# Algorithm Development in Computational Electrochemistry


**A thesis submitted for the degree of D.Phil. in Physical and Theoretical Chemistry**

**Ian James Cutress**

**August 2011**

Algorithm Development in Computational Electrochemistry

A thesis submitted for the degree of D.Phil.
in Physical and Theoretical Chemistry

Ian James Cutress
August 2011

---

## Abstract

This thesis presents algorithm development in computational chemistry, and applies new computer science concepts to voltammetric simulation. To begin, this thesis discusses why algorithm development is necessary, and inherent problems found in commercial simulation solvers. As a result of this discussion, this thesis describes the need for simulators to keep abreast of recent computational developments.

Algorithm development in this thesis is taken through stages.

Chapter 3 applies known theory relating to the stripping voltammetry at a macroelectrode to the diffusional model of a microdisk, using finite difference and alternating direction implicit simulation techniques.

Chapter 4 introduces the concept of parallel computing, and how computational hardware has developed recently to take advantage of out-of-order calculations, by processing them in parallel to reduce simulation time. The novel area of graphics card simulation for highly parallel algorithms is also explained in detail.

Chapter 5 discusses the adaptation of voltammetric finite difference algorithms to a purely parallel format for simulation by explicit solution. Through explicit solution, finite difference algorithms are applied to electrode geometries which necessitate a three-

dimensional solution – elliptical electrodes; square, rectangular, and microband electrodes; and dual microdisk electrodes in collector-generator mode.

Chapter 6 introduces 'Random Walk' simulations, whereby individual particles in the simulation are modelled and their trajectories over time are calculated. The random walk technique in this thesis is improved for pure three-dimensional diffusion, and adapted to graphics cards, allowing up to a factor 4000 increase in speed over previous computational methods. This method is adapted to various systems of low concentration confined voltammetry (chapter 6.4) and single molecule detection, ultra low concentration cyclic voltammetry (chapter 6.5), and underpotential deposition of thallium on mobile silver nanoparticles (chapter 6.6).

Overall, this thesis presents, and applies, a series of algorithm development concepts in computational electrochemistry.

**Acknowledgements**

My thanks extends to the co-authors of all the publications from the research performed – in particular to Dr. Frank Marken for keeping a cool head when things weren't going to plan, Dr. Neil Rees, and to Edmund Dickinson for providing substantial insight throughout. Professor Richard Compton, the supervisor of this thesis, deserves a substantial amount of compensation for having to deal with the ranting of a madman (me), as well as my insistence for inserting idiosyncratic language for publications.

I would also like to give thanks to members of the research group for providing substantial social interaction as well as encouragement in all areas work ethic related – Kris Ward, Sven Ernst, Rahmat Wibowo, Barbara Kozub, and Emma Rogers (and her fiancé, Dan Corcoran).

Josephine Scoble deserves a special mention for being a very supportive friend throughout, and being the supplier of many needed alcoholic beverages.

**Contents**

## Symbols, Constants and Dimensionless Variables

All symbols, constants and dimensionless variables used in the main text are listed here for quick reference.

### Symbols

| | |
|---|---|
| T | Absolute Temperature |
| $[c]$ | Concentration of species $c$ |
| $[c]_0$ | Concentration of species $c$ at the electrode |
| $[c]_x$ | Concentration of species $c$ at $x$ |
| $[c]_{bulk}$ | Concentration of species $c$ in bulk solution |
| $\nabla^2 \; or \; \Delta$ | Laplacian operator, defining system geometry |
| $D_c$ | Diffusion coefficient of species $c$ |
| $\boldsymbol{\nu}$ | Velocity profile |
| $z_c$ | Charge of species $c$ |
| $\nabla\phi$ | Potential field |
| $j$ | Diffusional flux at an electrode surface |
| $k$ | Rate constant |
| $E_f^0$ | Formal potential |
| $\alpha$ | Butler-Volmer transfer coefficient |
| $i$ | Current at the electrode |
| $A$ | Surface area of the electrode |
| $\Delta G^{\ddagger}$ | Gibbs energy of activation |
| $\Delta G_i^{\ddagger}$ | Activation energy from the distortion of the inner coordination shell |
| $\Delta G_0^{\ddagger}$ | Rearrangement energy from solvent dipoles between the initial state and the transition state |
| $\lambda$ | Reorganisation energy (Marcus-Hush kinetics) |
| $I_{lim}$ | Limiting current at a microelectrode |
| $n$ | Effective number of electrons transferred |
| $r$ | Radius of the microdisk |

### Constants

| | |
|---|---|
| F | Faraday Constant, 96485.3415 coulombs per mole of electrons |
| R | Gas Constant, 8.314462 joules per Kelvin per mole |

## Chapter 1: Introduction

### 1.1 Why Simulate Voltammetry?

The ability to simulate the current response of reduction-oxidation couple at an electroactive surface as a function of the applied potential is a key process in understanding or determining the mechanistic information behind an electron transfer from either electrode to species or vice versa. Computational voltammetry is now a key tool in an electrochemists arsenal. It is now possible to cite significant numbers of publications dedicated to the development and improvement of such algorithms and simulation from the 1960s and every decade to the present day [1-6].

A most recent example to represent the strengths of electrochemical and voltammetric simulation is the determination of the residence time required for thallium monolayer under-potential deposition on to silver nano-particle (AgNP) spheres when in contact with the electrode surface [7]. Experimentally, the AgNPs diffuse in solution until they come into contact with the electrode, and will stay on the electrode for a 'residence' time. During this time, thallium(I) ions can deposit on the NP surface. Through simulation, a reasonable estimate into the required residence time for varying levels of monolayer coverage was made, depending on the concentration of the thallium(I) in solution and the size of the NP.

Also recently, simulation of voltammetry aided in the determination of the reaction pathway for the reduction of anthraquinone monosulphate (AQ) [8], for understanding the quinone-mediated reduction of oxygen. In this work, the reaction pathway from AQ to anthrahydraquinone ($AQH_2$) is highly pH dependent, involving both electron transfers

and chemical reactions of intermediates. Through simulation of the theorised mechanism and comparisons made to the experimental data, the scheme of reactions was produced and confirmed.

### 1.2 Simulation Concepts – Mass Transport

Considerable effort has been expended in attempting to effectively model the three areas of simulation electrochemistry – diffusion (the ability of a species to move through a medium) [9-14], convection (the movement of a species due to external forces or natural gradients) [15-17] and migration (the attraction of charged species to its surrounding environment) [18-21]. This is often mathematically represented as a partial differential equation:

$$\frac{d[c]}{dt} = diffusion + convection + migration \quad [1.1]$$

$$diffusion = D_c \nabla^2 [\mathrm{c}] \quad [1.2]$$

$$convection = [c]\boldsymbol{\nu} \quad [1.3]$$

$$migration = z_c \frac{F}{RT} D_c [c] \nabla\phi \quad [1.4]$$

where species $c$ is of interest, $D_c$ is the diffusion coefficient of species $c$ in the medium of interest (water, organic solvent, matrix), $[c]$ is the concentration of species $c$, $\boldsymbol{\nu}$ is the applied velocity profile to the matrix, $z_c$ is the charge on species $c$, $\nabla\phi$ is the potential field, and $F, R, T$ are standard electrochemical constants as defined in the Symbols and Constants section at the beginning of this thesis.

### 1.2.1 Diffusion

The act of diffusion is the movement of a substance to equalise its concentration (strictly speaking, activity) within a volume, and its movement from an area of high concentration to low concentration. Ideal examples are gases diffusing in another gas, or a solute in a solvent. However, even if the concentration is equal, particles will of course still move but no net concentration change will take place. Fick's first law (equation 1.5, [22]) describes the diffusive flux of the concentration of a species at any point $x$, where $j$ is the net flux in moles per square centimetre per second (in SI units), $\frac{\partial c}{\partial x}$ is the local concentration gradient at point $x$ and $D$ is the diffusion coefficient.

$$j = -D\frac{\partial c}{\partial x} \qquad [1.5]$$

The diffusion of a species in solution will govern the form of interactions that species takes – either with a boundary condition (electrode, insulator) or another species with the potential for chemical reaction or electron transfer. The mode of diffusion to a surface is determined as a result of the geometry of the surface itself – at large macro-sized (>150 μm radius) electrodes, diffusion to the surface is perpendicular, , as shown in figure 1.1. At micro-electrodes, at short times only, diffusion to the surface also considered perpendicular, as radial diffusion due to the edge effects is not a significant contributor to the concentration flux at the electrode surface, as shown in figure 1.2.

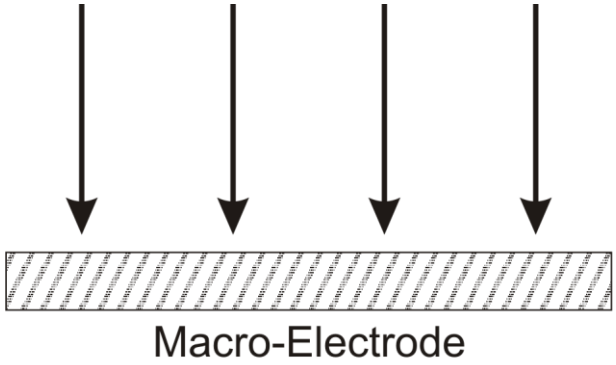



Figure 1.1 – Diffusion mode to a macro-electrode

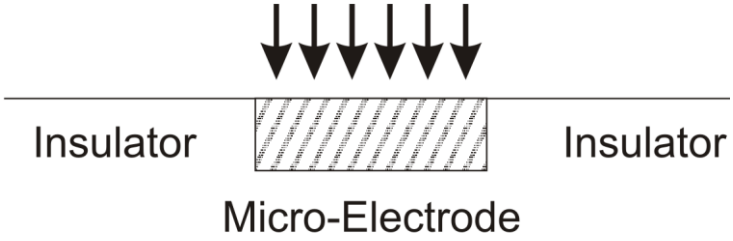



Figure 1.2 – Diffusion mode to a micro-electrode at short (Cottrellian) time scales

In the case of micro-electrodes, the planar diffusion concept at short times is known as the Cottrellian diffusion regime, after Frederick G. Cottrell (1877-1948) who first considered the planar diffusion mode at macro-electrodes in 1908 [23].

At longer time scales for micro-electrodes, the radial diffusion mode due to the edges of the electrode become a significant component of the concentration flux at the electrode surface, as material outside the perpendicular volume of the electrode has had sufficient time to diffuse through solution towards the electrode surface, as shown in figure 1.3.

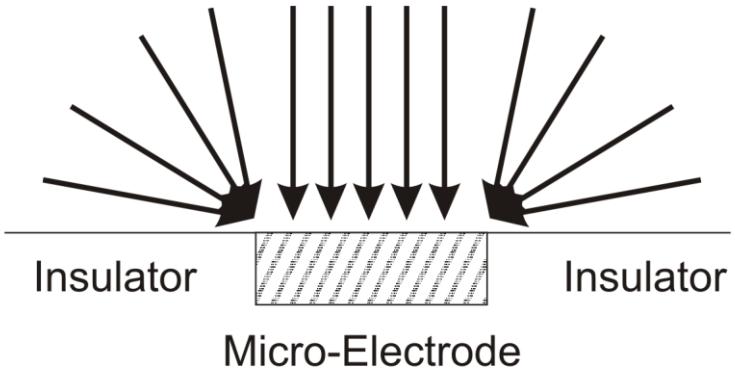



Figure 1.3 – Radial Diffusion mode to a micro-electrode at long time scales

### 1.2.2 Convection

Convection within a solution effects mass transport to the electrode surface by the application of a velocity field. The source of this velocity field can either be natural or forced: a common source of natural convection is a temperature gradient generated by exothermic or endothermic chemical reactions [24], whereas forced convection could result from a forced flow system [25-29], or applied microwave radiation, causing a localised temperature gradient [30-32]. Hydrodynamic systems, such as rotating electrodes [33-35] or sonoelectrochemistry [36,37], are also convection systems.

In terms of simulation, forced flow and hydrodynamic systems have been readily studied [38-48]. Systems such as these are more amenable to simulation within reasonable timescales due to their fixed velocity profiles irrespective of time. An example of this is the channel flow cell, which is a two dimension limited system, where the solution matrix is pumped at a constant rate over an electrode, as shown in figure 1.4. The velocity profile of the matrix is dependent on the pumping method used, and the surface tension of the matrix.

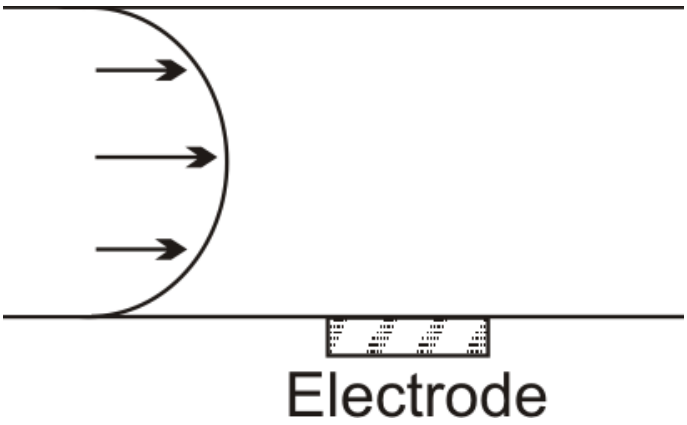



Figure 1.4 – Channel flow cell with velocity profile

### 1.2.3 Migration

As a result of charge transfer at the electrode surface, or between ions or metals in solution, a strong potential gradient will occur. The solution will rapidly move to dissipate the resultant electric field – this has a large effect in terms of transport of species to the electrode surface. If the charge is unable to fully disperse, distortions to voltammetry are observed due to the potential drop in solution – this is often negated by adding a sufficient quantity of 'supporting electrolyte' to the solution to better disperse the charge. This is the basis of a considerable number of electrochemical and voltammetric simulations, however migration-diffusion simulations involving relevant theory have been applied [49-55].

