2))+DABS(G(3))+DABS(G(4))+DABS(G(5))
     1     +DABS(G(6))+DABS(G(7))+DABS(G(8))+DABS(G(9))+DABS(G(10))
     2     +DABS(G(11))+DABS(G(12))+DABS(G(13))+DABS(G(14))
     3     +DABS(G(15))+DABS(G(16))

      IF (J.LT.NJ)  GO TO 24
      PRINT *, 'ERROR=', AMP

      DO 16 K=1,NJ
      DO 16 I=1,2
      C1(I,K)=C1(I,K)+C(I,K)
      C2(I,K)=C2(I,K)+C(I+2,K)
      C3(I,K)=C3(I,K)+C(I+4,K)
      C4(I,K)=C4(I,K)+C(I+6,K)
      C5(I,K)=C5(I,K)+C(I+8,K)
      C6(I,K)=C6(I,K)+C(I+10,K)
      C7(I,K)=C7(I,K)+C(I+12,K)
      C8(I,K)=C8(I,K)+C(I+14,K)
      RXN1(I,K)=RXN1(I,K)+C(I+16,K)
      RXN2(I,K)=RXN2(I,K)+C(I+18,K)
      RXN3(I,K)=RXN1(I,K)+C(I+20,K)
      RXN4(I,K)=RXN2(I,K)+C(I+22,K)
   16 CONTINUE

      WRITE(14,302) (JCOUNT)

      IF (DABS(AMP).LT.DABS(TOL)) GO TO  15

      IF (JCOUNT.LE.4) GO TO 22
      print 102

   15 CONTINUE
      PRINT *,'JCOUNT=',JCOUNT

      PRINT *, 'nf1=', nf

      DO 18 I=1,2
      DO 18 J=1,NJ
      BIG=C6(I,J)
      BIG2=1.0E-30
   18 IF (ABS(BIG).LE.BIG2) C6(I,J)=0.0

c      WRITE(13,335) (C1(1,J),C1(2,J),C2(1,J),C2(2,J),C3(1,J),C3(2,J),
c     1     C4(1,J),C4(2,J),C5(1,J),C5(2,J),C6(1,J),C6(2,J),C6(1,J),
c     2     C6(2,J),C6(1,J),C6(2,J),RXN1(1,J),RXN1(2,J),RXN2(1,J),
c     3     RXN2(2,J),RXN3(1,J),RXN3(2,J),RXN4(1,J),RXN4(2,J),J=1,NJ)

```

```
231       DO 19 J=1,NJ
232       CB(2*nf-1,J)=C6(1,J)
233    19 CB(2*nf,J)=C6(2,J)
234
235 c     for some reason nf is one greater then necessary
236       PRINT *, 'nf2=', nf
237
238 C      DO 17 I=1,2*nf-2
239       nf=nf-1
240     DO 17 J=1,NJ
241    17 WRITE(15,336) (CB(I,J), I=1,2*nf)
242
243   338 FORMAT (I5)
244       write (16,338) nf
245
246       PRINT *, 'DIFF(6)=', DIFF(6)
247
248       END PROGRAM CONVDIFFOSCILLATING
```

Code E.11. Oscillating Continuous Glucose Monitor for the Electrode Boundary Condition

```
      SUBROUTINE BC1(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(24,24),B(24,24),C(24,80001),D(24,49),G(24),
     1      X(24,24),Y(24,24)
      COMMON/NST/ N, NJ

      COMMON/VAR/ CONCSS(8,80001),RXNSS(4,80001),DIFF(8)
      COMMON/VARR/ PORGLU, HHH, KJ
      COMMON/CON/ C1(2,80001),C2(2,80001),C3(2,80001),C4(2,80001),
     1    C5(2,80001),C6(2,80001),C7(2,80001),C8(2,80001),
     2    RXN1(2,80001),RXN2(2,80001),RXN3(2,80001),RXN4(2,80001)
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/OTH/ POR1,POR2,H,EBIG,HH,IJ
      COMMON/BCI/ FLUX, omega
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT
      COMMON/DELT/ DELTA1, DELTA2, FREQ(400),CB(2010,80001)


  301 FORMAT (5x,'J=' I5,  24E15.6)

C   BOUNDARY CONDITION AT THE ELECTRODE, J=1
C     For Glucose, being consumed only
      G(1)=omega*(3.*C1(2,J)+C1(2,J+1))/4.
     1    +2.*POR1*DIFF(1)*(C1(1,J+1)-C1(1,J))/HHH**2.
     2    -(3.*RXN1(1,J)+RXN1(1,J+1))/4.
      B(1,1)=+2.*POR1*DIFF(1)/HHH**2.
      D(1,1)=-2.*POR1*DIFF(1)/HHH**2.
      B(1,2)=-omega*3./4.
      D(1,2)=-omega*1./4.
      B(1,17)=+3./4.
      D(1,17)=+1./4.

      G(2)=-omega*(3.*C1(1,J)+C1(1,J+1))/4.
     1    +2.*POR1*DIFF(1)*(C1(2,J+1)-C1(2,J))/HHH**2.
     2    -(3.*RXN1(2,J)+RXN1(2,J+1))/4.
      B(2,2)=+2.*POR1*DIFF(1)/HHH**2.
      D(2,2)=-2.*POR1*DIFF(1)/HHH**2.
      B(2,1)=omega*3./4.
      D(2,1)=omega*1./4.
      B(2,18)=+3./4.
      D(2,18)=+1./4.

C     For GOx, enzyme
      G(3)=omega*(3.*C2(2,J)+C2(2,J+1))/4.
     1    -(3.*RXN1(1,J)+RXN1(1,J+1))/4.
     2    +(3.*RXN4(1,J)+RXN4(1,J+1))/4.
      B(3,4)=-omega*3./4.
      D(3,4)=-omega*1./4.
      B(3,17)=+3./4.
      D(3,17)=+1./4.
      B(3,23)=-3./4.
      D(3,23)=-1./4.

      G(4)=-omega*(3.*C2(1,J)+C2(1,J+1))/4.
     1    -(3.*RXN1(2,J)+RXN1(2,J+1))/4.
     2    +(3.*RXN4(2,J)+RXN4(2,J+1))/4.
```

```
57          B(4,3)=omega*3./4.
58          D(4,3)=omega*1./4.
59          B(4,18)=+3./4.
60          D(4,18)=+1./4.
61          B(4,24)=-3./4.
62          D(4,24)=-1./4.
63
64 C        For Gluconic Acid, being produced only
65          G(5)=omega*(3.*C3(2,J)+C3(2,J+1))/4.
66         1     +2.*POR1*DIFF(3)*(C3(1,J+1)-C3(1,J))/HHH**2.
67         2     +(3.*RXN2(1,J)+RXN2(1,J+1))/4.
68          B(5,5)=+2.*POR1*DIFF(3)/HHH**2.
69          D(5,5)=-2.*POR1*DIFF(3)/HHH**2.
70          B(5,6)=-omega*3./4.
71          D(5,6)=-omega*1./4.
72          B(5,19)=-3./4.
73          D(5,19)=-1./4.
74
75          G(6)=-omega*(3.*C3(1,J)+C3(1,J+1))/4.
76         1     +2.*POR1*DIFF(3)*(C3(2,J+1)-C3(2,J))/HHH**2.
77         2     +(3.*RXN2(2,J)+RXN2(2,J+1))/4.
78          B(6,6)=+2.*POR1*DIFF(3)/HHH**2.
79          D(6,6)=-2.*POR1*DIFF(3)/HHH**2.
80          B(6,5)=omega*3./4.
81          D(6,5)=omega*1./4.
82          B(6,20)=-3./4.
83          D(6,20)=-1./4.
84
85 C        For GOx2, enzyme
86          G(7)=omega*(3.*C4(2,J)+C4(2,J+1))/4.
87         1     +(3.*RXN2(1,J)+RXN2(1,J+1))/4.
88         2     -(3.*RXN3(1,J)+RXN3(1,J+1))/4.
89          B(7,8)=-omega*3./4.
90          D(7,8)=-omega*1./4.
91          B(7,19)=-3./4.
92          D(7,19)=-1./4.
93          B(7,21)=+3./4.
94          D(7,21)=+1./4.
95
96          G(8)=-omega*(3.*C4(1,J)+C4(1,J+1))/4.
97         1     +(3.*RXN2(2,J)+RXN2(2,J+1))/4.
98         2     -(3.*RXN3(2,J)+RXN3(2,J+1))/4.
99          B(8,7)=omega*3./4.
100         D(8,7)=omega*1./4.
101         B(8,20)=-3./4.
102         D(8,20)=-1./4.
103         B(8,22)=+3./4.
104         D(8,22)=+1./4.
105
106 C       For O2, being consumed only
107         G(9)=1-C5(1,J)
108         B(9,9)=1.
109
110         G(10)=C5(2,J)
111         B(10,10)=-1.
112
113 C       For H2O2, reacting species
114         G(11)=1-C6(1,J)
```

```
115         B(11,11)=1.
116
117         G(12)=C6(2,J)
118         B(12,12)=-1.
119
120 C       For CX-GOx2, enzyme
121         G(13)=omega*(3.*C7(2,J)+C7(2,J+1))/4.
122        1     +(3.*RXN1(1,J)+RXN1(1,J+1))/4.
123        2     -(3.*RXN2(1,J)+RXN2(1,J+1))/4.
124         B(13,13)=-omega*3./4.
125         D(13,13)=-omega*1./4.
126         B(13,17)=-3./4.
127         D(13,17)=-1./4.
128         B(13,19)=+3./4.
129         D(13,19)=+1./4.
130
131         G(14)=-omega*(3.*C7(1,J)+C7(1,J+1))/4.
132        1     +(3.*RXN1(2,J)+RXN1(2,J+1))/4.
133        2     -(3.*RXN2(2,J)+RXN2(2,J+1))/4.
134         B(14,14)=omega*3./4.
135         D(14,14)=omega*1./4.
136         B(14,18)=-3./4.
137         D(14,18)=-1./4.
138         B(14,20)=+3./4.
139         D(14,20)=+1./4.
140
141 C       For CX-GOx, enzyme
142         G(15)=omega*(3.*C8(2,J)+C8(2,J+1))/4.
143        1     +(3.*RXN3(1,J)+RXN3(1,J+1))/4.
144        2     -(3.*RXN4(1,J)+RXN4(1,J+1))/4.
145         B(15,15)=-omega*3./4.
146         D(15,15)=-omega*1./4.
147         B(15,21)=-3./4.
148         D(15,21)=-1./4.
149         B(15,23)=+3./4.
150         D(15,23)=+1./4.
151
152         G(16)=-omega*(3.*C8(1,J)+C8(1,J+1))/4.
153        1     +(3.*RXN3(2,J)+RXN3(2,J+1))/4.
154        2     -(3.*RXN4(2,J)+RXN4(2,J+1))/4.
155         B(16,16)=omega*3./4.
156         D(16,16)=omega*1./4.
157         B(16,22)=-3./4.
158         D(16,22)=-1./4.
159         B(16,24)=+3./4.
160         D(16,24)=+1./4.
161
162 C       REACTION1
163         G(17)=-RXN1(1,J)+ratef1*CONCSS(2,J)*C1(1,J)
164        1      +ratef1*CONCSS(1,J)*C2(1,J)
165        2      -C7(1,J)/equilib1
166         B(17,1)=-ratef1*CONCSS(2,J)
167         B(17,3)=-ratef1*CONCSS(1,J)
168         B(17,13)=+1./equilib1
169         B(17,17)=+1.
170
171         G(18)=-RXN1(2,J)+ratef1*CONCSS(2,J)*C1(2,J)
172        1      +ratef1*CONCSS(1,J)*C2(2,J)
```

```
     2     -C7(2,J)/equilib1
      B(18,2)=-ratef1*CONCSS(2,J)
      B(18,4)=-ratef1*CONCSS(1,J)
      B(18,14)=+1./equilib1
      B(18,18)=+1.

C     REACTION2
      G(19)=-RXN2(1,J)+ratef2*C7(1,J)
      B(19,13)=-ratef2
      B(19,19)=+1.

      G(20)=-RXN2(2,J)+ratef2*C7(2,J)
      B(20,14)=-ratef2
      B(20,20)=+1.

C     REACTION3
      G(21)=-RXN3(1,J)+ratef3*CONCSS(4,J)*C5(1,J)
     1     +ratef3*CONCSS(5,J)*C4(1,J)
     2     -C8(1,J)/equilib3
      B(21,7)=-ratef3*CONCSS(5,J)
      B(21,9)=-ratef3*CONCSS(4,J)
      B(21,15)=+1./equilib3
      B(21,21)=+1.

      G(22)=-RXN3(2,J)+ratef3*CONCSS(5,J)*C4(2,J)
     1     +ratef3*CONCSS(5,J)*C4(2,J)
     2     -C8(2,J)/equilib3
      B(22,8)=-ratef3*CONCSS(5,J)
      B(22,10)=-ratef3*CONCSS(4,J)
      B(22,16)=+1./equilib3
      B(22,22)=+1.

C     REACTION4
      G(23)=-RXN4(1,J)+ratef4*C8(1,J)
      B(23,15)=-ratef4
      B(23,23)=+1.