### 1.2.4 Application to Simulation

Simulations which deal with diffusion, convection and migration all together are often highly limited due to computational complexity and the accuracy of the understanding of the processes themselves. Typically, the effects of convection and migration are assumed to be negligible unless they are specifically being studied – by using this assumption, a

greater number of systems are open to be studied computationally in sufficient timescales.

### 1.3 Simulation Concepts – Electrochemical Mechanisms

Within the electrochemical setup, a species perform various interactions – the two main ones fitting into either the category of electron transfer, or the category of a chemical reaction.

In terms of electron transfer, a species could transfer electrons between itself and the electrode, transfer electrons between itself and other species in solution, or transfer electrons within itself (for example, multi-metal clusters or charge distributed organic molecules).

With regards to chemical reactivity, a species could react in solution with other species, or isomerise to form new species, or could act as a catalyst in another chemical reaction between one or many other species, or potentially dissociate by itself into two (or more) other species.

Each of these steps, either electrochemical or chemical, can be considered as a fully reversible processes, quasi-reversible processes, or as irreversible processes, and each require specific treatment in simulation.

### 1.3.1 Notation

Testa and Reinmuth pioneered an easy-to-use notation for describing similar processes in electrochemistry [56]. For a reductive or oxidative heterogeneous electron transfer at an electrode surface, either reversible or irreversible (with a rate constant $k$), is denoted **E**:

$$A \overset{k}{\rightarrow} B \pm e^- \qquad [1.6]$$

For homogeneous stoichiometrically equivalent chemical reactions in solution, these are donated **C**:

$$A \overset{k}{\rightarrow} B \qquad [1.7]$$

$$A + B \overset{k}{\rightarrow} C \qquad [1.8]$$

Since the Testa and Reinmuth initial work, derivations of the notation are now commonplace in the literature. A stoichiometrically unbalanced chemical reaction whereby two of the same reactant produces a single product is labelled as $\mathbf{C_2}$:

$$2A \overset{k}{\rightarrow} B \qquad [1.9]$$

A catalytic chemical reaction where the initial reactant is regenerated is labelled as **C'** (C prime). Equation 1.10(ab) shows an EC' reaction where species S causes the regeneration of species A from species B:

$$A + e^- \rightarrow B \qquad [1.10a]$$

$$B + S \rightarrow A \qquad [1.10b]$$

Under the ECE mechanism, an electron transfer occurs, followed by a chemical reaction and a subsequent electron transfer. However, consider the example of disproportionation – if the first electron transfer if followed immediately by a chemical step, and the second electron transfer is thermodynamically more favourable than the first, the second electron transfer may occur at the electrode (ECE mechanism) or in bulk solution via disproportionation (DISP):

$$A \pm e^{-} \rightarrow B \qquad [1.11a]$$

$$B \rightarrow C \qquad [1.11b]$$

$$B + C \rightarrow A + Z \qquad [1.11c]$$

Further within the disproportionation mechanism, DISP 1 or DISP 2 may occur if reaction 1.11b or 1.11c are rate determining respectively – this has a direct impact on the voltammetry observed, and also on the respective simulation kinetics.

Other examples of customised notation are also in use, especially for differentiating between reversible and irreversible chemical steps ($C_{rev}$ and $C_{irr}$).

### 1.3.2 Heterogeneous Kinetics

Any process that occurs at a surface is deemed a heterogeneous reaction – either the in the electrode case this might be the electron transfer between a species and the electrode, the deposition of a species onto the electrode, adsorption of a species onto the electrode surface, the stripping of a species from the electrode, or surface-bound reactions. When discussing the electron transfer, the redox couple will have an associated rate of electron transfer dependant on the potential applied, the formal potential of the redox couple $E_f^0$, and the standard rate constant, $k_0$.

A fully reversible reaction occurs when $k_0$ tends towards an infinite rate. The opposite limiting behaviour occurs when $k_0$ is small and as such the reverse reaction is impossible and the reaction is said to be (electrochemically) irreversible. Values of $k_0$ between these two give an electron transfer 'quasi-reversible' status.

At the formal potential, $E_f^0$, a reversible reaction with a pseudo-infinitely fast rate constant result in a 1:1 ratio between products and reactants at the surface (providing both species have equal diffusion coefficients). The formal potential is temperature, pressure and local concentration dependant, but enables analysis of meaningful voltammetric measurements.

A brief overview on electrode kinetics is discussed in the next section, however more extensive and historical detail can be found in other sources [57].

#### 1.3.2.1 Nernst Kinetics

The simplest concept for electrochemical simulation is the application of Nernst kinetics. This considers the ratio of the species in a fully reversible redox couple ($A \pm e^- \rightarrow B$), where the rate constant tends towards a pseudo- infinite value (assuming a one electron transfer):

$$\frac{[A]}{[B]} = \exp\left[\frac{F}{RT}\left(E - E_f^0\right)\right] \qquad [1.12]$$

where $E$ is the applied potential and the electrode surface, and all other symbols are previously defined and found in the Symbols and Constants section at the beginning of this thesis. In simulation, the application of Nernst Kinetics (as described in later paragraphs) gives the optimum simulation throughput, but is often limited to a few select

cases for experiments where both the reactant and product are of a similar nature and no rearrangement of the product occurs after electron transfer.

#### 1.3.2.2 Butler-Volmer Kinetics

The Nernst category of reactions requires a pseudo-infinitely fast rate of electron transfer and thermodynamically favourable conditions. Butler [58] and Volmer [59] derived equations to show the relationship of the realised rate constant in terms of forward and backward rate behaviour in terms of the transfer coefficient, $\alpha$:

Redox Equation: $$A + e^- \underset{k_b}{\overset{k_f}{\rightleftharpoons}} B \quad [1.13]$$

Forward Rate Constant: $$k_f = k_0 \mathrm{e}^{-\alpha\theta} \quad [1.14]$$

Backwards Rate Constant: $$k_b = k_0 \mathrm{e}^{(1-\alpha)\theta} \quad [1.15]$$

$$\theta = \frac{F}{RT}\left(E - E_f^0\right) \quad [1.16]$$

The fully reversible Nernst conditions apply when $k_0$ (and thus $k_f$ and $k_b$) tend towards a pseudo-infinite value – the fully irreversible case is considered when $k_0$ tends to zero. All other values of $k_0$ are considered to give quasi-reversible voltammetry. The rate constant at a given potential is exponentially dependant on both the potential applied and the transfer coefficient, $\alpha$, where $0 \leq \alpha \leq 1$, which describes the transition state in terms of its likeness to either A or B [58,59]. This is shown in equation 1.17 for a one electron reduction [57] as in equation 1.13, where $i$ is the current, $A$ is the surface area of the electrode, and $[A]_0/[B]_0$ are the electrode surface concentrations of species A and species B respectively.

$$i = FAk^0\left\{[B]_0 exp\left[\frac{(1-\alpha)F}{RT}\left(E - E_f^0\right)\right] - [A]_0 exp\left[\frac{-\alpha F}{RT}\left(E - E_f^0\right)\right]\right\} \quad [1.17]$$

Using Butler-Volmer kinetic theory, under extremely favourable conditions (high overpotential, either reductive or oxidative), regardless of the rate constant, full conversion from one species to the other can occur. This situation is suitably applicable in most simulation cases, but not all. As a result, the integration of Marcus-Hush kinetic theory is currently being adapted into simulation [60-62].

#### 1.3.2.3 Marcus-Hush Kinetics

Marcus-Hush kinetics take into account the Gibbs energy of activation and reorganisation energy such that the transition state between A + B (and the transfer coefficient, $\alpha$) can be altered by the potential applied depending on the formal rate constant of the electron transfer, leading to incomplete electron transfer even at high over-potential [57,63-67]. This is shown in equation 1.18, where $\Delta G_i^{\ddagger}$ is the activation energy from the distortion of the inner coordination shell, $\Delta G_0^{\ddagger}$ is the rearrangement energy from solvent dipoles between the initial state and the transition state, and $\lambda$ is the reorganisation energy. Equation 1.19 correlates the transfer coefficient of Butler-Volmer kinetics to the Gibbs energy and reorganisation energy terms, and thus becomes potential dependant.

$$\Delta G^{\ddagger} = \Delta G_i^{\ddagger} + \Delta G_0^{\ddagger} = \frac{\lambda}{4}\left(1 + \frac{\Delta G}{\lambda}\right)^2 \qquad [1.18]$$

$$\alpha = \frac{1}{2}\left(1 + \frac{\Delta G}{\lambda}\right) \qquad [1.19]$$

### 1.3.3 Homogeneous Kinetics

Homogeneous (chemical) reactions within an electrochemical system are described by the corresponding rate equation in terms of each species involved. From a simulation perspective, any rate determining chemical step which follows a homogeneous electron transfer requires special consideration in terms of simulation space and accuracy due to the large diffusional distances which the non-reacted species can cover before eventually performing a homogeneous transformation.

### 1.3.4 Typical Electrochemical Mechanisms

**Table 1.1 – Typical Electrochemical Mechanisms**

| Notation | Observed Steps | Homogeneous Rate Laws |
|---|---|---|
| E | $A + e \rightarrow B$ | |
| CE | $A \rightleftarrows B$<br>$B + e \rightarrow C$ | $\frac{\partial[A]}{dt} = -k_f[A] + k_b[B]$<br>$\frac{\partial[B]}{dt} = +k_f[A] - k_b[B]$ |
| EC | $A + e \rightarrow B$<br>$B \overset{k}{\rightarrow} C$ | $\frac{\partial[B]}{dt} = -k[B]$ $\quad \frac{\partial[C]}{dt} = +k[B]$ |
| $EC_2$ | $A + e \rightarrow B$<br>$2B \overset{k}{\rightarrow} C$ | $\frac{\partial[B]}{dt} = -2k[B]^2$ $\quad \frac{\partial[C]}{dt} = +k[B]^2$ |
| EC' | $A + e \rightarrow B$<br>$B + S \overset{k}{\rightarrow} A$ | $\frac{\partial[A]}{dt} = +k[B][S]$ $\quad \frac{\partial[B]}{dt} = -k[B][S]$ |
| ECE | $A + e \rightarrow B$<br>$B \overset{k}{\rightarrow} C$<br>$C + e \rightarrow D$ | $\frac{\partial[B]}{dt} = -k[B]$ $\quad \frac{\partial[C]}{dt} = +k[B]$ |
| DISP1 | $A + e \rightarrow B$<br>$B \overset{k}{\rightarrow} C$<br>$B + C \rightarrow A + D$ | $\frac{\partial[A]}{dt} = +k[B]$ $\quad \frac{\partial[B]}{dt} = -k[B]$ |
| DISP2 | $A + e \rightarrow B$<br>$B \rightarrow C$<br>$B + C \overset{k}{\rightarrow} A + D$ | $\frac{\partial[A]}{dt} = +k[B]^2$ $\quad \frac{\partial[C]}{dt} = -2k[B]^2$ |

### 1.4 Simulation Concepts – Electrode Geometry

Mass transport to and from the electrode surface is often primarily governed by the electrode geometry itself – equations 1.1-1.4 show that the geometry has a direct effect on the diffusional mode of mass transport, as well as migration. For the most part, electrodes consist of a conducting metal or glassy carbon surface of a shape embedded against an insulating wall (which is resistant to chemical interactions and electron transfer for the experiment being studied). These 'shapes' can fall into a variety of categories – flush against the insulator surface, raised or recessed from the insulator, a single electrode as part of a regular or irregular array, or a three-dimensional shape. These are all designed in an attempt to provide improved mechanistic information regarding the redox couple, or enhanced analytical sensitivity.

#### 1.4.1 Macroelectrodes

Any electrode which is of a size bigger than a millimetre is considered a macroelectrode. Often, these electrodes are either flat against the insulator, or a smooth regular three-dimensional shape (hemisphere), and as such are named 'planar' electrodes, which indicate that all diffusion to the electrode surface is linear.

A flat macroelectrode with linear diffusion affords a one-dimensional Laplacian operator for the description of the electrode, where $z$ is the axis perpendicular to the electrode surface:

$$\nabla^2 = \frac{\partial^2[\mathrm{C}]}{\partial z^2} \qquad [1.20]$$

Hemispherical electrodes are produced by placing a drop of mercury over a smaller electrode, or growth via electrodeposition on the micro and nano scales. These also exhibit 'one-dimensional' diffusion perpendicular to the electrode surface:

$$\nabla^2 = \frac{\partial^2[\mathrm{C}]}{\partial \mathrm{r}^2} + \frac{2}{\mathrm{r}}\frac{\partial[\mathrm{C}]}{\partial \mathrm{r}} \qquad [1.21]$$

### 1.4.2 Microelectrodes

Microelectrodes are electrodes with at least one dimension of the order of microns (<20$\mu$m), and have been used in voltammetric analysis since the 1980s. The benefit over planar macroelectrodes allows the exploration of significantly faster kinetic processes than was previously thought possible. Microelectrodes exhibit this increase in sensitivity due to the convergent diffusion regime: at short periods of time, diffusion is effectively planar, and a Cottrellian $\left(1/\sqrt{t}\right)$ response is observed; at longer times, the decay of current is less rapid, such that the diffusion layer has spread beyond the edge of the electrode, and a steady state current is observed, consistent with convergent diffusion. The magnitude of the flux observed at the electrode surface under steady-state conditions is proportional to the size and shape of the microelectrode - thus, the smaller the microelectrode, the larger the current density and the faster material can diffuse to (and from) the electrode surface.