      G(24)=-RXN4(2,J)+ratef4*C8(2,J)
      B(24,16)=-ratef4
      B(24,24)=+1.

      WRITE(14,301) J, (G(K),K=1,N)

      RETURN
      END
```

Code E.12. Oscillating Continuous Glucose Monitor Subroutine for the Reaction Region

```
      SUBROUTINE REACTION(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(24,24),B(24,24),C(24,80001),D(24,49),G(24),
     1      X(24,24),Y(24,24)
      COMMON/NST/ N, NJ

      COMMON/VAR/ CONCSS(8,80001),RXNSS(4,80001),DIFF(8)
      COMMON/VARR/ PORGLU, HHH, KJ
      COMMON/CON/ C1(2,80001),C2(2,80001),C3(2,80001),C4(2,80001),
     1    C5(2,80001),C6(2,80001),C7(2,80001),C8(2,80001),
     2    RXN1(2,80001),RXN2(2,80001),RXN3(2,80001),RXN4(2,80001)
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/OTH/ POR1,POR2,H,EBIG,HH,IJ
      COMMON/BCI/ FLUX, omega
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT
      COMMON/DELT/ DELTA1, DELTA2, FREQ(400),CB(2010,80001)

  301 FORMAT (5x,'J=' I5,  24E15.6)

C     For Glucose, being consumed only
      G(1)=omega*C1(2,J)
     1     +POR1*DIFF(1)*(C1(1,J+1)-2.*C1(1,J)+C1(1,J-1))/HHH**2.
     3     -RXN1(1,J)
      B(1,1)=2.*POR1*DIFF(1)/HHH**2.
      A(1,1)=-POR1*DIFF(1)/HHH**2.
      D(1,1)=-POR1*DIFF(1)/HHH**2.
      B(1,2)=-omega
      B(1,17)=+1.

      G(2)=-omega*C1(1,J)
     1     +POR1*DIFF(1)*(C1(2,J+1)-2.*C1(2,J)+C1(2,J-1))/HHH**2.
     3     -RXN1(2,J)
      B(2,2)=2.*POR1*DIFF(1)/HHH**2.
      A(2,2)=-POR1*DIFF(1)/HHH**2.
      D(2,2)=-POR1*DIFF(1)/HHH**2.
      B(2,1)=omega
      B(2,18)=+1.

C     For GOx, enzyme
      G(3)=omega*C2(2,J)
     1     -RXN1(1,J)
     2     +RXN4(1,J)
      B(3,4)=-omega
      B(3,17)=+1.
      B(3,23)=-1.

      G(4)=-omega*C2(1,J)
     1     -RXN1(2,J)
     2     +RXN4(2,J)
      B(4,3)=omega
      B(4,17)=+1.
      B(4,24)=-1.

C     For Gluconic Acid, being produced only
      G(5)=omega*C3(2,J)
     1     +POR1*DIFF(3)*(C3(1,J+1)-2.*C3(1,J)+C3(1,J-1))/HHH**2.
```

```
     3      +RXN2(1,J)
       B(5,5)=2.*POR1*DIFF(3)/HHH**2.
       A(5,5)=-POR1*DIFF(3)/HHH**2.
       D(5,5)=-POR1*DIFF(3)/HHH**2.
       B(5,6)=-omega
       B(5,19)=-1.

       G(6)=-omega*C3(1,J)
     1      +POR1*DIFF(3)*(C3(2,J+1)-2.*C3(2,J)+C3(2,J-1))/HHH**2.
     3      +RXN2(2,J)
       B(6,6)=2.*POR1*DIFF(3)/HHH**2.
       A(6,6)=-POR1*DIFF(3)/HHH**2.
       D(6,6)=-POR1*DIFF(3)/HHH**2.
       B(6,5)=omega
       B(6,20)=-1.

C      For GOx2, enzyme
       G(7)=omega*C4(2,J)
     1      +RXN2(1,J)
     2      -RXN3(1,J)
       B(7,8)=-omega
       B(7,19)=-1.
       B(7,21)=+1.

       G(8)=-omega*C4(1,J)
     1      +RXN2(2,J)
     2      -RXN3(2,J)
       B(8,7)=omega
       B(8,20)=-1.
       B(8,22)=+1.

C      For O2, being consumed only
       G(9)=omega*C5(2,J)
     1      +POR1*DIFF(5)*(C5(1,J+1)-2.*C5(1,J)+C5(1,J-1))/HHH**2.
     3      -RXN3(1,J)
       B(9,9)=2.*POR1*DIFF(5)/HHH**2.
       A(9,9)=-POR1*DIFF(5)/HHH**2.
       D(9,9)=-POR1*DIFF(5)/HHH**2.
       B(9,10)=-omega
       B(9,21)=+1.

       G(10)=-omega*C5(1,J)
     1      +POR1*DIFF(5)*(C5(2,J+1)-2.*C5(2,J)+C5(2,J-1))/HHH**2.
     3      -RXN3(2,J)
       B(10,10)=2.*POR1*DIFF(5)/HHH**2.
       A(10,10)=-POR1*DIFF(5)/HHH**2.
       D(10,10)=-POR1*DIFF(5)/HHH**2.
       B(10,9)=omega
       B(10,22)=+1.

C      For H2O2, reacting species
       G(11)=omega*C6(2,J)
     1      +POR1*DIFF(6)*(C6(1,J+1)-2.*C6(1,J)+C6(1,J-1))/HHH**2.
     3      +RXN4(1,J)
       B(11,11)=2.*POR1*DIFF(6)/HHH**2.
       A(11,11)=-POR1*DIFF(6)/HHH**2.
       D(11,11)=-POR1*DIFF(6)/HHH**2.
       B(11,12)=-omega
```

```
115         B(11,23)=-1.
116
117         G(12)=-omega*C6(1,J)
118        1      +POR1*DIFF(6)*(C6(2,J+1)-2.*C6(2,J)+C6(2,J-1))/HHH**2.
119        3      +RXN4(2,J)
120         B(12,12)=2.*POR1*DIFF(6)/HHH**2.
121         A(12,12)=-POR1*DIFF(6)/HHH**2.
122         D(12,12)=-POR1*DIFF(6)/HHH**2.
123         B(12,11)=omega
124         B(12,24)=-1.
125
126 C       For CX-GOx2, enzyme
127         G(13)=omega*C7(2,J)
128        1      +RXN1(1,J)
129        2      -RXN2(1,J)
130         B(13,14)=-omega
131         B(13,17)=-1.
132         B(13,19)=+1.
133
134         G(14)=-omega*C7(1,J)
135        1      +RXN1(2,J)
136        2      -RXN2(2,J)
137         B(14,13)=omega
138         B(14,18)=-1.
139         B(14,20)=+1.
140
141 C       For CX-GOx2, enzyme
142         G(15)=omega*C8(2,J)
143        1      +RXN3(1,J)
144        2      -RXN4(1,J)
145         B(15,16)=-omega
146         B(15,21)=-1.
147         B(15,23)=+1.
148
149         G(16)=-omega*C8(1,J)
150        1      +RXN3(2,J)
151        2      -RXN4(2,J)
152         B(16,15)=omega
153         B(16,22)=-1.
154         B(16,24)=+1.
155
156 C       REACTION1
157         G(17)=-RXN1(1,J)+ratef1*CONCSS(2,J)*C1(1,J)
158        1      +ratef1*CONCSS(1,J)*C2(1,J)
159        2      -C7(1,J)/equilib1
160         B(17,1)=-ratef1*CONCSS(2,J)
161         B(17,3)=-ratef1*CONCSS(1,J)
162         B(17,13)=+1./equilib1
163         B(17,17)=+1.
164
165         G(18)=-RXN1(2,J)+ratef1*CONCSS(2,J)*C1(2,J)
166        1      +ratef1*CONCSS(1,J)*C2(2,J)
167        2      -C7(2,J)/equilib1
168         B(18,2)=-ratef1*CONCSS(2,J)
169         B(18,4)=-ratef1*CONCSS(1,J)
170         B(18,14)=+1./equilib1
171         B(18,18)=+1.
172
```

```
C       REACTION2
        G(19)=-RXN2(1,J)+ratef2*C7(1,J)
        B(19,13)=-ratef2
        B(19,19)=+1.

        G(20)=-RXN2(2,J)+ratef2*C7(2,J)
        B(20,14)=-ratef2
        B(20,20)=+1.

C       REACTION3
        G(21)=-RXN3(1,J)+ratef3*CONCSS(4,J)*C5(1,J)
       1     +ratef3*CONCSS(5,J)*C4(1,J)
       2     -C8(1,J)/equilib3
        B(21,7)=-ratef3*CONCSS(5,J)
        B(21,9)=-ratef3*CONCSS(4,J)
        B(21,15)=+1./equilib3
        B(21,21)=+1.

        G(22)=-RXN3(2,J)+ratef3*CONCSS(5,J)*C4(2,J)
       1     +ratef3*CONCSS(5,J)*C4(2,J)
       2     -C8(2,J)/equilib3
        B(22,8)=-ratef3*CONCSS(5,J)
        B(22,10)=-ratef3*CONCSS(4,J)
        B(22,16)=+1./equilib3
        B(22,22)=+1.

C       REACTION4
        G(23)=-RXN4(1,J)+ratef4*C8(1,J)
        B(23,15)=-ratef4
        B(23,23)=+1.

        G(24)=-RXN4(2,J)+ratef4*C8(2,J)
        B(24,16)=-ratef4
        B(24,24)=+1.

c       SAVE G OUT DATA
        DO 11 I=2,13
    11  If (I.EQ.J) WRITE(14,301) J, (G(K),K=1,N)
        IF (J.EQ.IJ/2) THEN
        WRITE(14,301) J, (G(K),K=1,N)
        ELSE IF (J.EQ.(IJ-1)) THEN
        WRITE(14,301) J, (G(K),K=1,N)
        ELSE IF (J.EQ.(IJ-2)) THEN
        WRITE(14,301) J, (G(K),K=1,N)
        ELSE IF (J.EQ.(IJ-3)) THEN
        WRITE(14,301) J, (G(K),K=1,N)
        END IF

        RETURN
        END
```

Code E.13. Oscillating Continuous Glucose Monitor Subroutine for the First Coupler

```
      SUBROUTINE COUPLER1(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(24,24),B(24,24),C(24,80001),D(24,49),G(24),
     1      X(24,24),Y(24,24)
      COMMON/NST/ N, NJ

      COMMON/VAR/ CONCSS(8,80001),RXNSS(4,80001),DIFF(8)
      COMMON/VARR/ PORGLU, HHH, KJ
      COMMON/CON/ C1(2,80001),C2(2,80001),C3(2,80001),C4(2,80001),
     1    C5(2,80001),C6(2,80001),C7(2,80001),C8(2,80001),
     2    RXN1(2,80001),RXN2(2,80001),RXN3(2,80001),RXN4(2,80001)
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/OTH/ POR1,POR2,H,EBIG,HH,IJ
      COMMON/BCI/ FLUX, omega
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT
      COMMON/DELT/ DELTA1, DELTA2, FREQ(400),CB(2010,80001)

  301 FORMAT (5x,'J=' I5,  24E15.6)

C     For Glucose, being consumed only
      G(1)=-HHH/2*omega*(C1(2,J+1)+3.*C1(2,J))/4.
     1     +HH/2*omega*(C1(2,J-1)+3.*C1(2,J))/4.
     2     +POR1*DIFF(1)/HH*(C1(1,J+1)-C1(1,J))
     3     -POR1*DIFF(1)/HHH*(C1(1,J)-C1(1,J-1))
     4     -(HH/2.)*(RXN1(1,J+1)+3.*RXN1(1,J))/4.
     5     -(HHH/2.)*(RXN1(1,J-1)+3.*RXN1(1,J))/4.
      B(1,1)=POR1*DIFF(1)/HH+POR1*DIFF(1)/HHH
      D(1,1)=-POR1*DIFF(1)/HH
      A(1,1)=-POR1*DIFF(1)/HHH
      B(1,2)=-HHH/2*omega*(3./4.)-HH/2*omega*(3./4.)
      D(1,2)=-HHH/2*omega*(1./4.)
      A(1,2)=-HH/2*omega*(1./4.)
      B(1,17)=(HH/2.)*3./4.+(HHH/2.)*(3./4.)
      D(1,17)=(HH/2.)*1./4.
      A(1,17)=(HHH/2.)*(1./4.)

      G(2)=-HHH/2*omega*(C1(1,J+1)+3.*C1(1,J))/4.
     1     -HH/2*omega*(C1(1,J-1)+3.*C1(1,J))/4.
     2     +POR1*DIFF(1)/HH*(C1(2,J+1)-C1(2,J))
     3     -POR1*DIFF(1)/HHH*(C1(2,J)-C1(2,J-1))
     4     -(HH/2.)*(RXN1(2,J+1)+3.*RXN1(2,J))/4.
     5     -(HHH/2.)*(RXN1(2,J-1)+3.*RXN1(2,J))/4.
      B(2,2)=POR1*DIFF(1)/HH+POR1*DIFF(1)/HHH
      D(2,2)=-POR1*DIFF(1)/HH
      A(2,2)=-POR1*DIFF(1)/HHH
      B(2,1)=+HHH/2*omega*(3./4.)+HH/2*omega*(3./4.)
      D(2,1)=+HHH/2*omega*(1./4.)
      A(2,1)=+HH/2*omega*(1./4.)
      B(2,18)=(HH/2.)*3./4.+(HHH/2.)*(3./4.)
      D(2,18)=(HH/2.)*1./4.
      A(2,18)=(HHH/2.)*(1./4.)