The most typical microelectrode in use is the microdisk, which can be made by sealing a cylindrical wire in glass or plastic, and polishing the end – or by purchasing commercially. Other microelectrodes in common use are the microband electrode (width of microns, length of mm) and the micromesh electrode (an array of microbands), but unusual microelectrode geometries have also been studied, such as the micro-cone[68] and

micro-cylinder [69]. Later in this thesis, we will discuss the situation of micro-elliptical electrodes[70], as well as micro-rectangular[71] and dual microdisk situations[72].

Due to the convergent nature of the diffusion to a microdisk, the flux at the electrode tends towards an asymptotic 'limiting' constant value, $i_{lim}$, depending on the radius of the disk[73], where $n$ is the number of electrons transferred and $r$ is the radius of the microdisk:

$$i_{lim} = 4nFD[C]_{bulk}r \qquad [1.22]$$

Due to the simulation of such microelectrodes often being two-dimensional, the Laplacian operator is similarly two-dimensional. In the case of the microdisk:

$$\nabla^2 = \frac{\partial^2[\mathrm{C}]}{\partial \mathrm{z}^2} + \frac{\partial^2[\mathrm{C}]}{\partial \mathrm{r}^2} + \frac{1}{\mathrm{r}}\frac{\partial[\mathrm{C}]}{\partial \mathrm{r}} \qquad [1.23]$$

And for the microband:

$$\nabla^2 = \frac{\partial^2[\mathrm{C}]}{\partial \mathrm{x}^2} + \frac{\partial^2[\mathrm{C}]}{\partial \mathrm{y}^2} \qquad [1.24]$$

A thorough explanation of these electrode systems can be found in ref. [67].

### 1.4.3 Arrays

An array of electrodes can occur by design, or by a reduction of the electrode area due to material blocking the electrode surface. By design, a series of microelectrodes can be manufactured in close proximity on the electrode surface, or by blocking, where material involved in electron transfer becomes stuck to the electrode and becomes unreactive. Given sufficient time, the diffusion from these electrodes will interact with each other

due to their close proximity. This has the knock-on effect of influencing the mass transport at the electrode the surface, and hence the current response.

The diffusion regime of an array held at constant potential sufficient to drive a simple redox couple will go through four distinct stages [67,74-76], with a special regime for nanoelectrodes.

- Category 1: At extremely short times, Cottrellian/linear diffusion to each electrode will occur (Figure 1.5) as each electrode acts independently.

Figure 1.5 – A microelectrode array at short time, under Cottrellian diffusion.

- Category 2: Still at short times, individual convergent diffusion occurs, but all the electrodes still act independently (Figure 1.6).

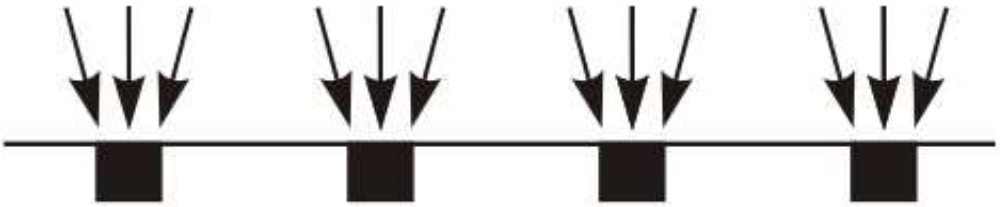


Figure 1.6 – Convergent at individual electrodes becomes sufficient to significantly influence flux at the electrode surface.

- Category 3: Overlapping diffusion results in slower mass transport to the electrode, resulting in a different concentration flux at the electrode surface compared to a single microelectrode.

Figure 1.7 – Diffusion fields from neighbouring electrodes begin to overlap, reducing mass transport to the electrode surface.

- Category 4: At long times, substantial overlap occurs until the array acts like a planar electrode of equivalent size.

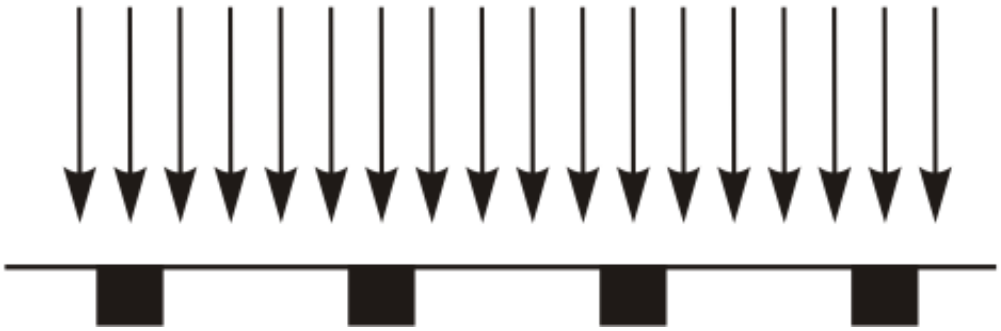


Figure 1.8 – Microelectrode array under effective planar diffusion at long times.

- Category 5: For arrays of nanoelectrodes, if the overall size of the array is on the micron scale, the array at long times will act like a microdisk[77].

In order to visualise these categories in terms of real world values, Menshykau et al [78] have studied the concept diffusional independence and microelectrode arrays, and produced a helpful graph in terms of electrode flux against time, recreated in figure 1.9. This figure shows the flux of a macrodisk, a microdisk, and the comparative flux of a single electrode in a microelectrode array, as well as in which areas each of the four diffusional categories occurs.

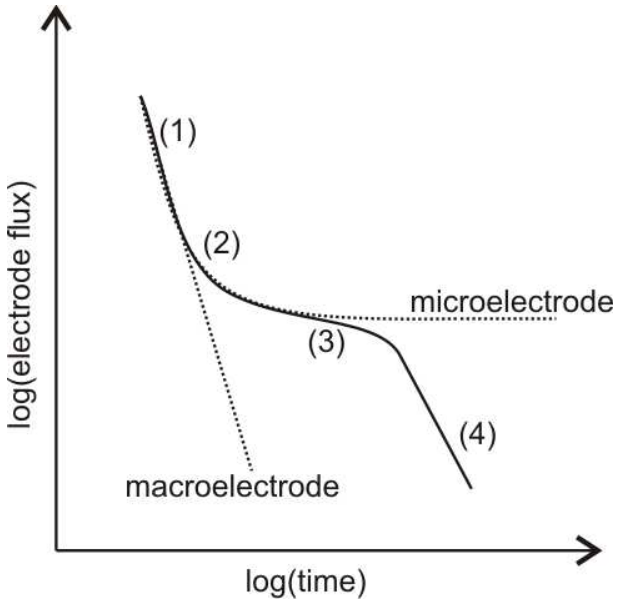



Figure 1.9 - The plot of electrode flux against time for a microelectrode array per electrode, showing at which times each diffusional category is represented, including the expected results from a macroelectrode and microelectrode. Modified from [78].

The simplest array to consider is a regular microdisk array, where an infinite number of microdisks are in a grid. Rather than simulate every microdisk, each microdisk is considered a unit cell of the whole array and only one is required to be simulated. In small electrode arrays, this inaccurately reports the electrodes at the edge of the array, and doubly so for electrodes at the corners.

As simulation of microdisks is typically done in two dimensions, the diffusion domain approximation (DDA) is used in the case of regular arrays. This technique converts the square unit cell to a circle of equal area, and thus uses the assumption that adjacent electrode diffusion layers will act in all directions similarly. This is represented in Figure 1.10.

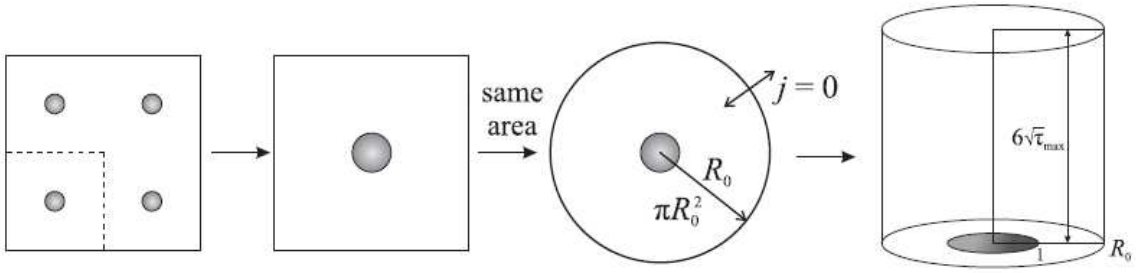



Figure 1.10 – Diffusion domain approximation, transforming the unit cell of an array into a simulation space.

For more complex arrays which are not regular, such as a dispersion of nanodisks on a surface as in figure 1.11, the array is split into theoretical 'Voronoi' cells by walls equidistant between the adjacent electrodes. Each of these domains could then subsequently have the diffusion domain approximation applied and be simulated. The final voltammetric response is then the sum of the responses of these cells, weighted by the distribution of the cell size [77,79,80].

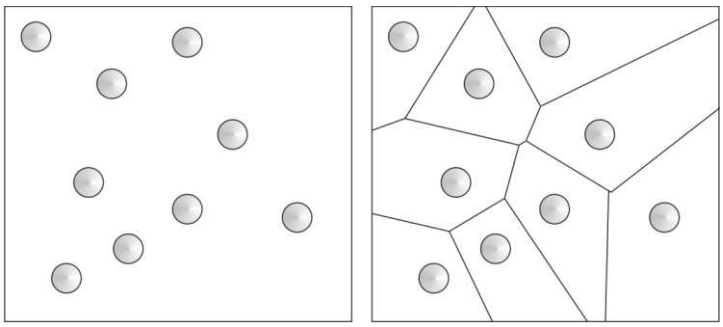


Figure 1.11 – Random array of nanodisks split into Voronoi cells.

### 1.5 Simulation Concepts – Potential as a Function of Time

Experimentally, how the applied potential varies as a function of time will afford different kinetic data. This technique will control how the mass transport at the electrode surface occurs, and subsequently how connected rate laws will function to form the overall result. Common potential functions focus on either the steady-state or the transient:

- At the steady-state regime, the system is in a constant rate of change of net flux, and thus time-dependant influences in the data are removed. Steady-state is achieved by stepping to a fixed potential and waiting for the system to reach a state of no change of the rate of net flux before taking a reading (chronoamperometry), giving kinetic data of the original species in solution. To study the species generated in this system, another potential step is applied in the opposite direction (passing through the formal potential) to shift the electron transfer back to generage the original species (double potential-step chronoamperometry).
- A transient regime will change the potential significantly over time, influencing electron transfer kinetics as a function of time. 'Linear Sweep Voltammetry' (LSV) starts at one potential and sweeps to another at a fixed scan rate (or 'potential ramp) in one direction only. 'Cyclic Voltammetry' is the same technique but the potential is also scanned in the opposite direction, back to the original potential (see figure 1.12). Typically a transient regime observes a peak in the voltammetry, corresponding to the transport-limited current, where electrochemical kinetics are 'infinitely fast' (c.f. section 1.3.2 on heterogeneous kinetics) and the current response is limited only by mass transport.

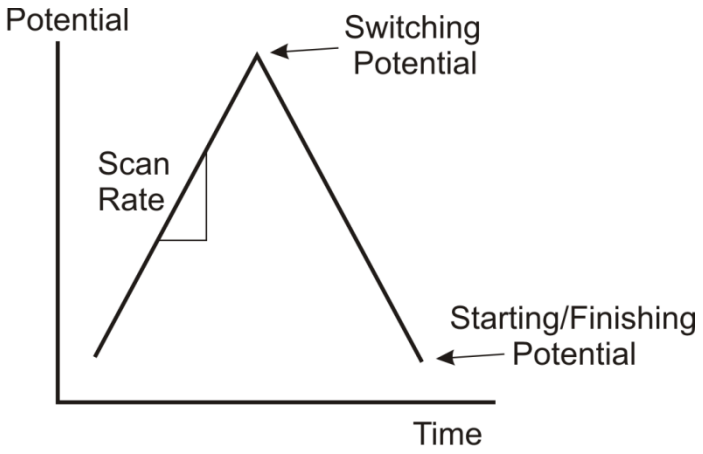



Figure 1.12 – Potential profile of a cyclic voltammogram

Other transient schemas are in common practice (square-wave voltammetry [81-84], pulse-wave voltammetry [85-89]), but are not within the scope of this thesis.

**Bibliography**

[1] S.W. Feldberg, C. Auerbach, Anal. Chem. 36 (1964) 505-&.
[2] S.W. Feldberg, L. Jeftic, J. Phys. Chem. 76 (1972) 2439-&.
[3] B.A. Coles, R.G. Compton, J. Electroanal. Chem. 127 (1981) 37-42.
[4] S.A. Lerke, D.H. Evans, S.W. Feldberg, J. Electroanal. Chem. 296 (1990) 299-315.
[5] I.J. Cutress, R.G. Compton, Electroanalysis 21 (2009) 2617-2625.
[6] I.J. Cutress, E.J.F. Dickinson, R.G. Compton, J. Electroanal. Chem. 655 (2011) 1-8.
[7] I.J. Cutress, N.V. Rees, R.G. Compton, Chem. Phys. Lett. (2011) submitted.
[8] R.G. Compton, Q. Li, C. Batchelor-McAuley, N.S. Lawrence, R.S. Hartshorne, Chemphyschem 12 (2011) 1255-1257.
[9] E.J.F. Dickinson, R.G. Compton, J. Phys. Chem. C 113 (2009) 17585-17589.
[10] D.J. Gavaghan, J. Electroanal. Chem. 456 (1998) 1-12.
[11] D.J. Gavaghan, J. Electroanal. Chem. 456 (1998) 13-23.
[12] D.J. Gavaghan, J. Electroanal. Chem. 456 (1998) 25-35.
[13] B.A. Brookes, D.J. Gavaghan, R.G. Compton, J. Phys. Chem. B 106 (2002) 4886-4896.
[14] O.V. Klymenko, D.J. Gavaghan, K.E. Harriman, R.G. Compton, J. Electroanal. Chem. 531 (2002) 25-31.
[15] N.P.C. Stevens, K.A. Gooch, A.C. Fisher, J. Phys. Chem. B 104 (2000) 1241-1248.
[16] M.E. Hyde, O.V. Klymenko, R.G. Compton, J. Electroanal. Chem. 534 (2002) 13-17.
[17] J.A. Alden, R.G. Compton, J. Phys. Chem. B 101 (1997) 8941-8954.
[18] J.G. Limon-Petersen, I. Streeter, N.V. Rees, R.G. Compton, J. Phys. Chem. C 112 (2008) 17175-17182.
[19] J.G. Limon-Petersen, J.T. Han, N.V. Rees, E.J.F. Dickinson, I. Streeter, R.G. Compton, J. Phys. Chem. C 114 (2010) 2227-2236.
[20] E.J.F. Dickinson, J.G. Limon-Petersen, N.V. Rees, R.G. Compton, J. Phys. Chem. C 113 (2009) 11157-11171.

[21] O.V. Klymenko, C. Amatore, I. Svir, Anal. Chem. 79 (2007) 6341-6347.
[22] A. Fick, Poggondorff's Annel. Physik 94 (1855) 59.
[23] F.G. Cottrell, Z. Phys. Chem. 44 (1908) 385.
[24] J. Ma, L. Yu, Z.W. Fu, Electrochim. Acta 51 (2006) 4802-4814.
[25] R.G. Compton, A.C. Fisher, R.G. Wellington, P.J. Dobson, P.A. Leigh, J. Phys. Chem. 97 (1993) 10410-10415.
[26] R.G. Compton, A.C. Fisher, G.H.W. Sanders, Electroanalysis 5 (1993) 615-617.
[27] N.P.C. Stevens, A.C. Fisher, J. Phys. Chem. B 101 (1997) 8259-8263.
[28] C. Amatore, A. Oleinick, O.V. Klymenko, I. Svir, Chemphyschem 6 (2005) 1581-1589.
[29] C. Amatore, O.V. Klymenko, I. Svir, Chemphyschem 7 (2006) 482-487.
[30] L. Rassaei, E. Vigil, R.W. French, M.F. Mahon, R.G. Compton, F. Marken, Electrochim. Acta 54 (2009) 6680-6685.
[31] L. Rassaei, R.W. French, R.G. Compton, F. Marken, Analyst 134 (2009) 887-892.
[32] I.J. Cutress, F. Marken, R.G. Compton, Electroanalysis 21 (2009) 113-123.
[33] R.G. Compton, M.E. Laing, D. Mason, R.J. Northing, P.R. Unwin, Proceedings of the Royal Society of London Series a-Mathematical Physical and Engineering Sciences 418 (1988) 113-154.
[34] R.G. Compton, R.G. Harland, Journal of the Chemical Society-Faraday Transactions I 85 (1989) 761-771.
[35] R.G. Compton, P.R. Unwin, Journal of the Chemical Society-Faraday Transactions I 85 (1989) 1821-1834.
[36] C.E. Banks, N.V. Rees, R.G. Compton, J. Electroanal. Chem. 535 (2002) 41-47.
[37] A.D. Clegg, N.V. Rees, C.E. Banks, R.G. Compton, Chemphyschem 7 (2006) 807-811.
[38] S.W. Feldberg, J. Electroanal. Chem. 109 (1980) 69-82.
[39] S.W. Feldberg, C.I. Goldstein, M. Rudolph, J. Electroanal. Chem. 413 (1996) 25-36.
[40] A.B. Miles, R.G. Compton, J. Electroanal. Chem. 487 (2000) 75-89.
[41] K.A. Gooch, A.C. Fisher, J. Phys. Chem. B 106 (2002) 10668-10673.
[42] E.C. Spencer, K.A. Gooch, A.C. Fisher, Electrochem. Commun. 4 (2002) 358-362.
[43] I.E. Henley, A.C. Fisher, J. Phys. Chem. B 107 (2003) 6579-6585.
[44] A.C. Fisher, S.M. Matthews, G.Q. Du, J. Solid State Electrochem. 10 (2006) 817-825.
[45] P. Mandin, C. Fabian, D. Lincot, J. Electrochem. Soc. 153 (2006) D40-D50.
[46] O.V. Klymenko, A.I. Oleinick, C. Amatore, I. Svir, Electrochim. Acta 53 (2007) 1100-1106.
[47] V.M. Volgin, A.D. Davydov, J. Electroanal. Chem. 600 (2007) 171-179.
[48] C. Amatore, N. Da Mota, C. Sella, L. Thouin, Anal. Chem. 82 (2010) 2434-2440.
[49] W.G. Huang, D.B. Hibbert, Physica A 233 (1996) 888-896.
[50] Y.K. Kwok, C.C.K. Wu, Computer Methods in Applied Mechanics and Engineering 132 (1996) 305-317.
[51] A.A. Moya, J. Horno, Electrochim. Acta 41 (1996) 285-290.
[52] J.C. Myland, K.B. Oldham, J. Electroanal. Chem. 529 (2002) 66-74.
[53] V.M. Volgin, O. Volgina, A.D. Davydov, Computational Biology and Chemistry 27 (2003) 185-196.
[54] I. Streeter, R.G. Compton, J. Phys. Chem. C 112 (2008) 13716-13728.
[55] R.G. Compton, J.G. Limon-Petersen, E.J.F. Dickinson, S.R. Belding, N.V. Rees, J. Electroanal. Chem. 650 (2010) 135-142.
[56] A.C. Testa, W.H. Reinmuth, Anal. Chem. 32 (1960) 1512-1514.
[57] A.J. Bard, L.R. Faulkner: Electrochemical Methods: Fundamentals and Applications, Wiley, Phoenix, 2001.
[58] J.A.V. Butler, Transactions of the Faraday Society 28 (1932) 379-382.
[59] T. Erdey-Grúz, M. Volmer, Z. Phys. Chem. 150 (1930) 203-213.
[60] K.B. Oldham, J.C. Myland, J. Electroanal. Chem. (2011) in press, doi: 10.1016/j.jelechem.2011.1001.1044.

[61] I.J. Cutress, R.G. Compton, Chem. Phys. Lett. 508 (2011) 306-313.
[62] S.W. Feldberg, Anal. Chem. 82 (2010) 5176-5183.
[63] R.A. Marcus, J. Chem. Phys. 24 (1956) 966-978.
[64] R.A. Marcus, J. Phys. Chem. 67 (1963) 853.
[65] R.A. Marcus, Annu. Rev. Phys. Chem. 15 (1964) 155.
[66] N.S. Hush, Transactions of the Faraday Society 57 (1961) 557.
[67] C.E. Banks, R.G. Compton: Understanding Voltammetry, World Scientific, Singapore, 2011.
[68] E.J.F. Dickinson, I. Streeter, R.G. Compton, J. Phys. Chem. B 112 (2008) 4059-4066.
[69] E.J.F. Dickinson, I. Streeter, R.G. Compton, J. Phys. Chem. C 112 (2008) 11637-11644.
[70] I.J. Cutress, R.G. Compton, J. Electroanal. Chem. 643 (2010) 102-109.
[71] I.J. Cutress, R.G. Compton, J. Electroanal. Chem. 645 (2010) 159-166.
[72] I.J. Cutress, Y.J. Wang, J.G. Limon-Petersen, S.E.C. Dale, L. Rassaei, F. Marken, R.G. Compton, J. Electroanal. Chem. 655 (2011) 147-153.
[73] Y. Saito, Review of Polarography 15 (1968) 177.
[74] S.R. Belding, R.G. Compton, J. Phys. Chem. C 114 (2010) 8309-8319.
[75] T.J. Davies, C.E. Banks, R.G. Compton, J. Solid State Electrochem. 9 (2005) 797-808.
[76] K.R. Ward, N.S. Lawrence, R.S. Hartshorne, R.G. Compton, J. Phys. Chem. C 115 (2011) 11204-11215.
[77] N. Godino, X. Borrise, F.X. Munoz, F.J. del Campo, R.G. Compton, J. Phys. Chem. C 113 (2009) 11119-11125.
[78] D. Menshykau, X.J. Huang, N.V. Rees, F.J. del Campo, F.X. Munoz, R.G. Compton, Analyst 134 (2009) 343-348.
[79] H. Reller, E. Kirowaeisner, E. Gileadi, J. Electroanal. Chem. 138 (1982) 65-77.
[80] C. Amatore, J.M. Savéant, D. Tessier, J. Electroanal. Chem. 147 (1983) 39-51.
[81] D.P. Whelan, J.J. Odea, J. Osteryoung, K. Aoki, J. Electroanal. Chem. 202 (1986) 23-36.
[82] K. Aoki, K. Maeda, K. Tokuda, H. Matsuda, K. Hasebe, Bunseki Kagaku 36 (1987) 420-424.
[83] M. Powell, J.C. Ball, Y.C. Tsai, M.F. Suarez, R.G. Compton, J. Phys. Chem. B 104 (2000) 8268-8278.
[84] A.B. Miles, R.G. Compton, J. Electroanal. Chem. 499 (2001) 1-16.
[85] K. Aoki, K. Tokuda, H. Matsuda, J. Electroanal. Chem. 175 (1984) 1-13.
[86] K. Aoki, K. Tokuda, H. Matsuda, J. Electroanal. Chem. 206 (1986) 47-56.
[87] M. Penczek, Z. Stojek, J. Electroanal. Chem. 213 (1986) 177-188.
[88] J.L. Melville, R.G. Compton, J. Electroanal. Chem. 501 (2001) 114-127.
[89] J.L. Melville, R.G. Compton, Electroanalysis 13 (2001) 123-130.

## Chapter 2: Why Develop Algorithms – Isn't Software Already Available?

An emerging trend in the literature is the use of off-the-shelf software for computer simulations of electrochemical systems. Software for the electrochemist can vary from those specifically made for electrochemistry (e.g. DIGISIM, explained below) which tackle simple systems, to larger commercial packages designed for many areas of physics and engineering. These larger packages often advertise the ability to simulate almost anything due to their algorithmic processing capabilities (particularly partial differential equations) in three spatial dimensions, either transient or steady state. As part of this thesis, we have examined one such product which is picking up traction in the electrochemical literature [1-7], COMSOL, and issued a warning to those wishing to use such software for future analyses.

### 2.1 The Importance of Electrochemical Simulation

Electrochemistry is a science where the interplay of theory and experiment is both highly significant and active - specifically, the effects of diffusion, migration and convection on an electrolytic process may be profound yet also subtle, so that the experimentalist needs to be guided by qualitative modelling of his or her problems. The early development of voltammetry was constrained by the need to work under limiting conditions, for example with large excesses of electrolyte to suppress migration, and experiments were often confined to chemically relatively simple studies which approximated to media which were amenable to analytical treatments in the mathematical sense. At the same time, most experiments were confined to electrodes in which planar diffusion avoided the need to solve transport equations in two or three dimensions.

The gradual evolution of computational electrochemistry possibly started with the work of S. Feldberg in the 1960s [8] has steadily opened up the ability to treat complex problems of academic interest. Such studies include, but are not limited to, modified electrodes [9,10], rough surfaces [11], weakly-supported electrolysis [12-14], heated and insonated electrodes [15-18], electrodes of various geometry [19], hydrodynamic electrodes [20,21], coupled homogeneous and adsorption kinetics [22,23], and electrode arrays [24]. A major landmark in the area was the publication of the authoritative text by Britz entitled 'Digital Simulation in Electrochemistry' in 1981 [25]. At that stage, as before, those seeking to model experiments were required to write the computer code needed for their simulations; the number of groups able to accomplish this was finite and much electrochemistry was done under experimental conditions for systems which were well defined and largely devoid of all but obvious physics. A significant step in the direction of 'simulations for all' was the development of the program DIGISIM™ pioneered by the work of Rudolph [26]. This enabled one-dimensional voltammetric problems to be solved with the major advantage of an almost unlimited number of participating chemical species. This for the first time enabled cyclic voltammetry to be analysed in ways not limited by the existence of analytical theories for various mechanistic types (EC, ECE, CE, DISP, etc.) but rather only constrained by the chemical insight of the investigator, although the scope for including unlimited species did nothing to constrain chemical imagination in some mechanistic analyses. DIGISIM was later enhanced by the addition of capabilities for the inclusion of convection and the extension to microelectrodes. DIGISIM is excellent for the study of cyclic voltammetry conducted under usual experimental conditions. However, for experimentalists wishing to direct electrochemistry into new areas, such as nanoelectrodes, hybrid approaches such as microwave activated [27,28] electrodes, or even to study

relatively simple (in concept) systems such as adjacent microdisc electrodes [29] or voltammetry in relatively weakly supported media, the scientist must make his or her own simulations.