C     For GOx, enzyme
      G(3)=-HHH/2*omega*(C2(2,J+1)+3.*C2(2,J))/4.
     1     +HH/2*omega*(C2(2,J-1)+3.*C2(2,J))/4.
     2     -(HH/2.)*(RXN1(1,J+1)+3.*RXN1(1,J))/4.
```

```
      3      -(HHH/2.)*(RXN1(1,J-1)+3.*RXN1(1,J))/4.
      4      +(HH/2.)*(RXN4(1,J+1)+3.*RXN4(1,J))/4.
      5      +(HHH/2.)*(RXN4(1,J-1)+3.*RXN4(1,J))/4.
       B(3,4)=-HHH/2*omega*(3./4.)-HH/2*omega*(3./4.)
       D(3,4)=-HHH/2*omega*(1./4.)
       A(3,4)=-HH/2*omega*(1./4.)
       B(3,17)=-(HH/2.)*3./4.-(HHH/2.)*(3./4.)
       D(3,17)=-(HH/2.)*1./4.
       A(3,17)=-(HHH/2.)*(1./4.)
       B(3,23)=(HH/2.)*3./4.+(HHH/2.)*(3./4.)
       D(3,23)=(HH/2.)*1./4.
       A(3,23)=(HHH/2.)*(1./4.)

       G(4)=-HHH/2*omega*(C2(1,J+1)+3.*C2(1,J))/4.
      1      -HH/2*omega*(C2(1,J-1)+3.*C2(1,J))/4.
      2      -(HH/2.)*(RXN1(2,J+1)+3.*RXN1(2,J))/4.
      3      -(HHH/2.)*(RXN1(2,J-1)+3.*RXN1(2,J))/4.
      4      +(HH/2.)*(RXN4(2,J+1)+3.*RXN4(2,J))/4.
      5      +(HHH/2.)*(RXN4(2,J-1)+3.*RXN4(2,J))/4.
       B(4,3)=HHH/2*omega*(3./4.)+HH/2*omega*(3./4.)
       D(4,3)=HHH/2*omega*(1./4.)
       A(4,3)=HH/2*omega*(1./4.)
       B(4,18)=-(HH/2.)*3./4.-(HHH/2.)*(3./4.)
       D(4,18)=-(HH/2.)*1./4.
       A(4,18)=-(HHH/2.)*(1./4.)
       B(4,24)=(HH/2.)*3./4.+(HHH/2.)*(3./4.)
       D(4,24)=(HH/2.)*1./4.
       A(4,24)=(HHH/2.)*(1./4.)

C      For Gluconic Acid, being produced only
       G(5)=HHH/2*omega*(C3(2,J+1)+3.*C3(2,J))/4.
      1      +HH/2*omega*(C3(2,J-1)+3.*C3(2,J))/4.
      2      +POR1*DIFF(3)/HH*(C3(1,J+1)-C3(1,J))
      3      -POR1*DIFF(3)/HHH*(C3(1,J)-C3(1,J-1))
      4      +(HH/2.)*(RXN2(1,J+1)+3.*RXN2(1,J))/4.
      5      +(HHH/2.)*(RXN2(1,J-1)+3.*RXN2(1,J))/4.
       B(5,5)=POR1*DIFF(3)/HH+POR1*DIFF(3)/HHH
       D(5,5)=-POR1*DIFF(3)/HH
       A(5,5)=-POR1*DIFF(3)/HHH
       B(5,6)=-HHH/2*omega*(3./4.)-HH/2*omega*(3./4.)
       D(5,6)=-HHH/2*omega*(1./4.)
       A(5,6)=-HH/2*omega*(1./4.)
       B(5,19)=-(HH/2.)*3./4.-(HHH/2.)*(3./4.)
       D(5,19)=-(HH/2.)*1./4.
       A(5,19)=-(HHH/2.)*(1./4.)

       G(6)=-HHH/2*omega*(C3(1,J+1)+3.*C3(1,J))/4.
      1      -HH/2*omega*(C3(1,J-1)+3.*C3(1,J))/4.
      2      +POR1*DIFF(3)/HH*(C3(2,J+1)-C3(2,J))
      3      -POR1*DIFF(3)/HHH*(C3(2,J)-C3(2,J-1))
      4      +(HH/2.)*(RXN2(2,J+1)+3.*RXN2(2,J))/4.
      5      +(HHH/2.)*(RXN2(2,J-1)+3.*RXN2(2,J))/4.
       B(6,6)=POR1*DIFF(3)/HH+POR1*DIFF(3)/HHH
       D(6,6)=-POR1*DIFF(3)/HH
       A(6,6)=-POR1*DIFF(3)/HHH
       B(6,5)=+HHH/2*omega*(3./4.)+HH/2*omega*(3./4.)
       D(6,5)=+HHH/2*omega*(1./4.)
       A(6,5)=+HH/2*omega*(1./4.)
```

```
      B(6,20)=-(HH/2.)*3./4.-(HHH/2.)*(3./4.)
      D(6,20)=-(HH/2.)*1./4.
      A(6,20)=-(HHH/2.)*(1./4.)

C     For GOx2, enzyme
      G(7)=HHH/2*omega*(C4(2,J+1)+3.*C4(2,J))/4.
     1    +HH/2*omega*(C4(2,J-1)+3.*C4(2,J))/4.
     2    +(HH/2.)*(RXN2(1,J+1)+3.*RXN2(1,J))/4.
     3    +(HHH/2.)*(RXN2(1,J-1)+3.*RXN2(1,J))/4.
     4    -(HH/2.)*(RXN3(1,J+1)+3.*RXN3(1,J))/4.
     5    -(HHH/2.)*(RXN3(1,J-1)+3.*RXN3(1,J))/4.
      B(7,8)=-HHH/2*omega*(3./4.)-HH/2*omega*(3./4.)
      D(7,8)=-HHH/2*omega*(1./4.)
      A(7,8)=-HH/2*omega*(1./4.)
      B(7,19)=(HH/2.)*3./4.+(HHH/2.)*(3./4.)
      D(7,19)=(HH/2.)*1./4.
      A(7,19)=(HHH/2.)*(1./4.)
      B(7,21)=-(HH/2.)*3./4.-(HHH/2.)*(3./4.)
      D(7,21)=-(HH/2.)*1./4.
      A(7,21)=-(HHH/2.)*(1./4.)

      G(8)=-HHH/2*omega*(C4(1,J+1)+3.*C4(1,J))/4.
     1    -HH/2*omega*(C4(1,J-1)+3.*C4(1,J))/4.
     2    +(HH/2.)*(RXN2(2,J+1)+3.*RXN2(2,J))/4.
     3    +(HHH/2.)*(RXN2(2,J-1)+3.*RXN2(2,J))/4.
     4    -(HH/2.)*(RXN3(2,J+1)+3.*RXN3(2,J))/4.
     5    -(HHH/2.)*(RXN3(2,J-1)+3.*RXN3(2,J))/4.
      B(8,7)=HHH/2*omega*(3./4.)+HH/2*omega*(3./4.)
      D(8,7)=HHH/2*omega*(1./4.)
      A(8,7)=HH/2*omega*(1./4.)
      B(8,20)=(HH/2.)*3./4.+(HHH/2.)*(3./4.)
      D(8,20)=(HH/2.)*1./4.
      A(8,20)=(HHH/2.)*(1./4.)
      B(8,22)=-(HH/2.)*3./4.-(HHH/2.)*(3./4.)
      D(8,22)=-(HH/2.)*1./4.
      A(8,22)=-(HHH/2.)*(1./4.)

C     For O2, being consumed only
      G(9)=HHH/2*omega*(C5(2,J+1)+3.*C5(2,J))/4.
     1    +HH/2*omega*(C5(2,J-1)+3.*C5(2,J))/4.
     2    +POR1*DIFF(5)/HH*(C5(1,J+1)-C5(1,J))
     3    -POR1*DIFF(5)/HHH*(C5(1,J)-C5(1,J-1))
     4    -(HH/2.)*(RXN3(1,J+1)+3.*RXN3(1,J))/4.
     5    -(HHH/2.)*(RXN3(1,J-1)+3.*RXN3(1,J))/4.
      B(9,9)=POR1*DIFF(5)/HH+POR1*DIFF(5)/HHH
      D(9,9)=-POR1*DIFF(5)/HH
      A(9,9)=-POR1*DIFF(5)/HHH
      B(9,10)=-HHH/2*omega*(3./4.)-HH/2*omega*(3./4.)
      D(9,10)=-HHH/2*omega*(1./4.)
      A(9,10)=-HH/2*omega*(1./4.)
      B(9,21)=(HH/2.)*3./4.+(HHH/2.)*(3./4.)
      D(9,21)=(HH/2.)*1./4.
      A(9,21)=(HHH/2.)*(1./4.)

      G(10)=-HHH/2*omega*(C5(1,J+1)+3.*C5(1,J))/4.
     1    -HH/2*omega*(C5(1,J-1)+3.*C5(1,J))/4.
     2    +POR1*DIFF(5)/HH*(C5(2,J+1)-C5(2,J))
     3    -POR1*DIFF(5)/HHH*(C5(2,J)-C5(2,J-1))
```

```
     4      -(HH/2.)*(RXN3(2,J+1)+3.*RXN3(2,J))/4.
     5      -(HHH/2.)*(RXN3(2,J-1)+3.*RXN3(2,J))/4.
      B(10,10)=POR1*DIFF(5)/HH+POR1*DIFF(5)/HHH
      D(10,10)=-POR1*DIFF(5)/HH
      A(10,10)=-POR1*DIFF(5)/HHH
      B(10,9)=+HHH/2*omega*(3./4.)+HH/2*omega*(3./4.)
      D(10,9)=+HHH/2*omega*(1./4.)
      A(10,9)=+HH/2*omega*(1./4.)
      B(10,22)=(HH/2.)*3./4.+(HHH/2.)*(3./4.)
      D(10,22)=(HH/2.)*1./4.
      A(10,22)=(HHH/2.)*(1./4.)

C     For H2O2, reacting species
      G(11)=HHH/2*omega*(C6(2,J+1)+3.*C6(2,J))/4.
     1      +HH/2*omega*(C6(2,J-1)+3.*C6(2,J))/4.
     2      +POR1*DIFF(6)/HH*(C6(1,J+1)-C6(1,J))
     3      -POR1*DIFF(6)/HHH*(C6(1,J)-C6(1,J-1))
     4      +(HH/2.)*(RXN4(1,J+1)+3.*RXN4(1,J))/4.
     5      +(HHH/2.)*(RXN4(1,J-1)+3.*RXN4(1,J))/4.
      B(11,11)=POR1*DIFF(6)/HH+POR1*DIFF(6)/HHH
      D(11,11)=-POR1*DIFF(6)/HH
      A(11,11)=-POR1*DIFF(6)/HHH
      B(11,12)=-HHH/2*omega*(3./4.)-HH/2*omega*(3./4.)
      D(11,12)=-HHH/2*omega*(1./4.)
      A(11,12)=-HH/2*omega*(1./4.)
      B(11,23)=-(HH/2.)*3./4.-(HHH/2.)*(3./4.)
      D(11,23)=-(HH/2.)*1./4.
      A(11,23)=-(HHH/2.)*(1./4.)

      G(12)=-HHH/2*omega*(C6(1,J+1)+3.*C6(1,J))/4.
     1      -HH/2*omega*(C6(1,J-1)+3.*C6(1,J))/4.
     2      +POR1*DIFF(6)/HH*(C6(2,J+1)-C6(2,J))
     3      -POR1*DIFF(6)/HHH*(C6(2,J)-C6(2,J-1))
     4      +(HH/2.)*(RXN4(2,J+1)+3.*RXN4(2,J))/4.
     5      +(HHH/2.)*(RXN4(2,J-1)+3.*RXN4(2,J))/4.
      B(12,12)=POR1*DIFF(6)/HH+POR1*DIFF(6)/HHH
      D(12,12)=-POR1*DIFF(6)/HH
      A(12,12)=-POR1*DIFF(6)/HHH
      B(12,11)=+HHH/2*omega*(3./4.)+HH/2*omega*(3./4.)
      D(12,11)=+HHH/2*omega*(1./4.)
      A(12,11)=+HH/2*omega*(1./4.)
      B(12,24)=-(HH/2.)*3./4.-(HHH/2.)*(3./4.)
      D(12,24)=-(HH/2.)*1./4.
      A(12,24)=-(HHH/2.)*(1./4.)