To this end, the use of finite element simulation packages produced commercially for the general engineering area have commended themselves to some electrochemists. Bearing in mind the essential need for quantitative results for almost all electrochemical studies, this chapter addresses the extent to which, if at all, such packages can provide quantitative information in the very specific area of electrochemistry.

**2.2 Previous Literature**

Current peer-reviewed literature has explored the use of off-the-shelf FEM software, including FEMLAB (the previous incarnation of COMSOL for use in Matlab), to varying degrees of success, accuracy and implementation. By far the most common use is for optimization of physical methods - for example, Mahnke et al. [1] have utilised simulations in 3D to optimise the shape of directly heated electrodes for minimal temperature gradients; however, no direct comparison to experiment was made. Cheng coupled electrochemistry, fluid mechanics, steric stabilisation and heat transfer into a computational fluid dynamics model, again with no direct comparison to experiment, leading to a suggested optimised route of deposition [2]. Guo et al. investigated cyclic voltammetry at microelectrode and recessed ultramicroelectrode arrays [3] – initially a 3D model of a representative array area was compared to a 2D model using the diffusion domain approximation. Both gave similar results, and thus 2D modelling was used throughout the investigation. 2D models resulted in scan rate vs. coverage data for different types of microelectrode array behaviour. Experimental data for recessed electrode systems were

compared to simulation data and resulted in agreement; however, the simulation repeatedly underestimated the reported current (*vide infra*). This current underestimation was also seen by Mazurenka et al. in cavity ring-down response for cyclic voltammetry and potential-step chronoamperometry (using 2D models in FEMLAB) [4]**,** and Schnippering et al. in a thin layer arrangement [5]. Holder et al. probed the potential step response of scanning electrochemical-atomic force microscopy via 3D FEM simulation – results show that the response follows Cottrelian behaviour within acceptable limits [6]. Basha et al. utilised 2D FEMLAB simulations to model various electrode situations, such as moving electrodes, parallel plate electrodes, hull cells, curvilinear hull cells, thin layer galvanic cells, through hold plating, and recessed disk electrodes, to find good agreement with previously published experiments [7]. Kitazumi has published a review on examples of FEM in electrochemistry, concluding that 'a well conditioned model is required, and it is of prime importance to take care during program use' [30].

Based on the reported use of general engineering finite element software, we embarked to find the limit to which electrochemistry is an applicable tool. Modelling and meshing of one, two and three-dimensional systems with standardised analytical solutions were compared to the answers of those solutions, and where applicable, well defined and published finite difference simulation. A comparison with kinetic phenomena, chronoamperometric transients and the rigorous nature of the adaptive time meshing system was also performed.

## 2.3 Results and Analysis

### 2.3.1 Meshing

COMSOL incorporates an automatic FEM meshing technique, useable at the click of a button. For the purposes of electrochemical solutions, this is often inappropriate unless specific bounds are defined and areas of interest selected. For example, Figure 1 shows the meshing in a three-dimensional microdisc problem – Figure 2.1a represents the standard automatic meshing, whereas Figure 2.1b represents user-defined meshing of a higher granularity.

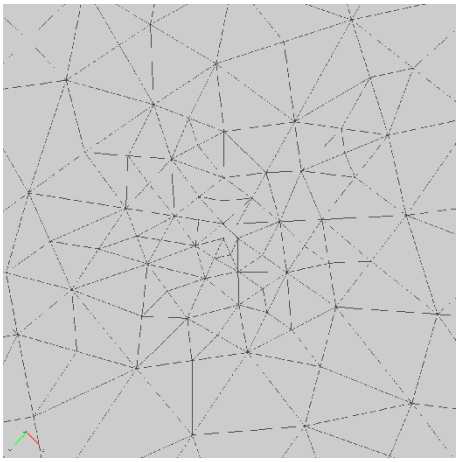


Figure 2.1a – 3D microdisc auto meshing

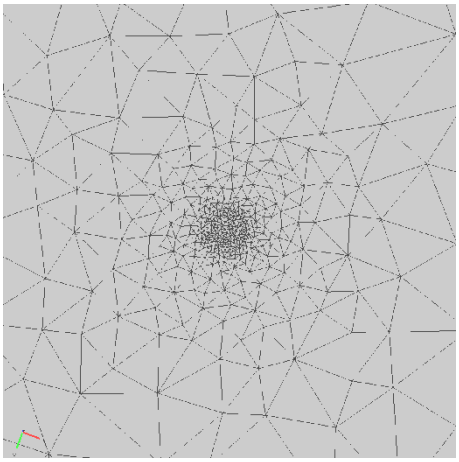


Figure 2.1b – 3D microdisc user-defined meshing with higher granularity

Meshing options include, but are not limited to, maximum size of element within a sub-domain, number of points along an edge boundary, and general gradations of granularity within the simulation. Figure 2.1b represents a 10 micron diameter microdisc electrode with a maximum element size not less than 1 micron, 100 elements around the edge boundary and a setting of 'fine' on the 'preferred mesh size' setting, compared to Figure 1a which has only the default setting of 'normal' on preferred mesh size. A system should

ideally be well meshed in all directions, so care must be taken if quantitative results are required. The limit with meshing, as with finite difference systems, is directly related to the physical memory of the computational system used for simulation. Figure 2 represents a 2D microdisc meshing, whereby a boundary condition of 250 points and a maximum boundary mesh point size of 1 micron is set, as well as a point condition of 1 micron maximum mesh size point.

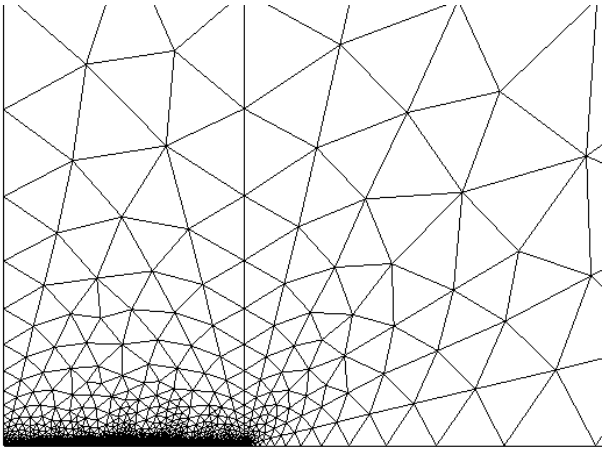


Figure 2.2 – 2D optimised mesh

The granularity of mesh definition was probed with respect to the steady state current, $i$(ss), at infinite cylinders and work done by Dickinson et al. [31] with finite difference methods. Table 2.1 demonstrates the results obtained by varying the number of mesh points along the boundary, the minimum size of a mesh point at the cylinder singularity, and the effect of increasing the distance to the bulk boundary (in multiples of cylinder radius) of the finite element model. Results conclude an interesting number of observations – such as the failure of the software to mesh when the boundary mesh density is set at 500 points, yet successful meshing at both 250 and 1000 points. An increase in boundary mesh points however does give a quantitative result closer to the expected steady state value. With

respect to the minimum mesh size at the singularity, successive decreases in relative error are observed until the minimum mesh size reaches a value where the software fails to mesh – the meshing failure on this instance was related to a lack of system memory on the machine used (Intel Q9550 @ 2.83 GHz with 3.25 GiB RAM under 32-bit Windows XP). As the distance to the bulk boundary is increased (for example, where a slow scan rate is used and thus a larger diffusion layer is required), but keeping the mesh boundary density, the singularity point maximum mesh size and the scan rate constant, the simulation error compared to finite difference increases. When the distance to the bulk boundary rose to 20 cylinder radii and above, a singularity size point of $10^{-9}$ resulted in a failed mesh generation, thus a singularity size point of $10^{-8}$ was used instead, and is the cause of the greater loss of accuracy observed.

**Table 2.1 – Meshing at 'Infinite' Cylinders**

| **1mm*10µm Cylinder** | | $i$(ss, FD) = $5.762 \times 10^{-14}$ A |
|---|---|---|
| **Mesh Density (Boundary)** | ***i*(ss, FEM) / A** | **% Error vs. FD** |
| 1000 | $5.44 \times 10^{-14}$ | 5.58 |
| 500 | N/A | N/A |
| 250 | $5.35 \times 10^{-14}$ | 7.14 |
| **Mesh Size (Point)** | | |
| $10^{-8}$ | $5.52 \times 10^{-14}$ | 4.28 |
| $10^{-9}$ | $5.68 \times 10^{-14}$ | 1.39 |
| **Distance to bulk boundary / cylinder radii** | | |
| 5 | $5.71 \times 10^{-14}$ | 0.97 |
| 10 | $5.68 \times 10^{-14}$ | 1.39 |
| 20 | $5.50 \times 10^{-14}$ | 4.50 |
| 50 | $5.52 \times 10^{-14}$ | 4.24 |

In all of Table 2.1, a preferred mesh size setting of 'extra-fine' was used, with two successive software determined refinements.

### 2.3.2 System Dimensionality

One of the main marketed features of FEM simulation packages is the ability to simulate in three dimensions within a reasonable degree of accuracy. However, finite difference electrochemical simulation often requires a high level of accuracy to obtain representative results where singularities (such as disc edges) are present [32]. Accordingly, one, two and three-dimensional systems were probed for their accuracy relating to analytical solutions.

#### 2.3.2.1 One Dimension

A rotating disc electrode (RDE) system was modelled as a representative one-dimensional system and compared against the Levich equation for an RDE [33]. Table 2.2 and Figure 2.3 show the current divided by area ($i_{lim}/nFA$) results against various rotation speeds ($\omega$ / $Hz$), using a highly refined mesh. A constant error of 0.06% was observed at all rotation speeds, showing good agreement between the finite element software and the analytical solution.

**Table 2.2 – Comparison of RDE FEM simulation and Levich Equation**

| $\sqrt{\omega}/Hz^{1/2}$ | ($i_{lim}$/nFA) / mol m$^2$ s$^{-1}$ | Levich eqn / mol m$^2$ s$^{-1}$ | % Error |
|---|---|---|---|
| 1.00 | 1.555 x10$^{-5}$ | 1.554 x10$^{-5}$ | 0.06 |
| 2.00 | 3.110 x10$^{-5}$ | 3.108 x10$^{-5}$ | 0.06 |
| 3.16 | 4.917 x10$^{-5}$ | 4.914 x10$^{-5}$ | 0.06 |
| 4.47 | 6.954 x10$^{-5}$ | 6.950 x10$^{-5}$ | 0.06 |
| 6.32 | 9.835 x10$^{-5}$ | 9.828 x10$^{-5}$ | 0.06 |

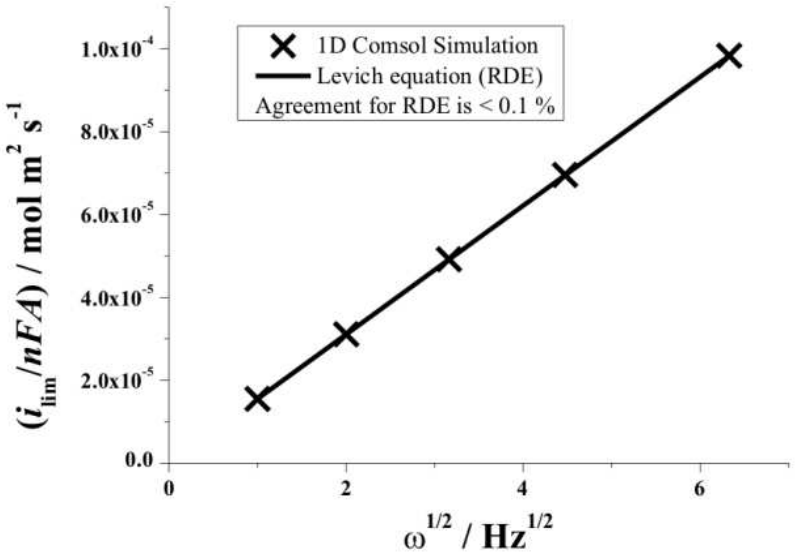



Figure 2.3 – Comparison of RDE FEM simulation and Levich Equation

#### 2.3.2.2 Two Dimensions

Similarly, both the steady state current at a rotating ring electrode (RRE) and channel flow electrode (see Chapter 1.2.2) were simulated as typical two dimensional problems [34-36]. Figure 2.4 and Table 2.3 represent the limiting ring current calculated for RRE simulation and comparison assuming a ring width of 0.0005 m. From the results, the RRE has a large margin of error at slow rotation rates, and the error is not constant.