C     For CX-GOx2, enzyme
      G(13)=HHH/2*omega*(C7(2,J+1)+3.*C7(2,J))/4.
     1      +HH/2*omega*(C7(2,J-1)+3.*C7(2,J))/4.
     2      +(HH/2.)*(RXN1(1,J+1)+3.*RXN1(1,J))/4.
     3      +(HHH/2.)*(RXN1(1,J-1)+3.*RXN1(1,J))/4.
     4      -(HH/2.)*(RXN2(1,J+1)+3.*RXN2(1,J))/4.
     5      -(HHH/2.)*(RXN2(1,J-1)+3.*RXN2(1,J))/4.
      B(13,14)=-HHH/2*omega*(3./4.)-HH/2*omega*(3./4.)
      D(13,14)=-HHH/2*omega*(1./4.)
      A(13,14)=-HH/2*omega*(1./4.)
      B(13,17)=(HH/2.)*3./4.+(HHH/2.)*(3./4.)
      D(13,17)=(HH/2.)*1./4.
      A(13,17)=(HHH/2.)*(1./4.)
```

```
231         B(13,19)=-(HH/2.)*3./4.-(HHH/2.)*(3./4.)
232         D(13,19)=-(HH/2.)*1./4.
233         A(13,19)=-(HHH/2.)*(1./4.)
234
235         G(14)=-HHH/2*omega*(C7(1,J+1)+3.*C7(1,J))/4.
236        1       -HH/2*omega*(C7(1,J-1)+3.*C7(1,J))/4.
237        2       +(HH/2.)*(RXN1(2,J+1)+3.*RXN1(2,J))/4.
238        3       +(HHH/2.)*(RXN1(2,J-1)+3.*RXN1(2,J))/4.
239        4       -(HH/2.)*(RXN2(2,J+1)+3.*RXN2(2,J))/4.
240        5       -(HHH/2.)*(RXN2(2,J-1)+3.*RXN2(2,J))/4.
241         B(14,13)=HHH/2*omega*(3./4.)+HH/2*omega*(3./4.)
242         D(14,13)=HHH/2*omega*(1./4.)
243         A(14,13)=HH/2*omega*(1./4.)
244         B(14,18)=(HH/2.)*3./4.+(HHH/2.)*(3./4.)
245         D(14,18)=(HH/2.)*1./4.
246         A(14,18)=(HHH/2.)*(1./4.)
247         B(14,20)=-(HH/2.)*3./4.-(HHH/2.)*(3./4.)
248         D(14,20)=-(HH/2.)*1./4.
249         A(14,20)=-(HHH/2.)*(1./4.)
250
251 C       For CX-GOx, enzyme
252         G(15)=HHH/2*omega*(C8(2,J+1)+3.*C8(2,J))/4.
253        1       +HH/2*omega*(C8(2,J-1)+3.*C8(2,J))/4.
254        2       +(HH/2.)*(RXN3(1,J+1)+3.*RXN3(1,J))/4.
255        3       +(HHH/2.)*(RXN3(1,J-1)+3.*RXN3(1,J))/4.
256        4       -(HH/2.)*(RXN4(1,J+1)+3.*RXN4(1,J))/4.
257        5       -(HHH/2.)*(RXN4(1,J-1)+3.*RXN4(1,J))/4.
258         B(15,16)=-HHH/2*omega*(3./4.)-HH/2*omega*(3./4.)
259         D(15,16)=-HHH/2*omega*(1./4.)
260         A(15,16)=-HH/2*omega*(1./4.)
261         B(15,21)=(HH/2.)*3./4.+(HHH/2.)*(3./4.)
262         D(15,21)=(HH/2.)*1./4.
263         A(15,21)=(HHH/2.)*(1./4.)
264         B(15,23)=-(HH/2.)*3./4.-(HHH/2.)*(3./4.)
265         D(15,23)=-(HH/2.)*1./4.
266         A(15,23)=-(HHH/2.)*(1./4.)
267
268         G(16)=-HHH/2*omega*(C8(1,J+1)+3.*C8(1,J))/4.
269        1       -HH/2*omega*(C8(1,J-1)+3.*C8(1,J))/4.
270        2       +(HH/2.)*(RXN3(2,J+1)+3.*RXN3(2,J))/4.
271        3       +(HHH/2.)*(RXN3(2,J-1)+3.*RXN3(2,J))/4.
272        4       -(HH/2.)*(RXN4(2,J+1)+3.*RXN4(2,J))/4.
273        5       -(HHH/2.)*(RXN4(2,J-1)+3.*RXN4(2,J))/4.
274         B(16,15)=HHH/2*omega*(3./4.)+HH/2*omega*(3./4.)
275         D(16,15)=HHH/2*omega*(1./4.)
276         A(16,15)=HH/2*omega*(1./4.)
277         B(16,22)=(HH/2.)*3./4.+(HHH/2.)*(3./4.)
278         D(16,22)=(HH/2.)*1./4.
279         A(16,22)=(HHH/2.)*(1./4.)
280         B(16,23)=-(HH/2.)*3./4.-(HHH/2.)*(3./4.)
281         D(16,23)=-(HH/2.)*1./4.
282         A(16,23)=-(HHH/2.)*(1./4.)
283
284 C       REACTION1
285         G(17)=-RXN1(1,J)+ratef1*CONCSS(2,J)*C1(1,J)
286        1       +ratef1*CONCSS(1,J)*C2(1,J)
287        2       -C7(1,J)/equilib1
288         B(17,1)=-ratef1*CONCSS(2,J)
```

```
      B(17,3)=-ratef1*CONCSS(1,J)
      B(17,13)=+1./equilib1
      B(17,17)=+1.

      G(18)=-RXN1(2,J)+ratef1*CONCSS(2,J)*C1(2,J)
     1     +ratef1*CONCSS(1,J)*C2(2,J)
     2     -C7(2,J)/equilib1
      B(18,2)=-ratef1*CONCSS(2,J)
      B(18,4)=-ratef1*CONCSS(1,J)
      B(18,14)=+1./equilib1
      B(18,18)=+1.

C     REACTION2
      G(19)=-RXN2(1,J)+ratef2*C7(1,J)
      B(19,13)=-ratef2
      B(19,19)=+1.

      G(20)=-RXN2(2,J)+ratef2*C7(2,J)
      B(20,14)=-ratef2
      B(20,20)=+1.

C     REACTION3
      G(21)=-RXN3(1,J)+ratef3*CONCSS(4,J)*C5(1,J)
     1     +ratef3*CONCSS(5,J)*C4(1,J)
     2     -C8(1,J)/equilib3
      B(21,7)=-ratef3*CONCSS(5,J)
      B(21,9)=-ratef3*CONCSS(4,J)
      B(21,15)=+1./equilib3
      B(21,21)=+1.

      G(22)=-RXN3(2,J)+ratef3*CONCSS(5,J)*C4(2,J)
     1     +ratef3*CONCSS(5,J)*C4(2,J)
     2     -C8(2,J)/equilib3
      B(22,8)=-ratef3*CONCSS(5,J)
      B(22,10)=-ratef3*CONCSS(4,J)
      B(22,16)=+1./equilib3
      B(22,22)=+1.

C     REACTION4
      G(23)=-RXN4(1,J)+ratef4*C8(1,J)
      B(23,15)=-ratef4
      B(23,23)=+1.

      G(24)=-RXN4(2,J)+ratef4*C8(2,J)
      B(24,16)=-ratef4
      B(24,24)=+1.

      WRITE(14,301) J, (G(K),K=1,N)

      RETURN
      END
```

Code E.14. Oscillating Continuous Glucose Monitor Subroutine for the Inner Region

```
      SUBROUTINE INNER(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(24,24),B(24,24),C(24,80001),D(24,49),G(24),
     1      X(24,24),Y(24,24)
      COMMON/NST/ N, NJ

      COMMON/VAR/ CONCSS(8,80001),RXNSS(4,80001),DIFF(8)
      COMMON/VARR/ PORGLU, HHH, KJ
      COMMON/CON/ C1(2,80001),C2(2,80001),C3(2,80001),C4(2,80001),
     1    C5(2,80001),C6(2,80001),C7(2,80001),C8(2,80001),
     2    RXN1(2,80001),RXN2(2,80001),RXN3(2,80001),RXN4(2,80001)
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/OTH/ POR1,POR2,H,EBIG,HH,IJ
      COMMON/BCI/ FLUX, omega
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT
      COMMON/DELT/ DELTA1, DELTA2, FREQ(400),CB(2010,80001)

  301 FORMAT (5x,'J=' I5,  24E15.6)

C     For Glucose, being consumed only
      G(1)=omega*C1(2,J)
     1     +POR1*DIFF(1)*(C1(1,J+1)-2.*C1(1,J)+C1(1,J-1))/HH**2.
     3     -RXN1(1,J)
      B(1,1)=2.*POR1*DIFF(1)/HH**2.
      A(1,1)=-POR1*DIFF(1)/HH**2.
      D(1,1)=-POR1*DIFF(1)/HH**2.
      B(1,2)=-omega
      B(1,17)=+1.

      G(2)=-omega*C1(1,J)
     1     +POR1*DIFF(1)*(C1(2,J+1)-2.*C1(2,J)+C1(2,J-1))/HH**2.
     3     -RXN1(2,J)
      B(2,2)=2.*POR1*DIFF(1)/HH**2.
      A(2,2)=-POR1*DIFF(1)/HH**2.
      D(2,2)=-POR1*DIFF(1)/HH**2.
      B(2,1)=omega
      B(2,18)=+1.

C     For GOx, enzyme
      G(3)=omega*C2(2,J)
     1     -RXN1(1,J)
     2     +RXN4(1,J)
      B(3,4)=-omega
      B(3,17)=+1.
      B(3,23)=-1.

      G(4)=-omega*C2(1,J)
     1     -RXN1(2,J)
     2     +RXN4(2,J)
      B(4,3)=omega
      B(4,17)=+1.
      B(4,24)=-1.

C     For Gluconic Acid, being produced only
      G(5)=omega*C3(2,J)
     1     +POR1*DIFF(3)*(C3(1,J+1)-2.*C3(1,J)+C3(1,J-1))/HH**2.
```

```
     3      +RXN2(1,J)
       B(5,5)=2.*POR1*DIFF(3)/HH**2.
       A(5,5)=-POR1*DIFF(3)/HH**2.
       D(5,5)=-POR1*DIFF(3)/HH**2.
       B(5,6)=-omega
       B(5,19)=-1.

       G(6)=-omega*C3(1,J)
     1      +POR1*DIFF(3)*(C3(2,J+1)-2.*C3(2,J)+C3(2,J-1))/HH**2.
     3      +RXN2(2,J)
       B(6,6)=2.*POR1*DIFF(3)/HH**2.
       A(6,6)=-POR1*DIFF(3)/HH**2.
       D(6,6)=-POR1*DIFF(3)/HH**2.
       B(6,5)=omega
       B(6,20)=-1.

C      For GOx2, enzyme
       G(7)=omega*C4(2,J)
     1      +RXN2(1,J)
     2      -RXN3(1,J)
       B(7,8)=-omega
       B(7,19)=-1.
       B(7,21)=+1.

       G(8)=-omega*C4(1,J)
     1      +RXN2(2,J)
     2      -RXN3(2,J)
       B(8,7)=omega
       B(8,20)=-1.
       B(8,22)=+1.

C      For O2, being consumed only
       G(9)=omega*C5(2,J)
     1      +POR1*DIFF(5)*(C5(1,J+1)-2.*C5(1,J)+C5(1,J-1))/HH**2.
     3      -RXN3(1,J)
       B(9,9)=2.*POR1*DIFF(5)/HH**2.
       A(9,9)=-POR1*DIFF(5)/HH**2.
       D(9,9)=-POR1*DIFF(5)/HH**2.
       B(9,10)=-omega
       B(9,19)=+1.

       G(10)=-omega*C5(1,J)
     1      +POR1*DIFF(5)*(C5(2,J+1)-2.*C5(2,J)+C5(2,J-1))/HH**2.
     3      -RXN3(2,J)
       B(10,10)=2.*POR1*DIFF(5)/HH**2.
       A(10,10)=-POR1*DIFF(5)/HH**2.
       D(10,10)=-POR1*DIFF(5)/HH**2.
       B(10,9)=omega
       B(10,20)=+1.

C      For H2O2, reacting species
       G(11)=omega*C6(2,J)
     1      +POR1*DIFF(6)*(C6(1,J+1)-2.*C6(1,J)+C6(1,J-1))/HH**2.
     3      +RXN4(1,J)
       B(11,11)=2.*POR1*DIFF(6)/HH**2.
       A(11,11)=-POR1*DIFF(6)/HH**2.
       D(11,11)=-POR1*DIFF(6)/HH**2.
       B(11,12)=-omega
```

```
115         B(11,23)=-1.
116
117         G(12)=-omega*C6(1,J)
118        1      +POR1*DIFF(6)*(C6(2,J+1)-2.*C6(2,J)+C6(2,J-1))/HH**2.
119        3      +RXN4(2,J)
120         B(12,12)=2.*POR1*DIFF(6)/HH**2.
121         A(12,12)=-POR1*DIFF(6)/HH**2.
122         D(12,12)=-POR1*DIFF(6)/HH**2.
123         B(12,11)=omega
124         B(12,24)=-1.
125
126 C       For CX-GOx2, enzyme
127         G(13)=omega*C7(2,J)
128        1      +RXN1(1,J)
129        2      -RXN2(1,J)
130         B(13,14)=-omega
131         B(13,17)=-1.
132         B(13,19)=+1.
133
134         G(14)=-omega*C7(1,J)
135        1      +RXN1(2,J)
136        2      -RXN2(2,J)
137         B(14,13)=omega
138         B(14,18)=-1.
139         B(14,20)=+1.
140
141 C       For CX-GOx2, enzyme
142         G(15)=omega*C8(2,J)
143        1      +RXN3(1,J)
144        2      -RXN4(1,J)
145         B(15,16)=-omega
146         B(15,21)=-1.
147         B(15,23)=+1.
148
149         G(16)=-omega*C8(1,J)
150        1      +RXN3(2,J)
151        2      -RXN4(2,J)
152         B(16,15)=omega
153         B(16,22)=-1.
154         B(16,24)=+1.
155
156 C       REACTION1
157         G(17)=-RXN1(1,J)+ratef1*CONCSS(2,J)*C1(1,J)
158        1      +ratef1*CONCSS(1,J)*C2(1,J)
159        2      -C7(1,J)/equilib1
160         B(17,1)=-ratef1*CONCSS(2,J)
161         B(17,3)=-ratef1*CONCSS(1,J)
162         B(17,13)=+1./equilib1
163         B(17,17)=+1.
164
165         G(18)=-RXN1(2,J)+ratef1*CONCSS(2,J)*C1(2,J)
166        1      +ratef1*CONCSS(1,J)*C2(2,J)
167        2      -C7(2,J)/equilib1
168         B(18,2)=-ratef1*CONCSS(2,J)
169         B(18,4)=-ratef1*CONCSS(1,J)
170         B(18,14)=+1./equilib1
171         B(18,18)=+1.
172
```

```
C       REACTION2
        G(19)=-RXN2(1,J)+ratef2*C7(1,J)
        B(19,13)=-ratef2
        B(19,19)=+1.

        G(20)=-RXN2(2,J)+ratef2*C7(2,J)
        B(20,14)=-ratef2
        B(20,20)=+1.

C       REACTION3
        G(21)=-RXN3(1,J)+ratef3*CONCSS(4,J)*C5(1,J)
       1     +ratef3*CONCSS(5,J)*C4(1,J)
       2     -C8(1,J)/equilib3
        B(21,7)=-ratef3*CONCSS(5,J)
        B(21,9)=-ratef3*CONCSS(4,J)
        B(21,15)=+1./equilib3
        B(21,21)=+1.

        G(22)=-RXN3(2,J)+ratef3*CONCSS(5,J)*C4(2,J)
       1     +ratef3*CONCSS(5,J)*C4(2,J)
       2     -C8(2,J)/equilib3
        B(22,8)=-ratef3*CONCSS(5,J)
        B(22,10)=-ratef3*CONCSS(4,J)
        B(22,16)=+1./equilib3
        B(22,22)=+1.

C       REACTION4
        G(23)=-RXN4(1,J)+ratef4*C8(1,J)
        B(23,15)=-ratef4
        B(23,23)=+1.

        G(24)=-RXN4(2,J)+ratef4*C8(2,J)
        B(24,16)=-ratef4
        B(24,24)=+1.

c       SAVE G OUT DATA
        DO 11 I=2,13
   11   If (I.EQ.J) WRITE(14,301) J, (G(K),K=1,N)
        IF (J.EQ.IJ/2) THEN
        WRITE(14,301) J, (G(K),K=1,N)
        ELSE IF (J.EQ.(IJ-1)) THEN
        WRITE(14,301) J, (G(K),K=1,N)
        ELSE IF (J.EQ.(IJ-2)) THEN
        WRITE(14,301) J, (G(K),K=1,N)
        ELSE IF (J.EQ.(IJ-3)) THEN
        WRITE(14,301) J, (G(K),K=1,N)
        END IF

        RETURN
        END
```

Code E.15. Oscillating Continuous Glucose Monitor Subroutine for the Second Coupler

```
      SUBROUTINE COUPLER2(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(24,24),B(24,24),C(24,80001),D(24,49),G(24),
     1      X(24,24),Y(24,24)
      COMMON/NST/ N, NJ

      COMMON/VAR/ CONCSS(8,80001),RXNSS(4,80001),DIFF(8)
      COMMON/VARR/ PORGLU, HHH, KJ
      COMMON/CON/ C1(2,80001),C2(2,80001),C3(2,80001),C4(2,80001),
     1    C5(2,80001),C6(2,80001),C7(2,80001),C8(2,80001),
     2    RXN1(2,80001),RXN2(2,80001),RXN3(2,80001),RXN4(2,80001)
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/OTH/ POR1,POR2,H,EBIG,HH,IJ
      COMMON/BCI/ FLUX, omega
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT
      COMMON/DELT/ DELTA1, DELTA2, FREQ(400),CB(2010,80001)

  301 FORMAT (5x,'J=' I5,  24E15.6)

C     For Glucose, being consumed only
      G(1)=HH/2*omega*(C1(2,J+1)+3.*C1(2,J))/4.
     1     +H/2*omega*(C1(2,J-1)+3.*C1(2,J))/4.
     2     +PORGLU*DIFF(1)/H*(C1(1,J+1)-C1(1,J))
     3     -POR1*DIFF(1)/HH*(C1(1,J)-C1(1,J-1))
     5     -(HH/2.)*(RXN1(1,J-1)+3.*RXN1(1,J))/4.
      B(1,1)=PORGLU*DIFF(1)/H+POR1*DIFF(1)/HH
      D(1,1)=-PORGLU*DIFF(1)/H
      A(1,1)=-POR1*DIFF(1)/HH
      B(1,2)=-HH/2*omega*(3./4.)-H/2*omega*(3./4.)
      D(1,2)=-HH/2*omega*(1./4.)
      A(1,2)=-H/2*omega*(1./4.)
      B(1,17)=(HH/2.)*(3./4.)
      A(1,17)=(HH/2.)*(1./4.)

      G(2)=-HH/2*omega*(C1(1,J+1)+3.*C1(1,J))/4.
     1     -H/2*omega*(C1(1,J-1)+3.*C1(1,J))/4.
     2     +PORGLU*DIFF(1)/H*(C1(2,J+1)-C1(2,J))
     3     -POR1*DIFF(1)/HH*(C1(2,J)-C1(2,J-1))
     5     -(HH/2.)*(RXN1(2,J-1)+3.*RXN1(2,J))/4.
      B(2,2)=PORGLU*DIFF(1)/H+POR1*DIFF(1)/HH
      D(2,2)=-PORGLU*DIFF(1)/H
      A(2,2)=-POR1*DIFF(1)/HH
      B(2,1)=+HH/2*omega*(3./4.)+H/2*omega*(3./4.)
      D(2,1)=+HH/2*omega*(1./4.)
      A(2,1)=+H/2*omega*(1./4.)
      B(2,18)=(HH/2.)*(3./4.)
      A(2,18)=(HH/2.)*(1./4.)

C     For GOx, enzyme
      G(3)=HH/2*omega*(C2(2,J+1)+3.*C2(2,J))/4.
     1     +H/2*omega*(C2(2,J-1)+3.*C2(2,J))/4.
     3     -(HH/2.)*(RXN1(1,J-1)+3.*RXN1(1,J))/4.
     5     +(HH/2.)*(RXN4(1,J-1)+3.*RXN4(1,J))/4.
      B(3,4)=-HH/2*omega*(3./4.)-H/2*omega*(3./4.)
      D(3,4)=-HH/2*omega*(1./4.)
      A(3,4)=-H/2*omega*(1./4.)
```

```
57        B(3,17)=-(HH/2.)*(3./4.)
58        A(3,17)=-(HH/2.)*(1./4.)
59        B(3,23)=(HH/2.)*(3./4.)
60        A(3,23)=(HH/2.)*(1./4.)
61
62        G(4)=-HH/2*omega*(C2(1,J+1)+3.*C2(1,J))/4.
63       1      -H/2*omega*(C2(1,J-1)+3.*C2(1,J))/4.
64       3      -(HH/2.)*(RXN1(2,J-1)+3.*RXN1(2,J))/4.
65       5      +(HH/2.)*(RXN4(2,J-1)+3.*RXN4(2,J))/4.
66        B(4,3)=HH/2*omega*(3./4.)+H/2*omega*(3./4.)
67        D(4,3)=HH/2*omega*(1./4.)
68        A(4,3)=H/2*omega*(1./4.)
69        B(4,18)=-(HH/2.)*(3./4.)
70        A(4,18)=-(HH/2.)*(1./4.)
71        B(4,24)=(HH/2.)*(3./4.)
72        A(4,24)=(HH/2.)*(1./4.)
73
74 C      For Gluconic Acid, being produced only
75        G(5)=HH/2*omega*(C3(2,J+1)+3.*C3(2,J))/4.
76       1      +H/2*omega*(C3(2,J-1)+3.*C3(2,J))/4.
77       2      +PORGLU*DIFF(3)/H*(C3(1,J+1)-C3(1,J))
78       3      -POR1*DIFF(3)/HH*(C3(1,J)-C3(1,J-1))
79       5      +(HH/2.)*(RXN2(1,J-1)+3.*RXN2(1,J))/4.
80        B(5,5)=PORGLU*DIFF(3)/H+POR1*DIFF(3)/HH
81        D(5,5)=-PORGLU*DIFF(3)/H
82        A(5,5)=-POR1*DIFF(3)/HH
83        B(5,6)=-HH/2*omega*(3./4.)-H/2*omega*(3./4.)
84        D(5,6)=-HH/2*omega*(1./4.)
85        A(5,6)=-H/2*omega*(1./4.)
86        B(5,19)=-(HH/2.)*(3./4.)
87        A(5,19)=-(HH/2.)*(1./4.)
88
89        G(6)=-HH/2*omega*(C3(1,J+1)+3.*C3(1,J))/4.
90       1      -H/2*omega*(C3(1,J-1)+3.*C3(1,J))/4.
91       2      +PORGLU*DIFF(3)/H*(C3(2,J+1)-C3(2,J))
92       3      -POR1*DIFF(3)/HH*(C3(2,J)-C3(2,J-1))
93       5      +(HH/2.)*(RXN2(2,J-1)+3.*RXN2(2,J))/4.
94        B(6,6)=PORGLU*DIFF(3)/H+POR1*DIFF(3)/HH
95        D(6,6)=-PORGLU*DIFF(3)/H
96        A(6,6)=-POR1*DIFF(3)/HH
97        B(6,5)=+HH/2*omega*(3./4.)+H/2*omega*(3./4.)
98        D(6,5)=+HH/2*omega*(1./4.)
99        A(6,5)=+H/2*omega*(1./4.)
100       B(6,20)=-(HH/2.)*(3./4.)
101       A(6,20)=-(HH/2.)*(1./4.)
102
103 C      For GOx2, enzyme
104       G(7)=HH/2*omega*(C4(2,J+1)+3.*C4(2,J))/4.
105      1      +H/2*omega*(C4(2,J-1)+3.*C4(2,J))/4.
106      3      +(HH/2.)*(RXN2(1,J-1)+3.*RXN2(1,J))/4.
107      5      -(HH/2.)*(RXN3(1,J-1)+3.*RXN3(1,J))/4.
108       B(7,8)=-HH/2*omega*(3./4.)-H/2*omega*(3./4.)
109       D(7,8)=-HH/2*omega*(1./4.)
110       A(7,8)=-H/2*omega*(1./4.)
111       B(7,19)=(HH/2.)*(3./4.)
112       A(7,19)=(HH/2.)*(1./4.)
113       B(7,21)=-(HH/2.)*(3./4.)
114       A(7,21)=-(HH/2.)*(1./4.)
```