**Table 2.3 – Comparison of Rotating Ring Eletode FEM and Levich Equation**

| $\sqrt{\omega}/Hz^{1/2}$ | ($i_{lim}$/nF) / mol $s^{-1}$ | Levich eqn / mol $s^{-1}$ | % Error |
|---|---|---|---|
| 1.00 | 4.538 x$10^{-11}$ | 4.466 x$10^{-11}$ | 1.61 |
| 2.00 | 8.957 x$10^{-11}$ | 8.932 x$10^{-11}$ | 0.28 |
| 3.16 | 1.412 x$10^{-10}$ | 1.412 x$10^{-10}$ | -0.06 |
| 4.47 | 1.997 x$10^{-10}$ | 1.997 x$10^{-10}$ | -0.04 |
| 6.32 | 2.826 x$10^{-10}$ | 2.825 x$10^{-10}$ | 0.04 |

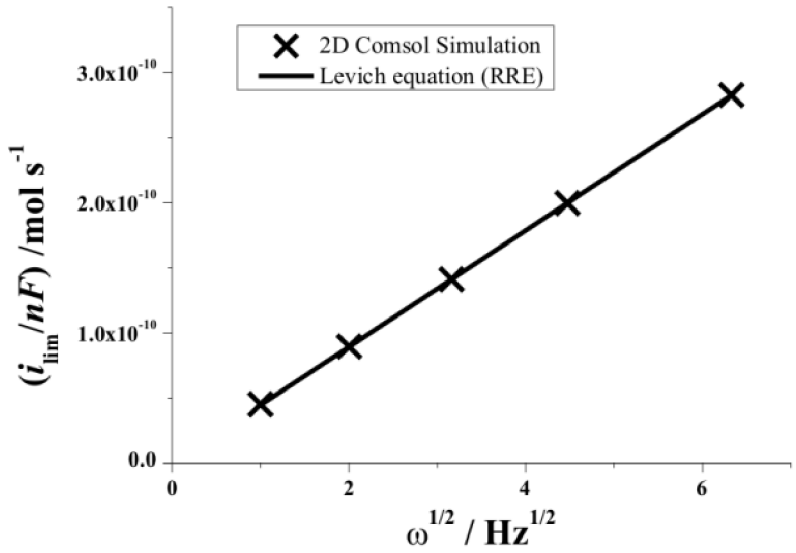



Figure 2.4 – Comparison of RRE FEM and Levich equation

Figure 2.5 and Table 2.4 show the Channel Flow system (for an electrode length of 10 micron and a cell height of 100 micron), and like the RRE system, the error against the Levich equation is not a constant error. It is important to note that the errors can at times be relatively large, such as when the rotation speed, $\omega$= 1 in RRE or the channel flow rate, $V_{0,}$ = 10 m s$^{-1}$ in channel flow systems.

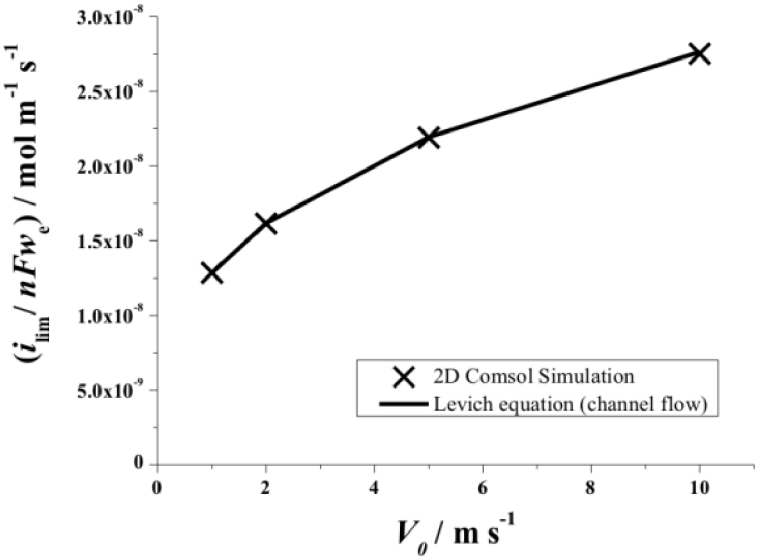



Figure 2.5 – Comparison of Channel Flow FEM and Levich Equation

**Table 2.4 – Comparison of Channel Flow FEM and Levich Equation**

| $V_0$ / m $s^{-1}$ | ($i_{lim}$/nFw) / mol $m^{-1}$ $s^{-1}$ | Levich eqn / mol $m^{-1}$ $s^{-1}$ | % Error |
|---|---|---|---|
| 1 | 1.285 x$10^{-8}$ | 1.283 x$10^{-8}$ | 0.13 |
| 2 | 1.615 x$10^{-8}$ | 1.616 x$10^{-8}$ | -0.08 |
| 5 | 2.191 x$10^{-8}$ | 2.193 x$10^{-8}$ | -0.11 |
| 10 | 2.752 x$10^{-8}$ | 2.764 x$10^{-8}$ | -0.41 |

#### 2.3.2.3 Three Dimensions

Three dimensional systems involving a single microdisc, and a single hemisphere, were analysed against expected steady state values [37]. Figures 2.1a and 2.1b demonstrate the meshing for a 3D microdisc, whereas Figures 2.6a and 2.6b represent meshing at a hemisphere. The standard finite element meshing generated by the software is shown in figure 2.6a, whereas Figure 2.6b shows the optimisation of that mesh as defined by the user, and it is of note that this mesh is entirely user dependant. These two systems were selected as they are well defined systems, often modelled by finite difference methods in one dimension in the case of the hemisphere [14], and in two dimensions for the case of the microdisc [32,38]. In 3D, the hemisphere is represented by a protrusion into the simulation space with a modified boundary condition, whereas the microdisc is a two dimensional ($xy$) boundary condition superimposed on the 3D simulation space.

Table 2.5 shows the results of 3D FEM simulations with the unoptimised and optimised meshing of the steady state current of a 10$\mu$m diameter microdisc, and table 2.6 shows the results of a 5$\mu$m radius hemisphere. The results indicate that the accuracy of the hemisphere can be changed from a 3.69% underestimation in the steady state current to a

1.55% overestimation. This is typically an unacceptable level of error for a finite difference simulation – especially when considering a 1D hemisphere simulation, where the error is expected, and required, to be negligible, and for which at steady state the result is known analytically. However, in the 3D microdisc FEM simulation, the initial error without optimisation is a 63.29% below the expected value, dropping to 18.54% when optimising up to the memory limit. Again, these results are unacceptable for quantitative comparison with experiment.

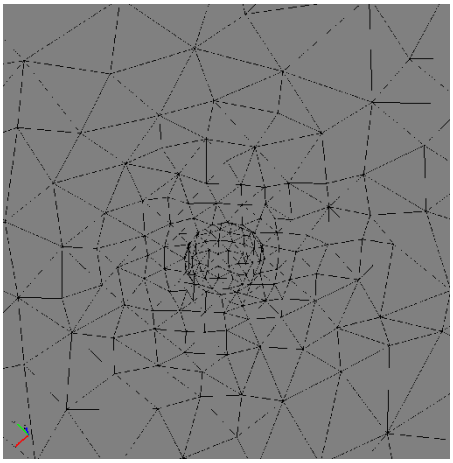


Figure 2.6a – Meshing of a 3D hemisphere using default options

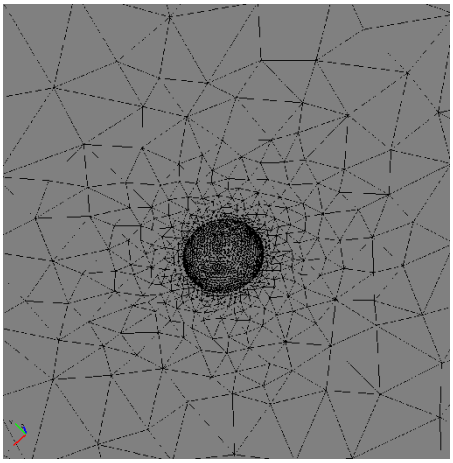


Figure 2.6b – Meshing of a 3D hemisphere under more rigorous meshing conditions.

**Table 2.5 – Comparison of steady state currents for 3D microdisc FEM to analytical solutions**

| | | | $i$(ss) / A |
|---|---|---|---|
| 3D Microdisc | Expected | | $1.06 \times 10^{-5}$ |
| | FEM | Normal mesh | $3.90 \times 10^{-6}$ |
| | | User Defined | $8.65 \times 10^{-6}$ |
| | Error | Normal mesh | 63.29% |
| | | User Defined | 18.54% |

**Table 2.6 – Comparison of steady state currents for 3D hemisphere FEM to analytical solutions**

| | | | $i$(ss) / A |
|---|---|---|---|
| 3D Hemisphere | Expected | | 3.33 x$10^{-5}$ |
| | FEM | Normal mesh | 3.21 x$10^{-5}$ |
| | | User Defined | 3.39 x$10^{-5}$ |
| | Error | Normal mesh | 3.69% |
| | | User Defined | -1.55% |

3D simulations can be used to observe physical phenomena in areas of interest when common assumptions, such as the diffusion domain approximation in the case of small ultramicroelectrode and nanoelectrode arrays, are no longer valid, as shown by the work of Godino et al [39].

#### 2.3.2.4 Accuracy in Three Dimensions and Hardware Limitations

The problems arising from accuracy in 3D simulation in both finite difference and finite element are often due to the limits of available computer memory, and hence limitation of the achievable granularity. A typical, highly accurate 2D finite difference grid of a microdisc will require 300x150 nodes with each requesting a double precision floating point variable - 45000 x 8 bytes per double = 360KB of computer memory; whereas for a grid of 300x300x150 nodes for three dimensional problems (i.e. a single microdisc represented in 3D), this now requires 108MB+ of memory. Double precision variables consumes 8 bytes of memory each at an accuracy of 13 significant figures, with a minimum error in the 16$^{th}$ significant figure due to binary rounding IEEE-754 modes – this is compared to the single precision variables, typically used in scientific computation, which affords faster

computation at the expense of accuracy (6 significant figures, minimum error in the 8$^{th}$ significant figure). Electrochemical simulation is typically performed using double precision to achieve maximum accuracy.

Given that a single lab machine dedicated to simulation may currently contain 1GB – 4GB+ of total memory (for a 64-bit operating system, possibly 16-64 GB+ in the next few years), this gets used up quickly as more and more electrodes are added. Thus a jump to computer clusters, or highly refined memory efficient algorithms, must be made. The way around this is to simplify the system – reduce the number of elements or use a different gridding system (Finite Difference to Finite Element or vice versa). The resultant downside of this method is a loss in accuracy, as shown in Table 2.5 with respect to the microdisc. Off-the-shelf packages are often designed in this single computer context, where memory is an issue for simulation of large complex systems without approximation or reduction in accuracy.

Godino at al., in both 2D and 3D representation of nanoelectrode arrays, initially start with a recessed electrode reaction for which results are known, and modify meshing parameters to within an error to which they deem appropriate, then expand into different reaction mechanisms and compare results against experiment (for example, an experimental result for peak current of 1.7 ± 0.1 nA against a simulation result of 1.76 nA) [39]. For novel systems and mechanisms, for where meshing is one of the most significant factors in quantitative result comparison against experiment, this approach is seemingly the most successful where applicable, and requires significant user interaction over standard settings. Nevertheless, Godino reported that whilst COMSOL simulations gave good results for arrays with small numbers of nanoelectrodes (6x6 array), the software (due to hardware

limitations) did not successfully produce quantitative results for large array sizes (61x61 array).

### 2.3.3 Electrochemical Kinetic Modelling

The capability of the FEM package to simulate 2D Butler-Volmer kinetics at microelectrodes was probed against well defined finite difference simulation [19].

For an E reaction (as described in Table 1.1), such as reaction 2.1 where $A$ and $B$ are species of interest, electron transfer at an electrode is often modelled by Nernstian (equation 2.2) or Butler-Volmer (equation 2.3) kinetics [24,37].

$$A + e \rightleftarrows B \qquad (2.1)$$

$$A_0 = B_0 e^{-\theta} \qquad (2.2)$$

Figures 2.7a and 2.7b show the comparison between FEM and FD BV linear sweep microdisc simulations at $\alpha$ = 0.5 and $\alpha$ = 1.5 for $n$ = 2. In these figures, the dimensionless flux is plotted against dimensionless potential. In both situations, the FEM simulation underestimates the diffusional current.

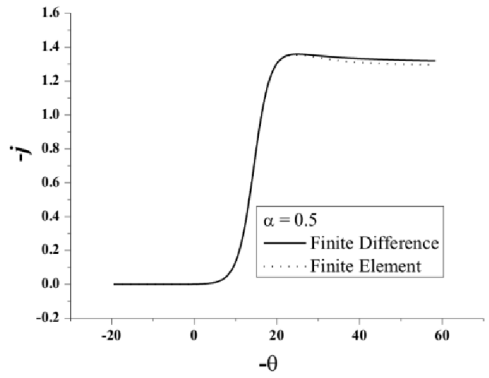



Figure 2.7a – BV Kinetics when $\alpha$ = 0.5

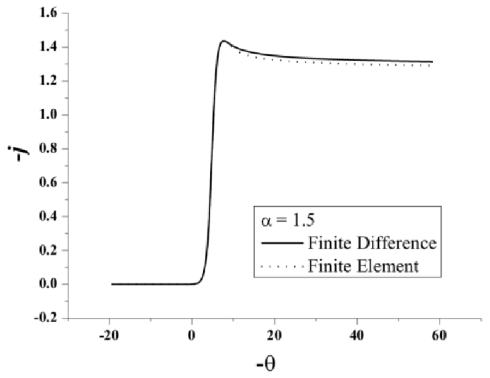



Figure 7b – BV Kinetics when $\alpha$ = 1.5

The EC reaction is described by reaction 2.1 followed by reaction 2.3.

$$B \underset{k_b}{\overset{k_f}{\rightleftarrows}} C \qquad (2.3)$$

For coupled kinetic systems, such as a reversible EC and an irreversible EC mechanism, we also see the FEM package underestimate the current given in from the FD simulations [40], as seen in figures 2.8a and 2.8b.