```
115
116       G(8)=-HH/2*omega*(C4(1,J+1)+3.*C4(1,J))/4.
117      1      -H/2*omega*(C4(1,J-1)+3.*C4(1,J))/4.
118      3      +(HH/2.)*(RXN2(2,J-1)+3.*RXN2(2,J))/4.
119      5      -(HH/2.)*(RXN3(2,J-1)+3.*RXN3(2,J))/4.
120       B(8,7)=HH/2*omega*(3./4.)+H/2*omega*(3./4.)
121       D(8,7)=HH/2*omega*(1./4.)
122       A(8,7)=H/2*omega*(1./4.)
123       B(8,20)=(HH/2.)*(3./4.)
124       A(8,20)=(HH/2.)*(1./4.)
125       B(8,22)=-(HH/2.)*(3./4.)
126       A(8,22)=-(HH/2.)*(1./4.)
127
128 C     For O2, being consumed only
129       G(9)=HH/2*omega*(C5(2,J+1)+3.*C5(2,J))/4.
130      1      +H/2*omega*(C5(2,J-1)+3.*C5(2,J))/4.
131      2      +POR2*DIFF(5)/H*(C5(1,J+1)-C5(1,J))
132      3      -POR1*DIFF(5)/HH*(C5(1,J)-C5(1,J-1))
133      5      -(HH/2.)*(RXN3(1,J-1)+3.*RXN3(1,J))/4.
134       B(9,9)=POR2*DIFF(5)/H+POR1*DIFF(5)/HH
135       D(9,9)=-POR2*DIFF(5)/H
136       A(9,9)=-POR1*DIFF(5)/HH
137       B(9,10)=-HH/2*omega*(3./4.)-H/2*omega*(3./4.)
138       D(9,10)=-HH/2*omega*(1./4.)
139       A(9,10)=-H/2*omega*(1./4.)
140       B(9,21)=(HH/2.)*(3./4.)
141       A(9,21)=(HH/2.)*(1./4.)
142
143       G(10)=-HH/2*omega*(C5(1,J+1)+3.*C5(1,J))/4.
144      1      -H/2*omega*(C5(1,J-1)+3.*C5(1,J))/4.
145      2      +POR2*DIFF(5)/H*(C5(2,J+1)-C5(2,J))
146      3      -POR1*DIFF(5)/HH*(C5(2,J)-C5(2,J-1))
147      5      -(HH/2.)*(RXN3(2,J-1)+3.*RXN3(2,J))/4.
148       B(10,10)=POR2*DIFF(5)/H+POR1*DIFF(5)/HH
149       D(10,10)=-POR2*DIFF(5)/H
150       A(10,10)=-POR1*DIFF(5)/HH
151       B(10,9)=+HH/2*omega*(3./4.)+H/2*omega*(3./4.)
152       D(10,9)=+HH/2*omega*(1./4.)
153       A(10,9)=+H/2*omega*(1./4.)
154       B(10,22)=(HH/2.)*(3./4.)
155       A(10,22)=(HH/2.)*(1./4.)
156
157 C     For H2O2, reacting species
158       G(11)=HH/2*omega*(C6(2,J+1)+3.*C6(2,J))/4.
159      1      +H/2*omega*(C6(2,J-1)+3.*C6(2,J))/4.
160      2      +POR2*DIFF(6)/H*(C6(1,J+1)-C6(1,J))
161      3      -POR1*DIFF(6)/HH*(C6(1,J)-C6(1,J-1))
162      5      +(HH/2.)*(RXN4(1,J-1)+3.*RXN4(1,J))/4.
163       B(11,11)=POR2*DIFF(6)/H+POR1*DIFF(6)/HH
164       D(11,11)=-POR2*DIFF(6)/H
165       A(11,11)=-POR1*DIFF(6)/HH
166       B(11,12)=-HH/2*omega*(3./4.)-H/2*omega*(3./4.)
167       D(11,12)=-HH/2*omega*(1./4.)
168       A(11,12)=-H/2*omega*(1./4.)
169       B(11,23)=-(HH/2.)*(3./4.)
170       A(11,23)=-(HH/2.)*(1./4.)
171
172       G(12)=-HH/2*omega*(C6(1,J+1)+3.*C6(1,J))/4.
```

```
     1      -H/2*omega*(C6(1,J-1)+3.*C6(1,J))/4.
     2      +POR2*DIFF(6)/H*(C6(2,J+1)-C6(2,J))
     3      -POR1*DIFF(6)/HH*(C6(2,J)-C6(2,J-1))
     5      +(HH/2.)*(RXN4(2,J-1)+3.*RXN4(2,J))/4.
      B(12,12)=POR2*DIFF(6)/H+POR1*DIFF(6)/HH
      D(12,12)=-POR2*DIFF(6)/H
      A(12,12)=-POR1*DIFF(6)/HH
      B(12,11)=+HH/2*omega*(3./4.)+H/2*omega*(3./4.)
      D(12,11)=+HH/2*omega*(1./4.)
      A(12,11)=+H/2*omega*(1./4.)
      B(12,24)=-(HH/2.)*(3./4.)
      A(12,24)=-(HH/2.)*(1./4.)

C     For CX-GOx2, enzyme
      G(13)=HH/2*omega*(C7(2,J+1)+3.*C7(2,J))/4.
     1      +H/2*omega*(C7(2,J-1)+3.*C7(2,J))/4.
     3      +(HH/2.)*(RXN1(1,J-1)+3.*RXN1(1,J))/4.
     5      -(HH/2.)*(RXN2(1,J-1)+3.*RXN2(1,J))/4.
      B(13,14)=-HH/2*omega*(3./4.)-H/2*omega*(3./4.)
      D(13,14)=-HH/2*omega*(1./4.)
      A(13,14)=-H/2*omega*(1./4.)
      B(13,17)=(HH/2.)*(3./4.)
      A(13,17)=(HH/2.)*(1./4.)
      B(13,19)=-(HH/2.)*(3./4.)
      A(13,19)=-(HH/2.)*(1./4.)

      G(14)=-HH/2*omega*(C7(1,J+1)+3.*C7(1,J))/4.
     1      -H/2*omega*(C7(1,J-1)+3.*C7(1,J))/4.
     3      +(HH/2.)*(RXN1(2,J-1)+3.*RXN1(2,J))/4.
     5      -(HH/2.)*(RXN2(2,J-1)+3.*RXN2(2,J))/4.
      B(14,13)=HH/2*omega*(3./4.)+H/2*omega*(3./4.)
      D(14,13)=HH/2*omega*(1./4.)
      A(14,13)=H/2*omega*(1./4.)
      B(14,18)=(HH/2.)*(3./4.)
      A(14,18)=(HH/2.)*(1./4.)
      B(14,20)=-(HH/2.)*(3./4.)
      A(14,20)=-(HH/2.)*(1./4.)

C     For CX-GOx, enzyme
      G(15)=HH/2*omega*(C8(2,J+1)+3.*C8(2,J))/4.
     1      +H/2*omega*(C8(2,J-1)+3.*C8(2,J))/4.
     3      +(HH/2.)*(RXN3(1,J-1)+3.*RXN3(1,J))/4.
     5      -(HH/2.)*(RXN4(1,J-1)+3.*RXN4(1,J))/4.
      B(15,16)=-HH/2*omega*(3./4.)-H/2*omega*(3./4.)
      D(15,16)=-HH/2*omega*(1./4.)
      A(15,16)=-H/2*omega*(1./4.)
      B(15,21)=(HH/2.)*(3./4.)
      A(15,21)=(HH/2.)*(1./4.)
      B(15,23)=-(HH/2.)*(3./4.)
      A(15,23)=-(HH/2.)*(1./4.)

      G(16)=-HH/2*omega*(C8(1,J+1)+3.*C8(1,J))/4.
     1      -H/2*omega*(C8(1,J-1)+3.*C8(1,J))/4.
     3      +(HH/2.)*(RXN3(2,J-1)+3.*RXN3(2,J))/4.
     5      -(HH/2.)*(RXN4(2,J-1)+3.*RXN4(2,J))/4.
      B(16,15)=HH/2*omega*(3./4.)+H/2*omega*(3./4.)
      D(16,15)=HH/2*omega*(1./4.)
      A(16,15)=H/2*omega*(1./4.)
```

```
231         B(16,22)=(HH/2.)*(3./4.)
232         A(16,22)=(HH/2.)*(1./4.)
233         B(16,24)=-(HH/2.)*(3./4.)
234         A(16,24)=-(HH/2.)*(1./4.)
235
236 C       REACTION1
237         G(17)=-RXN1(1,J)+ratef1*CONCSS(2,J)*C1(1,J)
238        1     +ratef1*CONCSS(1,J)*C2(1,J)
239        2     -C7(1,J)/equilib1
240         B(17,1)=-ratef1*CONCSS(2,J)
241         B(17,3)=-ratef1*CONCSS(1,J)
242         B(17,13)=+1./equilib1
243         B(17,17)=+1.
244
245         G(18)=-RXN1(2,J)+ratef1*CONCSS(2,J)*C1(2,J)
246        1     +ratef1*CONCSS(1,J)*C2(2,J)
247        2     -C7(2,J)/equilib1
248         B(18,2)=-ratef1*CONCSS(2,J)
249         B(18,4)=-ratef1*CONCSS(1,J)
250         B(18,14)=+1./equilib1
251         B(18,18)=+1.
252
253 C       REACTION2
254         G(19)=-RXN2(1,J)+ratef2*C7(1,J)
255         B(19,13)=-ratef2
256         B(19,19)=+1.
257
258         G(20)=-RXN2(2,J)+ratef2*C7(2,J)
259         B(20,14)=-ratef2
260         B(20,20)=+1.
261
262 C       REACTION3
263         G(21)=-RXN3(1,J)+ratef3*CONCSS(4,J)*C5(1,J)
264        1     +ratef3*CONCSS(5,J)*C4(1,J)
265        2     -C8(1,J)/equilib3
266         B(21,7)=-ratef3*CONCSS(5,J)
267         B(21,9)=-ratef3*CONCSS(4,J)
268         B(21,15)=+1./equilib3
269         B(21,21)=+1.
270
271         G(22)=-RXN3(2,J)+ratef3*CONCSS(5,J)*C4(2,J)
272        1     +ratef3*CONCSS(5,J)*C4(2,J)
273        2     -C8(2,J)/equilib3
274         B(22,8)=-ratef3*CONCSS(5,J)
275         B(22,10)=-ratef3*CONCSS(4,J)
276         B(22,16)=+1./equilib3
277         B(22,22)=+1.
278
279 C       REACTION4
280         G(23)=-RXN4(1,J)+ratef4*C8(1,J)
281         B(23,15)=-ratef4
282         B(23,23)=+1.
283
284         G(24)=-RXN4(2,J)+ratef4*C8(2,J)
285         B(24,16)=-ratef4
286         B(24,24)=+1.
287
288         WRITE(14,301) J, (G(K),K=1,N)
```

```
289
290         RETURN
291         END
```

Code E.16. Oscillating Continuous Glucose Monitor Subroutine for the GLM Region

```
      SUBROUTINE OUTER(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(24,24),B(24,24),C(24,80001),D(24,49),G(24),
     1      X(24,24),Y(24,24)
      COMMON/NST/ N, NJ

      COMMON/VAR/ CONCSS(8,80001),RXNSS(4,80001),DIFF(8)
      COMMON/VARR/ PORGLU, HHH, KJ
      COMMON/CON/ C1(2,80001),C2(2,80001),C3(2,80001),C4(2,80001),
     1    C5(2,80001),C6(2,80001),C7(2,80001),C8(2,80001),
     2    RXN1(2,80001),RXN2(2,80001),RXN3(2,80001),RXN4(2,80001)
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/OTH/ POR1,POR2,H,EBIG,HH,IJ
      COMMON/BCI/ FLUX, omega
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT
      COMMON/DELT/ DELTA1, DELTA2, FREQ(400),CB(2010,80001)

  301 FORMAT (5x,'J=' I5,  24E15.6)

C     For Glucose, being consumed only
      G(1)=omega*C1(2,J)
     1     +POR2*DIFF(1)*(C1(1,J+1)-2.*C1(1,J)+C1(1,J-1))/H**2.
      B(1,1)=2.*POR2*DIFF(1)/H**2.
      A(1,1)=-POR2*DIFF(1)/H**2.
      D(1,1)=-POR2*DIFF(1)/H**2.
      B(1,2)=-omega

      G(2)=-omega*C1(1,J)
     1     +POR2*DIFF(1)*(C1(2,J+1)-2.*C1(2,J)+C1(2,J-1))/H**2.
      B(2,2)=2.*POR2*DIFF(1)/H**2.
      A(2,2)=-POR2*DIFF(1)/H**2.
      D(2,2)=-POR2*DIFF(1)/H**2.
      B(2,1)=omega

C     For GOx, enzyme
      G(3)=C2(1,J)
      B(3,3)=-1.

      G(4)=C2(2,J)
      B(4,4)=-1.

C     For Gluconic Acid, being produced only
      G(5)=omega*C3(2,J)
     1     +POR2*DIFF(3)*(C3(1,J+1)-2.*C3(1,J)+C3(1,J-1))/H**2.
      B(5,5)=2.*POR2*DIFF(3)/H**2.
      A(5,5)=-POR2*DIFF(3)/H**2.
      D(5,5)=-POR2*DIFF(3)/H**2.
      B(5,6)=-omega

      G(6)=-omega*C3(1,J)
     1     +POR2*DIFF(3)*(C3(2,J+1)-2.*C3(2,J)+C3(2,J-1))/H**2.
      B(6,6)=2.*POR2*DIFF(3)/H**2.
      A(6,6)=-POR2*DIFF(3)/H**2.
      D(6,6)=-POR2*DIFF(3)/H**2.
      B(6,5)=omega

```

```
C       For GOx2, enzyme
        G(7)=C4(1,J)
        B(7,7)=-1.

        G(8)=C4(2,J)
        B(8,8)=-1.

C       For O2, being consumed only
        G(9)=omega*C5(2,J)
       1     +POR2*DIFF(5)*(C5(1,J+1)-2.*C5(1,J)+C5(1,J-1))/H**2.
        B(9,9)=2.*POR2*DIFF(5)/H**2.
        A(9,9)=-POR2*DIFF(5)/H**2.
        D(9,9)=-POR2*DIFF(5)/H**2.
        B(9,10)=-omega

        G(10)=-omega*C5(1,J)
       1     +POR2*DIFF(5)*(C5(2,J+1)-2.*C5(2,J)+C5(2,J-1))/H**2.
        B(10,10)=2.*POR2*DIFF(5)/H**2.
        A(10,10)=-POR2*DIFF(5)/H**2.
        D(10,10)=-POR2*DIFF(5)/H**2.
        B(10,9)=omega

C       For H2O2, reacting species
        G(11)=omega*C6(2,J)
       1     +POR2*DIFF(6)*(C6(1,J+1)-2.*C6(1,J)+C6(1,J-1))/H**2.
        B(11,11)=2.*POR2*DIFF(6)/H**2.
        A(11,11)=-POR2*DIFF(6)/H**2.
        D(11,11)=-POR2*DIFF(6)/H**2.
        B(11,12)=-omega

        G(12)=-omega*C6(1,J)
       1     +POR2*DIFF(6)*(C6(2,J+1)-2.*C6(2,J)+C6(2,J-1))/H**2.
        B(12,12)=2.*POR2*DIFF(6)/H**2.
        A(12,12)=-POR2*DIFF(6)/H**2.
        D(12,12)=-POR2*DIFF(6)/H**2.
        B(12,11)=omega

C       For CX-GOx2, enzyme
        G(13)=C7(1,J)
        B(13,13)=-1.