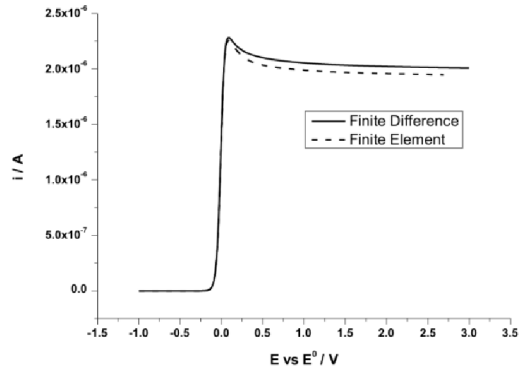



Figure 2.8a – Comparison of FEM vs. FD for an EC reversible reaction

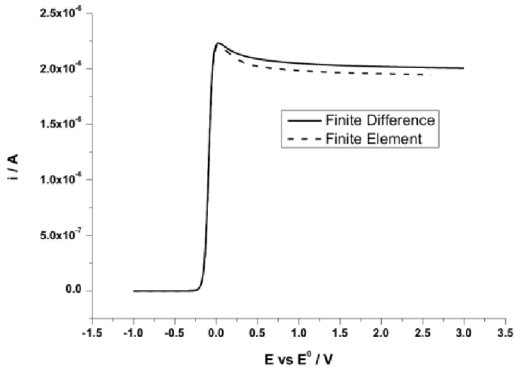



Figure 2.8b – Comparison of FEM vs. FD for an EC irreversible reaction

Figure 2.9 shows $\beta$, the effective transfer coefficient, where $\beta = n - \alpha$ with respect to time, of Figures 2.7a and 2.7b. The calculation of $\beta$ is from the derivative of the Tafel plot of Figures 2.7a and 2.7b with respect to potential, and is used in this circumstance to demonstrate that BV kinetics are correctly applied in FEM simulations, and these are accurate in the region where current is kinetically limited.

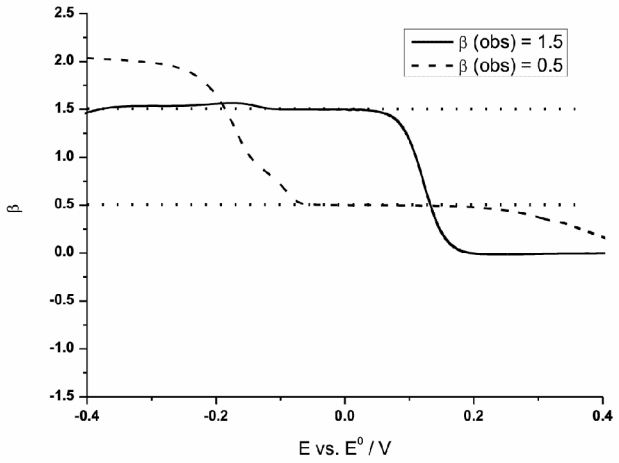



Figure 2.9 – $\beta$ against Potential for Figures 2.8a and 2.8b

### 2.3.4 Chronoamperometry in Two Dimensions

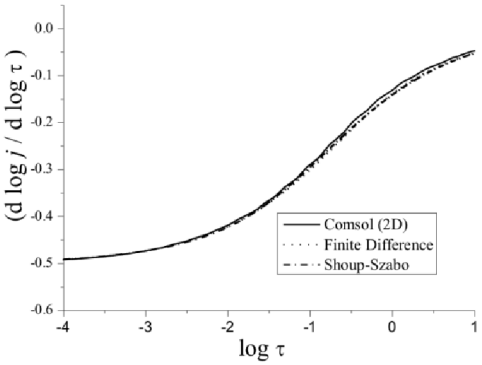



Figure 2.10a – Comparison of rate of change of logarithmic flux against rate of change of logarithmic time for the Shoup-Szabo equation, FEM simulation, and FD simulation

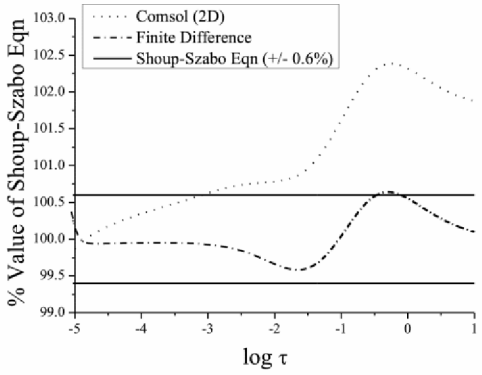



Figure 2.10b – Comparison of FEM simulation and FD simulation with respect to the value of the Shoup-Szabo equation

Figure 2.10a shows the comparison of the FEM package to follow the Shoup-Szabo equation for the potential step chronoamperometry transient at a 10 micron radius microdisc in two dimensions compared to FD simulation [41]. All three lines are relatively close, but if a plot

of error against log $\tau$ is made such as Figure 10b, it is clear that the FEM simulation error quickly exceeds the relative error inherent in the Shoup-Szabo equation, whereas FD simulation stays within reasonable tolerance.

### 2.3.5 Simulation Timing

One of the marketable benefits of FEM simulation and adaptive algorithms is the ability to increase granularity where it matters and provide results as quickly as possible. COMSOL decreases total simulation time utilising a variety of methods – parallelism of the internal algorithms to take advantage of multi-CPU core hardware, and an adaptive time grid. The adaptive time grid has a direct impact on results – if the time grid marches on too fast in areas of activity, the algorithm will switch to finely divided timesteps to get to what it believes is the correct result. This is seen in Figure 11a, which shows a reversible EC cyclic voltammogram (CV) at a planar electrode recorded between $E_f^\theta$ ± 1V at 1 $Vs^{-1}$, and the subsequent error that is produced if no restriction is placed on the time marching algorithm. There are two ways to manipulate this algorithm – by reducing absolute tolerance such that successive timesteps have to match closer criteria and deviate less from each other, or to put a limit on the maximum step size. Both options have a negative impact on the time taken to reach the final solution. Figure 2.11b shows the same simulation as Figure 2.11a, but with a factor 100 reduction in absolute tolerance.

The time taken for the simulation of the linear sweeps in Figure 8a was measured to take 506 seconds for COMSOL going from -1V to 3V at $1Vs^{-1}$ with 10000 grid elements and a maximum time step of 0.01V. For the equivalent FD simulation, with much smaller timesteps, only 476 seconds were taken. However, it should be noted that the FEM simulation was adapted for multithreading, so was at times using more than one CPU core,

whereas the FD simulation is not multithreaded. Also of note is that the FEM simulation actually failed to converge consecutive time steps after 2.6V, and thus no data above 2.6V was obtained, whereas the FD simulation was wholly successful.

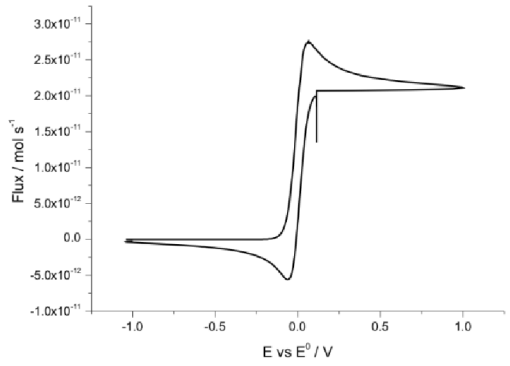



Figure 2.11a – FEM CV showing default timing settings

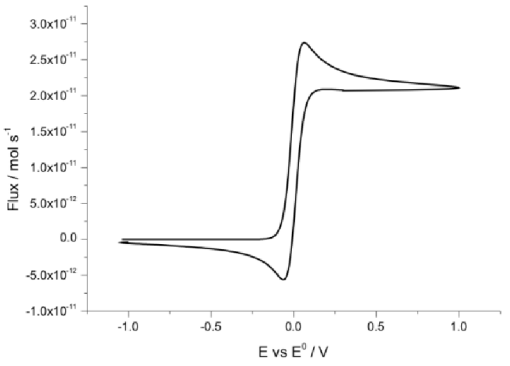



Figure 2.11b – FEM CV showing user defined timing settings

## 2.4 Conclusions

With respect to electrochemistry, 'off-the-shelf' general engineering software should be viewed with caution. Electrochemical simulation requires a very high degree of accuracy, more so as systems become more complex. Algorithmic methods and meshing methods are often designed to increase throughput and accuracy, but one should be aware of the possible caveats these 'enhancements' may imply. COMSOL Multiphysics™ is a software solution in a number of fields as a generic finite element solver, and is not specific to electrochemistry. However, in electrochemistry, the system modelled must be well defined to achieve appropriate quantitative results, as shown by the method taken by Godino et al. [39]. However, successful 'general engineering' modelling is often difficult when working with new theoretical phenomena, rather than theoretically established experimental conditions, or even novel experiments for which there is no prior theory. Nevertheless, as

highlighted in recent literature, under appropriate model conditions, the software will follow trends for optimisation of known methods [1,2]. As such, it may provide a quicker route to those answers due to the attractive and simple user interface, as well as circumnavigating in-house software development. It should be noted that one of the strengths of this software is its graphical and postprocessing functions, useful for teaching aids and mechanistic visualisations where models can be simulated [30].

### 2.5 How to Probe New Systems

Despite experimentalists turning to general engineering finite element software, there are commonplace techniques readily available for electrochemical theoreticians to probe new and different electrochemical phenomena. As mentioned previously in this chapter – the standard in this area is often finite difference simulation, which we will examine next as part of this thesis, in the case of anodic stripping voltammetry.

### Bibliography


[1] N. Mahnke, A. Markovic, H. Duwensee, F. Wachholz, G.U. Flechsig, U. van Rienen, Sensors and Actuators B-Chemical 137 (2009) 363-369.
[2] J.T. Cheng, European Physical Journal-Applied Physics 44 (2008) 87-99.
[3] J.D. Guo, E. Lindner, Anal. Chem. 81 (2009) 130-138.
[4] M. Mazurenka, L. Wilkins, J.V. Macpherson, P.R. Unwin, S.R. Mackenzie, Anal. Chem. 78 (2006) 6833-6839.
[5] M. Schnippering, P.R. Unwin, J. Hult, T. Laurila, C.F. Kaminski, J.M. Langridge, R.L. Jones, M. Mazurenka, S.R. Mackenzie, Electrochem. Commun. 10 (2008) 1827-1830.
[6] M.N. Holder, C.E. Gardner, J.V. Macpherson, P.R. Unwin, J. Electroanal. Chem. 585 (2005) 8-18.
[7] C.A. Basha, M.S. Shankar, V. Boovaragavan, S.R. Pullabhotla, Chemical Engineering & Technology 32 (2009) 659-672.
[8] S.W. Feldberg, C. Auerbach, Anal. Chem. 36 (1964) 505-&.
[9] M.L. Longmire, M. Watanabe, H. Zhang, T.T. Wooster, R.W. Murray, Anal. Chem. 62 (1990) 747-752.
[10] C.P. Andrieux, B. Limoges, D. Marchal, J.M. Savéant, Anal. Chem. 78 (2006) 3138-3143.
[11] D. Menshykau, I. Streeter, R.G. Compton, J. Phys. Chem. C 112 (2008) 14428-14438.
[12] O.V. Klymenko, C. Amatore, I. Svir, Anal. Chem. 79 (2007) 6341-6347.
[13] C. Amatore, K. Knobloch, L. Thouin, J. Electroanal. Chem. 601 (2007) 17-28.
[14] E.J.F. Dickinson, J.G. Limon-Petersen, N.V. Rees, R.G. Compton, J. Phys. Chem. C 113 (2009) 11157-11171.