        G(14)=C7(2,J)
        B(14,14)=-1.

C       For CX-GOx, enzyme
        G(15)=C8(1,J)
        B(15,15)=-1.

        G(16)=C8(2,J)
        B(16,16)=-1.

C       REACTION1
        G(17)=-RXN1(1,J)
        B(17,17)=+1.

        G(18)=-RXN1(2,J)
        B(18,18)=+1.

```

```
C       REACTION2
        G(19)=-RXN2(1,J)
        B(19,19)=+1.

        G(20)=-RXN2(2,J)
        B(20,20)=+1.

C       REACTION3
        G(21)=-RXN3(1,J)
        B(21,21)=+1.

        G(22)=-RXN3(2,J)
        B(22,22)=+1.

C       REACTION4
        G(23)=-RXN4(1,J)
        B(23,23)=+1.

        G(24)=-RXN4(2,J)
        B(24,24)=+1.

c       SAVE G OUT DATA
        IF (J.EQ.(IJ+(NJ-IJ)/2)) THEN
        WRITE(14,301) J, (G(K),K=1,N)
        ELSE IF (J.EQ.(NJ-1)) THEN
        WRITE(14,301) J, (G(K),K=1,N)
        END IF

        RETURN
        END
```

Code E.17. Oscillating Continuous Glucose Monitor Subroutine for the Bulk Boundary Condition

```
      SUBROUTINE BCNJ(J)
      IMPLICIT DOUBLE PRECISION (A-H, O-Z)
      COMMON/BAT/ A(24,24),B(24,24),C(24,80001),D(24,49),G(24),
     1      X(24,24),Y(24,24)
      COMMON/NST/ N, NJ

      COMMON/VAR/ CONCSS(8,80001),RXNSS(4,80001),DIFF(8)
      COMMON/VARR/ PORGLU, HHH, KJ
      COMMON/CON/ C1(2,80001),C2(2,80001),C3(2,80001),C4(2,80001),
     1    C5(2,80001),C6(2,80001),C7(2,80001),C8(2,80001),
     2    RXN1(2,80001),RXN2(2,80001),RXN3(2,80001),RXN4(2,80001)
      COMMON/RTE/ ratef1,equilib1,ratef2,ratef3,equilib3,ratef4
      COMMON/OTH/ POR1,POR2,H,EBIG,HH,IJ
      COMMON/BCI/ FLUX, omega
      COMMON/BUL/ CBULK(8), PARH2O2, PAR02, PARGLUCOSE, JCOUNT
      COMMON/DELT/ DELTA1, DELTA2, FREQ(400),CB(2010,80001)

  301 FORMAT (5x,'J=' I5,  24E15.6)

      DO 42 I=1,2
C     For Glucose, being consumed only
      G(I)=C1(I,J)
      B(I,I)=-1.
C     For GOx, enzyme
      G(2+I)=C2(I,J)
      B(2+I,2+I)=-1.
C     For Gluconic Acid, being produced only
      G(4+I)=C3(I,J)
      B(4+I,4+I)=-1.
C     For GOx2, enzyme
      G(6+I)=C4(I,J)
      B(6+I,6+I)=-1.
C     For O2, being consumed only
      G(8+I)=C5(I,J)
      B(8+I,8+I)=-1.
C     For H2O2, reacting species
      G(10+I)=C6(I,J)
      B(10+I,10+I)=-1.
C     For CX-GOx2, enzyme
      G(12+I)=C7(I,J)
      B(12+I,12+I)=-1.
C     For CX-GOx, enzyme
      G(14+I)=C8(I,J)
   42 B(14+I,14+I)=-1.

C     REACTION1
      G(17)=-RXN1(1,J)
      B(17,17)=+1.

      G(18)=-RXN1(2,J)
      B(18,18)=+1.

C     REACTION2
      G(19)=-RXN2(1,J)
```

```
55          B(19,19)=+1.
56
57          G(20)=-RXN2(2,J)
58          B(20,20)=+1.
59
60  C       REACTION3
61          G(21)=-RXN3(1,J)
62          B(21,21)=+1.
63
64          G(22)=-RXN3(2,J)
65          B(22,22)=+1.
66
67  C       REACTION4
68          G(23)=-RXN4(1,J)
69          B(23,23)=+1.
70
71          G(24)=-RXN4(2,J)
72          B(24,24)=+1.
73
74          WRITE(14,301)  J,  (G(K),K=1,N)
75
76          RETURN
77          END
```

REFERENCES

[1] S.-L. Wu, *Influence of Disk Electrode Geometry on Local and Global Impedance Response*, Ph.D. dissertation, University of Florida, Gainesville, FL (2010).

[2] S.-L. Wu, M. E. Orazem, B. Tribollet, and V. Vivier, "The Influence of Coupled Faradaic and Charging Currents on Impedance Spectroscopy," *Electochimica Acta*, **131** (2014) 3–12.

[3] M. E. Orazem and B. Tribollet, *Electrochemical Impedance Spectroscopy (in press)*, 2nd edition (Hoboken, NJ: John Wiley & Sons, 2017).

[4] B. Tribollet and J. S. Newman, "Analytic Expression for the Warburg Impedance for a Rotating Disk Electrode," *J. Electrochem. Soc.*, **130** (1983) 822–824.

[5] M. E. Orazem and B. Tribollet, *Electrochemical Impedance Spectroscopy* (Hoboken, NJ: John Wiley & Sons, 2008).

[6] M. E. Orazem, *Mathematical Modeling and Optimization of Liquid-Junction Photovoltaic Cells*, Ph.D. dissertation, University of California, Berkeley, California (1983).

[7] C. Deslouis, I. Epelboin, M. Keddam, and J. C. Lestrade, "Impédance de Diffusion d'un Disque Tournant en Régime Hydrodynamique Laminaire. Etude Expérimentale et Comparaison avec le Modèle de Nernst," *J. Electroanal. Chem.*, **28** (1970) 57–63.

[8] D. Schumann, "Electrochimie.- Sur l'Impédance de Diffusion en Basse Fréquence," *Comptes Rendus Hebdomadaires des Séances de l'Académie des Sciences*, **262C** (1966) 624.

[9] Y. Filinovskii and V. A. Kir'yanov, "Contribution to Theory of Non-Stationary Convective Diffusion on Rotating Disc Electrode," *Dokl. Akad. Nauk SSSR*, **156** (1964) 1412.

[10] E. Levart and D. Schuhmann, "Sur la Détermination Générale de L'impédance de Concentration (Diffusion Convective et Réaction Chimique) pour une Electrode à Disque Tournant," *J. Electroanal. Chem.*, **53** (1974) 77–94.

[11] D. A. Scherson and J. S. Newman, "The Warburg Impedance in the Presence of Convective Flow," *J. Electrochem. Soc.*, **127** (1980) 110–113.

[12] E. Levart and D. Schuhmann, "Comparison of Some Solutions for the Warburg Impedance of a Rotating Disk," *J. Electrochem. Soc.*, **122** (1975) 1082.

[13] E. Levart and D. Schuhmann, "Discussion on: The Warburg Impedance in the Presence of Convective Flow by D. A. Scherson and J. Newman," *J. Electrochem. Soc.*, **127** (1980) 2649.

[14] C. Deslouis, C. Gabrielli, and B. Tribollet, "An Analytical Solution of the Nonsteady Convective Diffusion Equation for Rotating Electrodes," *J. Electrochem. Soc.*, **130** (1983) 2044–2046.

[15] B. Tribollet, J. S. Newman, and W. H. Smyrl, "Determination of the Diffusion Coefficient from Impedance Data in the Low Frequency Range," *J. Electrochem. Soc.*, **135** (1988) 134–138.

[16] V. Homsy and J. S. Newman, "An Asymptotic Solution for the Warburg Impedance of a Rotating Disk," *J. Electrochem. Soc.*, **121** (1974) 521.

[17] T. von Kármán, "Über Laminaire und Turbulente Reibung," *Zeitschrift für angewandte Mathematik und Mechanik*, **1** (1921) 233–252.

[18] W. G. Cochran, "The Flow Due to a Rotating Disc," *Proceedings of the Cambridge Philosophical Society*, **30** (1934) 365–375.

[19] H. Gerischer, "Concentration Polarization in Electrolytes After Previous Chemical Reaction and its Part in the Stationary Polarization Resistance at Equilibrium Potential," *Zeitschrift fr physikalische Chemie (Neue Folge)*, **197** (1951) 92–104.

[20] A. Lasia, *Electrochemical impedance spectroscopy and its applications* (Springer, 2014).

[21] V. G. Levich, *Physicochemical Hydrodynamics* (Englewood Cliffs, NJ: Prentice Hall, 1962).

[22] K. J. A. and L. V. G., "The use of a Rotating Disc Electrode in the Study of Electrochemical Kinetics and Electrolytic Processes," *Zhur. Fiz. Khim.*, **32** (1958) 1565–1675.

[23] K. J. A. and L. V. G., "An Application of a Rotating Disc Electrode to the Study of Kinetic and Catalytic Processes in Electrochemistry," *Doki. Akad. Nauk SSSR*, **117** (1957) 441–444.

[24] D. E. Smith, "Alternating Current Polargraphy of Electrode Processes with Coupled Homogeneous Reactions: I. Theory for systems with First-Order Preceding, Following and Catalytic Chemical Reactions," *Analyrical Chemistry*, **35** (1963) 602–609.

[25] D. E. Smith, "Alternating Current Polargraphy of Electrode Processes with Coupled Homogeneous Reactions: II. Experimental Results with Catalytic Reductions," *Analytical Chemistry*, **35** (1963) 610–614.

[26] R. Jurczakowski and P. Polczyǹski, "Impedance of Mediated Electrochemical Processes. Novel Impedance Element for Unequal Diffusivities." *Journal of Physical Chemistry C*, (2014).

[27] B. Timmer, M. Sluyters-Rehbach, and J. H. Sluyters, "On the Impedance of Galvanic Cells. XIX The Potential Dependence of the Faradaic Impedance in the case of

an Irrversible Electrode Reaction," *Electroanalytical Chemistry and Interfacial Electrochemistry*, **14** (1967) 181–191.

[28] B. Tribollet and J. S. Newman, "Impedance Model for a Concentrated Solution," *J. Electrochem. Soc.*, **131** (1984) 2780–2785.

[29] A. K. Hauser and J. S. Newman, "The Effect of Schmidt Number on the Faradaic Impedance of the Dissolution of a Copper Rotating Disk," *J. Electrochem. Soc.*, **136** (1989) 2896–2902.

[30] A. K. Hauser and J. S. Newman, "Potential and Concentration Variations of a Reacting, Supporting Electrolyte," *J. Electrochem. Soc.*, **136** (1989) 3319–3325.

[31] B. Bossche, L. Bortler, J. Deconinck, S. Vandeputte, and A. Hubin, "Quasi-One-Dimensional Steady-State Analysis of Multi-Ion Electrochemical Systems at a Rotating Disc Electrode Controlled by Diffusion, Migration, Convection and Homogeneous Reactions," *Journal of Electroanalytical Chemistry*, **397** (1995) 35–44.

[32] C. Deslouis, I. Grateur, G. Maurin, and B. Tribollet, "Interfacial pH Measurement During the Reduction of Dissolved Oxygen in a Submered Impinging Jet Cell," *Journal of Applied Electrochemistry*, **27** (1997) 482–492.

[33] E. Remita, B. Tribollet, E. Sutter, V. Vivier, F. Ropital, and J. Kittel, "Hydrogen Evolution in Aqueous Solutions Containing Dissolved CO_2: Quantitative Contribution of the Buffering Effect," *Corros. Sci.*, **50** (2008) 1433–1440.

[34] J. Vazquez-Arenas and M. Pritzker, "Transient and Steady-State Model of Cobalt Deposition in Borate-Sulfate Solutions," *Electrochimica Acta*, **55** (2010) 8376 – 8387.

[35] J. Vazquez-Arenas and M. Pritzker, "Comprehensive Impedance Model of Cobalt Deposition in Sulfate Solutions Accounting for Homogeneous Reactions and Adsorptive Effects," *Electrochimica Acta*, **56** (2011) 8023 – 8033.

[36] R. A. Thomas W. Chapman, "The effect of an irreversible homogeneous reaction on finite-layer diffusion impedance," *Electochimica Acta*, **56** (2010) 128–132.

[37] T. Tran, B. Brown, S. Nešiú, and B. Tribollet, "Investigation of the Electrochemical Mechanisms for Acedic Acid Corrosion of Mild Steel," *Corrosion Science*, **70** (2014) 223–229.

[38] S. B. Adler, "Factors Governing Oxygen Reduction in Solid Oxide Fuel Cell Cathodes," *Chemical Reviews*, **104** (2004) 4791–4843.

[39] S. B. Adler, J. A. Lane, and B. C. H. Steele, "Electrode Kinetics of Porous Mixed-Conducting Oxygen Electrodes," *Journal of The Electrochemical Society*, **143** (1996) 3554–3564.

[40] S. B. Adler, "Mechanism and Kinetics of Oxygen Reduction on Porous La_{1-x} Sr_x $CoO_{3-\delta}$ Electrodes," *Solid State Ionics*, **111** (1998) 125–134.

[41] R. Schiller, J. Balog, and G. Nagy, "Continuous-time Random-walk Theory of Interfering Diffusion and Chemical Reaction with an Application to Electrochemical Impedance Spectra of Oxidized Zr1Nb," *The Journal of Chemical Physics*, **123** (2005).

[42] A. B. Boukamp and H. Bouwmeester, "Interpretation of the Gerischer Impedance in Solid State Ionics," *Solid State Ionics*, **157** (2003) 29–33.

[43] R. Kumar and R. Kant, "Theory of Generalized Gerischer Admittance of Realistic Fractal Electrode," *The Journal of Physical Chemistry C*, **113** (2009) 19558–19567.

[44] A. Jukic and M. Metikoŏ-Hukovic, "The Hydrogen Evolution Reaction on Pure and Polypyrrole-Coated GdNi4Al Electrodes," *Electrochimica Acta*, **48** (2003) 3929 – 3937. Electrocatalysis:From Theory to Industrial Applications.

[45] N. Anicet, C. Bourdillon, C. Demaille, J. Moiroux, and J.-M. Savant, "Catalysis of the Electrochemical Oxidation of Glucose by Glucose Oxidase and a Single Electron Cosubstrate: Kinetics in Viscous Solutions," *Journal of Electroanalytical Chemistry*, **410** (1996) 199 – 202.

[46] S. Updike and G. Hicks, "The Enzyme Electrode," *Nature*, **214** (1967) 986–988.

[47] A. Heller and B. Feldman, "Electrochemical Glucose Sensors and their Applications in Diabetes Management," *Chemical Reviews*, **108** (2008) 2482–2505.

[48] D. Müller, "Studies on the new enzyme glucoseoxidase," *Biochem. Z*, **199** (1928).

[49] C. E. Coulthard, R. Michaelis, W. F. Short, G. Sykes, G. E. H. Skrimshire, A. F. B. Standfast, J. H. Birkinshaw, and H. Raistrick, "Notatin: An Anti-bacterial Glucose-aerodehydrogenase from Penicillium notatum Westling," *Nature*, **150** (1942) 634.

[50] C. E. Coulthard, R. Michaelis, W. F. Short, G. Sykes, G. E. H. Skrimshire, A. F. B. Standfast, J. H. Birkinshaw, and H. Raistrick, "Notatin: an anti-bacterial glucose-aerodehydrogenase from Penicillium notatum Westling and Penicillium resticulosum sp. nov." *Biochem Journal*, **35** (1945) 24.

[51] W. Franke and F. Lorenz, "Glucose oxidase," *Liebigs Annalen Der Chemie*, **531** (1937).

[52] R. Bentley and A. Neuberger, "The mechanism of the action of notatin," *Biochemical Journal*, **45** (1949) 584–590.

[53] T. Nakamura and Y. Ogura, "Kinetic Studies on the Action of Glucose Oxidase," *The Journal of Biochemistry*, **52** (1962) 214.

[54] Q. H. Gibson, B. E. P. Swoboda, and V. Massey, "Kinetics and Mechanism of Action of Glucose Oxidase," *The Journal of Biological Chemistry*, **239** (1964) 3927.

[55] H. Bright and Q. Gibson, "The Oxidation of 1-Deuterated Glucose by Glucose Oxidase," *The Journal of Biological Chemistry*, **242** (1967) 994–1003.

[56] H. J. Bright and M. Appleby, "The pH Dependence of the Individual Steps in the Glucose Oxidase Reaction," *The Journal of Biological Chemistry*, **244** (1969) 3625–3634.

[57] S. B. Bankar, M. V. Bule, R. S. Singhal, and L. Ananthanarayan, "Glucose OxidaseAn Overview," *Biotechnology Advances*, **27** (2009) 489–501.

[58] D. A. Gough and J. K. Leypoldt, "Transient Studies of Glucose, Oxygen, and Hydroquinone at a Membrane-Covered Rotated Disk Electrode," *Journal of The Electrochemical Society*, **127** (1980) 1278–1286.

[59] D. A. Gough and J. K. Leypoldt, "Membrane-Covered, Rotated Disc Electrode," *ANALYTICAL CHEMISTRY*, **51** (1979) 439–444.

[60] S. van Stroe-Biezen, J. van der Loo, L. Janssen, and F. Everaerts, "Determination of the Inherent Kinetics of Immobilized Glucose Oxidase Using a Diffusion Cell," *Bioprocess Engineering 15*, (1996) 87–94.

[61] S. van Stroe-Biezen, F. Everaerts, L. Janssen, and R. Tacken, "Diffusion Coefficients of Oxygen, Hydrogen Peroxide and Glucose in a Hydrogel," *Analytica Chimica Ataca*, **273** (1993) 553–560.

[62] P. Han and D. M. Bartels, "Temperature Dependence of Oxygen Diffusion in H2O and D2O," *The Journal of Physical Chemistry*, **100** (1996) 5597–5602.

[63] Y. Degani and A. Heller, "Direct Electrical Communication Between Chemically Modified Enzymes and Metal Electrodes. I. Electron Transfer From Glucose Oxidase to Metal Electrodes via Electron Relays, Bound Covalently to the Enzyme," *The Journal of Physical Chemistry*, **91** (1987) 1285–1289.

[64] A. M. Albisser, B. S. Leibel, T. G. Ewart, Z. Davidovac, C. K. Botz, and W. Zingg, "Control of diabetes by an artificial pancreas," *Diabetes*, **23** (1974) 389396.

[65] U. Hoss, E. S. Budiman, H. Liu, and M. P. Christiansen, "Continuous Glucose Monitoring in the Subcutaneous Tissue over a 14-Day Sensor Wear Period," *Journal of Diabetes Science and Technology*, **7** (2013) 1210–1219.

[66] J. J. Chamberlain and D. Small, "Case Study: Successful Use of a Single Subcutaneous Continuous Glucose Monitor Sensor for 28 Days in a Patient with Type 1 Diabetes," *Clin. Diabetes*, **26** (2008) 138–139.

[67] C. Zhao and Y. Fu, "Statistical analysis based online sensor failure detection for continuous glucose monitoring in type I diabetes," *Chemometrics and Intelligent Laboratory Systems*, **144** (2015) 128 – 137.

[68] U. Klueh, J. Frailey, Y. Qiao, O. Antar, and D. L. Kreutzer, "Cell Based Metabolic Barriers to Glucose Diffusion: Macrophages and Continuous Glucose Monitoring," *Biomaterials*, **35** (2014) 3145–3153.

[69] M. T. Novak, F. Yuan, and W. M. Reichert, "Modeling the Relative Impact of Capsular Tissue Effects on Implanted Glucose Sensor Time Lag and Signal Attenuation," *Anal. Bioanal. Chem.*, **398** (2010) 1695–1705.

[70] U. Klueh, Z. Liu, T. Ouyang, B. Cho, B. Feldman, T. P. Henning, and D. Kreutzer, "Blood-Induced Interference of Glucose Sensor Function in Vitro: Implications for in Vivo Sensor Function," *Journal of Diabetes Science and Technology*, **1** (2007) 842–849.

[71] E. Katz and I. Willner, "Probing Biomolecular Interactions at Conductive and Semiconductive Surfaces by Impedance Spectroscopy: Routes to Impedimetric Immunosensors, DNA-Sensors, and Enzyme Biosensors. Electroanalysis," *Electroanalysis*, **15** (2003) 913–947.

[72] K. J. Otto, M. D. Johnson, and D. R. Kipke, "Voltage Pulses Change Neural Interface Properties and Improve Unit Recordings With Chronically Implanted Microelectrodes," *IEEE Transactions on Biomedical Engineering*, **53** (2006) 333–340.

[73] V. Sankar, E. Patrick, R. Dieme, J. C. Sanchez, A. Prasad, and T. Nishida, "Electrode impedance analysis of chronic tungsten microwire neural implants: understanding abiotic vs. biotic contributions," *Frontiers in Neuroengineering*, **7** (2014) 1.

[74] J. S. Newman and K. E. Thomas-Alyea, *Electrochemical Systems*, 3rd edition (Hoboken, NJ: John Wiley & Sons, 2004).

[75] D.-T. Chin and C. Tsang, "Mass Transfer to an Impinging Jet Electrode," *J. Electrochem. Soc.*, **125** (1978) 1461–1470.

[76] J. M. Esteban, G. Hickey, and M. E. Orazem, "The Impinging Jet Electrode: Measurement of the Hydrodynamic Constant and Its Use for Evaluating Film Persistency," *Corrosion*, **46** (1990) 896–901.

[77] C. B. Diem and M. E. Orazem, "The Influence of Velocity on the Corrosion of Copper in Alkaline Chloride Solutions," *Corrosion*, **50** (1994) 290–300.

[78] O. Devos, C. Gabrielli, and B. Tribollet, "Nucleation-Growth Processes of Scale Crystallization under Electrochemical Reaction Investigated by In Situ Microscopy," *Electrochem. Solid-State Lett.*, **4** (2001) C73–C76.

[79] M. E. Orazem, J. C. Cardoso, Filho, and B. Tribollet, "Application of a Submerged Impinging Jet for Corrosion Studies: Development of Models for the Impedance Response," *Electrochim. Acta*, **46** (2001) 3685–3698.

[80] H. Cachet, O. Devos, G. Folcher, C. Gabrielli, H. Perrot, and B. Tribollet, "In situ Investigation of Crystallization Processes by Coupling of Electrochemical and Optical Measurements: Application to $CaCO_3$ Deposit," *Electrochem. Solid-State Lett.*, **4** (2001) C23–C25.

[81] S. Witt, G. Wohlfahrt, D. Schomburg, H. J. Hecht, and H. M. Kalisz, "Conserved arginine-516 of Penicillium amagasakiense glucose oxidase is essential for the efficient binding of beta-D-glucose." *Biochemical Journal*, **347** (2000) 553–559.

[82] G. J. Brug, A. L. G. van den Eeden, M. Sluyters-Rehbach, and J. H. Sluyters, "The Analysis of Electrode Impedances Complicated by the Presence of a Constant Phase Element," *J. Electroanal. Chem.*, **176** (1984) 275–295.

[83] Z. Lukacs, "The Numerical Evaluation of the Distortion of EIS Data Due to the Distribution of Parameters," *J. Electroanal. Chem.*, **432** (1997) 79–83.

[84] L. Young, "Anodic Oxide Films 4: The Interpretation of Impedance Measurements on Oxide Coated Electrodes on Niobium," *Trans. Farad. Soc.*, **51** (1955) 1250–1260.

[85] R. Jurczakowski, C. Hitz, and A. Lasia, "Impedance of Porous Au-Based Electrodes," *J. Electroanal. Chem.*, **572** (2004) 355–366.

[86] J. H. Sluyters, "On the Impedance of Galvanic Cells I. Theory," *Recueil des Travaux Chimiques des Pays-Bas*, **79** (1960) 1092–1100.

[87] P. Delahay, "Electrode Processes without a Priori Separation of Double-Layer Charging," *J. Phys. Chem.*, **70** (1966) 2373–2379.

[88] P. Delahay and G. G. Susbielle, "Double-Layer Impedance of Electrodes with Charge-Transfer Reaction," *J. Phys. Chem.*, **70** (1966) 3150–3157.

[89] P. Delahay, K. Holub, G. G. Susbielle, and G. Tessari, "Double-Layer Perturbation without Equilibrium between Concentrations and Potential," *J. Phys. Chem.*, **71** (1967) 779–780.

[90] M. Sluyters-Rehbach, "Impedances of Electrochemical Systems: Terminology, Nomenclature, and Representation: I. Cells with Metal Electrodes and Liquid Solutions," *Pure Appl. Chem.*, **66** (1994) 1831–1891.

[91] E. Barsoukov and J. R. Macdonald, *Impedance Spectroscopy: Theory, Experiment, and Applications*, 2nd edition (John Wiley & Sons, 2005).

[92] K. Nisancioglu and J. S. Newman, "Separation of Double-Layer Charging and Faradaic Processes at Electrodes," *J. Electrochem. Soc.*, **159** (2012) E–59–E61.

[93] R. E. White, "On Newmans Numerical Technique for Solving Boundary Value Problems," *Ind. Eng. Chem. Fundam.*, **17** (1978) 367–369.

[94] D. A. Curtis, T. I. Evans, and R. E. White, "A Comparison of Newman's Numerical Technique and deBoor's Algorithm," *J. Electrochem. Soc.*, **136** (1989) 3392–3393.

BIOGRAPHICAL SKETCH

Morgan Harding received her bachelors of Chemical Engineering at the University of Idaho in May 2012. She then became a Ph.D. student at the University of Florida studying Electrochemical Impedance Spectroscopy in Professor Mark Orazems group. She has been a very active student as a member and leader of many student organizations at the University of Florida.