[15] P.J. Welford, B.A. Brookes, V. Climent, R.G. Compton, J. Electroanal. Chem. 513 (2001) 8-15.
[16] A. Beckmann, B.A. Coles, R.G. Compton, P. Gründler, F. Marken, A. Neudeck, J. Phys. Chem. B 104 (2000) 764-769.
[17] G.G. Wildgoose, D. Giovanelli, N.S. Lawrence, R.G. Compton, Electroanalysis 16 (2004) 421-433.
[18] R.G. Compton, J.C. Eklund, S.D. Page, T.J. Mason, D.J. Walton, J. Appl. Electrochem. 26 (1996) 775-784.
[19] E.J.F. Dickinson, I. Streeter, R.G. Compton, J. Phys. Chem. B 112 (2008) 4059-4066.
[20] S.P. Sullivan, M.L. Johns, S.M. Matthews, A.C. Fisher, Electrochem. Commun. 7 (2005) 1323-1328.
[21] I.E. Henley, A.C. Fisher, R.G. Compton, C.E. Banks, J. Phys. Chem. B 109 (2005) 7843-7849.
[22] C. Amatore, J. Pinson, J.M. Savéant, A. Thiebault, J. Electroanal. Chem. 107 (1980) 59-74.
[23] C.E. Banks, R.G. Compton: Understanding Voltammetry, World Scientific, Singapore, 2011.
[24] T.J. Davies, R.G. Compton, J. Electroanal. Chem. 585 (2005) 63-82.
[25] D. Britz, in D. Britz (Ed.), Digital Simulation in Electrochemistry. Springer, New York, 2005, p. 187.
[26] M. Rudolph, J. Electroanal. Chem. 338 (1992) 85-98.
[27] F. Marken, Y.C. Tsai, B.A. Coles, S.L. Matthews, R.G. Compton, New J. Chem. 24 (2000) 653-658.
[28] I.J. Cutress, F. Marken, R.G. Compton, Electroanalysis 21 (2009) 113-123.
[29] R.W. French, S.N. Gordeev, P.R. Raithby, F. Marken, J. Electroanal. Chem. 632 (2009) 206-210.
[30] Y. Kitazumi, Review of Polarography 55 (2009) 39-45
[31] E.J.F. Dickinson, I. Streeter, R.G. Compton, J. Phys. Chem. C 112 (2008) 11637-11644.
[32] A. Oleinick, C. Amatore, I. Svir, Electrochem. Commun. 6 (2004) 588-594.
[33] V.G. Levich: Physicochemical Hydrodynamics, Prentice-Hall, New Jersey, 1962.
[34] R.S. Parikh, K.C. Liddell, J. Electrochem. Soc. 136 (1989) 679-688.
[35] J.A. Cooper, R.G. Compton, Electroanalysis 10 (1998) 141-155.
[36] O.V. Klymenko, A.I. Oleinick, C. Amatore, I. Svir, Electrochim. Acta 53 (2007) 1100-1106.
[37] A.J. Bard, L.R. Faulkner: Electrochemical Methods: Fundamentals and Applications, Wiley, Phoenix, 2001.
[38] J.A. Alden, R.G. Compton, J. Phys. Chem. B 101 (1997) 8941-8954.
[39] N. Godino, X. Borrise, F.X. Munoz, F.J. del Campo, R.G. Compton, J. Phys. Chem. C 113 (2009) 11119-11125.
[40] A.S. Barnes, I. Streeter, R.G. Compton, J. Phys. Chem. C 112 (2008) 8690-8693.
[41] D. Shoup, A. Szabo, J. Electroanal. Chem. 140 (1982) 237-245.

## Chapter 3 - The Implicit Finite Difference Approach, Applied to Anodic Stripping Voltammetry

Algorithm development can take on many approaches, some of which are part of this thesis – applying theory to a new area, developing new solutions and using novel methods to previous solutions are just some examples. This chapter deals with the first approach mentioned – applying theory to a new area. In this work, the task was to model anodic stripping voltammetry (explained below) at microdisks and microdisk arrays. By using theory used to solve the problem at macroelectrodes [1-3], this chapter details the work in introducing a radial diffusion element in a microdisk system. Also in this chapter, the implicit finite difference approach of electrochemical simulation is introduced, coupled with the alternating direction implicit method of solving such systems, which has been widely applied in recent electrochemical simulation theory literature [4-10].

### 3.1 Finite Difference Method

For a grid of points or nodes over a simulation space, each point in the space can describe a number of factors relating to the simulation – concentration, electric field, temperature, and so on. In terms of concentration, the material balance gradients are approximated to the differences in the concentrations of surrounding points. Consider the concentration gradient of species $C$ at point $x$ in one dimension, where $[C]_x$ describes the concentration of $C$ at $x$:

$$\frac{\partial [C]}{\partial x} \approx \frac{[C]_{x+1} - [C]_{x-1}}{2\Delta x} \qquad [3.1]$$

The second derivative around point $x$ is determined by combining the half-differences from the adjacent half-points next to $x$:

$$\frac{\partial^2[C]}{\partial x^2} \approx \frac{\frac{\partial[C]}{\partial x}\Big|_{x+1/2} - \frac{\partial[C]}{\partial x}\Big|_{x-1/2}}{\Delta x} \approx \frac{\frac{[C]_{x+1}-[C]_x}{\Delta x} - \frac{[C]_x-[C]_{x-1}}{\Delta x}}{\Delta x} \approx \frac{[C]_{x+1}-2[C]_x+[C]_{x-1}}{(\Delta x)^2} \quad [3.2]$$

Equations 3.1 and 3.2 can be applied to the partial differential equation based electrochemical systems, as described in Chapter 1, to reveal a set of linear equations which can be solved. This method essentially describes the simulation space as a regular grid (or irregular if an expanding grid is used – this is examined in more detail later), which is often compatible with typical electrochemical systems.

Applying this to Fick's first law in one dimension gives a series of linear equations which can be repeatedly solved over time:

$$\frac{\partial[C]}{\partial t} \approx D\left(\frac{\partial^2[C]}{\partial x^2}\right) = D\left(\frac{[C]_{x+1}-2[C]_x+[C]_{x-1}}{(\Delta x)^2}\right) \quad [3.3]$$

These equations are either solved explicitly or implicitly. In the explicit method, the concentrations at time $t$ are known and used to calculate time $t+1$:

$$\frac{[C]_x^{t+1}-[C]_x^t}{\Delta t} = D\left(\frac{[C]_{x+1}^t-2[C]_x^t+[C]_{x-1}^t}{(\Delta x)^2}\right) \quad [3.4]$$

$$[C]_x^{t+1} = \frac{D\Delta t}{(\Delta x)^2}([C]_{x+1}^t - 2[C]_x^t + [C]_{x-1}^t) + [C]_x^t \quad [3.5]$$

The explicit method is covered in more detail in Chapter 5, including conditions of stability. The implicit method takes the view that the concentrations at time $t+1$ are a series of unknowns, and the equations are thus coupled into a series of simultaneous equations with an equal set of unknowns, which must be solved together:

$$\frac{[C]_x^{t+1}-[C]_x^t}{\Delta t} = D\left(\frac{[C]_{x+1}^{t+1}-2[C]_x^{t+1}+[C]_{x-1}^{t+1}}{(\Delta x)^2}\right) \quad [3.6]$$

$$[C]_x^t = [C]_x^{t+1} - \frac{D\Delta t}{(\Delta x)^2}([C]_{x+1}^{t+1} - 2[C]_x^{t+1} + [C]_{x-1}^{t+1}) \quad [3.7]$$

The implicit method is algorithmically more complex than the explicit method, but does offer the advantage of unconditional stability with respect to time.

### 3.2 The Alternating Direction Implicit (ADI) Method

For a system in two dimensions (for ease of future use, these are labelled $r$ and $z$), such as a microdisk, the linear system has to be solved in both directions using Fick's Laws:

$$\frac{\partial[C]}{\partial t} \approx D\left(\frac{\partial^2[C]}{\partial z^2} + \frac{\partial^2[C]}{\partial r^2} + \frac{1}{r}\frac{\partial[C]}{\partial r}\right) =$$

$$D\left(\frac{[C]_{z+1}-2[C]_z+[C]_{z-1}}{(\Delta z)^2} + \frac{[C]_{r+1}-2[C]_r+[C]_{r-1}}{(\Delta r)^2} + \frac{1}{r}\frac{[C]_{r+1}-[C]_{r-1}}{2\Delta r}\right) \qquad [3.8]$$

The alternating direction implicit (ADI) method is a straightforward solution to solving what are essentially two dimensional simultaneous equations whilst retaining a high degree of algorithm stability.

ADI splits equation 3.8 into two half time steps – by treating one dimension explicitly and the other dimension implicitly in the same half time step. Thus the explicit values known in one direction are fed into the series of simultaneous equations to solve the other direction. For example, using the $r$ direction explicitly to solve the $z$ direction implicitly:

$$\frac{[C]_x^{t+1}-[C]_x^t}{\frac{1}{2}\Delta t} = D\left(\frac{[C]_{z+1}^{t+1}-[C]_z^{t+1}+[C]_{z-1}^{t+1}}{(\Delta z)^2} + \frac{[C]_{r+1}^t-2[C]_r^t+[C]_r^t}{(\Delta r)^2} + \frac{1}{r}\frac{[C]_{r+1}^t-[C]_{r-1}^t}{2\Delta r}\right) \qquad [3.9]$$

$$[C]_x^{t+1} - \frac{1}{2}D\Delta t\left(\frac{[C]_{z+1}^{t+1}-[C]_z^{t+1}+[C]_{z-1}^{t+1}}{(\Delta z)^2}\right) = \frac{1}{2}D\Delta t\left(\frac{[C]_{r+1}^t-2[C]_r^t+[C]_r^t}{(\Delta r)^2}\right) + \frac{\frac{1}{2}D\Delta t}{r}\left(\frac{[C]_{r+1}^t-[C]_{r-1}^t}{2\Delta r}\right) + [C]_x^t$$

[3.10]

By solving equation 3.10 for the concentrations in the $z$ direction, the next half time step concentrations can be calculated for the $r$ direction, and so on until the desired time in

simulation is achieved. By using this method, the problem of anodic stripping voltammetry is discussed below.

### 3.3 Anodic Stripping Voltammetry

Stripping voltammetry is a popular electroanalytical methodology which offers high sensitivity and, often, good selectivity particularly for the detection of metal ions [11,12]. Stripping analysis is a two-step technique. The first step utilises either electrolytic deposition or absorption of the metal ions, and hence preconcentration, onto the electrode surface. The subsequent step involves application of a potential sweep (oxidising in Anodic Stripping Voltammetry, ASV, and reducing in Cathodic Stripping Voltammetry, CSV) so as to induce the electro-dissolution of the deposited material. The charge passed during the second step reflects the amount of material on the electrode surface and hence the concentration of the metal ion in solution. The key feature of the experiment arises since the pre-concentration leads to a significant improvement of both sensitivity and the limit of detection as compared to the direct solution phase electroanalytical detection. Sub-nanomolar concentrations can often be accessed in favourable systems.

Classically, ASV has been conducted using mercury electrodes so that the reduction of the metal ion leads to the formation of an amalgam. More recently, solid electrodes as supports for Hg films (or functional membranes) have been introduced, and materials such as boron-doped diamond [13-15] and edge plane pyrrolytic graphite [16-18] have considerable promise whilst screen printed electrodes provide a cheap, disposable alternative [19].

The sensitivity of the ASV technique is related to the mass transport conditions under which the pre-concentration step takes place. For this reason the ASV experiment has been undertaken under flowing conditions using channel [2,20-22] or wall-jet electrodes [23] as well as under insonation [24-27] or microwave irradiation [28-31], where the presence of strong convection leads to very considerable improvements in sensitivity. At the same time, analogous improvement can be obtained under diffusion only conditions by switching from planar to convergent diffusion regimes. In experimental practice this means a change from a macro to microelectrode geometry, most usually implemented in the form of a microdisk electrode. Sensitivity can be further enhanced by forming arrays of the microelectrodes, resulting in an improved signal-to-noise ratio and superior analytical performance [31]. In particular, several papers have been published describing the fabrication and characterisation of iridium based mercury ultramicroelectrode arrays [32-34] and these function much as classical ASV via the formation and re-oxidation of amalgams; except that the convergent diffusion regime leads to appreciable sensitivity enhancement.

Whilst the theory of ASV has been well developed for macroelectrodes under planar diffusion using solid electrodes [1] and for mercury films under convective conditions [2,3], the response of mercury film microelectrodes and arrays of these electrodes is essentially unexplored.

### 3.4. Theory

#### 3.4.1 Experimental System

The stripping process of a thin film of pre-concentrated solid $B$ by removing an electron occurs according to the following equation below:

$$B(ads) - e^{-} \rightleftharpoons A^{+}(aq) \qquad [3.11]$$

Experimentally, species $B$ is either in the form of a molecule adsorbed on to the surface of the electrode, or alternatively dissolves in a thin layer of mercury at the electrode surface. For theoretical purposes, these two forms of species $B$ are considered interchangeable. The partial differential equation describing diffusion to and from a microdisk electrochemical surface is represented by Fick's second law [28, 29]. Due to the symmetrical nature of a microdisk, one plane of the disk is simulated in cylindrical coordinates, and the result is integrated over the area of the disk. As shown in Figure 3.1, $r$ and $z$ are the ordinates of the cylindrical system:

$$\frac{\partial[A]}{\partial t} = D_A \nabla^2 = D_A \left( \frac{\partial^2[A]}{\partial z^2} + \frac{\partial^2[A]}{\partial r^2} + \frac{1}{r}\frac{\partial[A]}{\partial r} \right) \qquad [3.12]$$

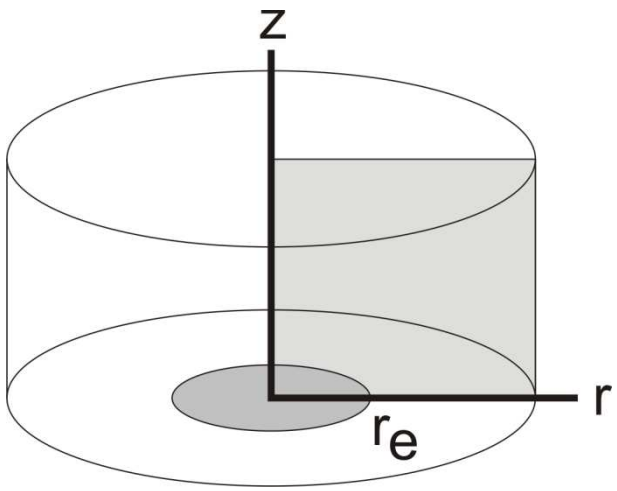



Figure 3.1 Schematic representation of a microdisk electrode in cylindrical space.

### 3.4.2 Kinetics

This process of a single electron transfer at the microdisk surface is simulated by Butler-Volmer kinetics [35-38] (as described in Chapter 1) according to equation 3.13 below. All variables have been previously defined and are found at the front of this thesis.

$$D_A \frac{\partial [A]}{\partial z}\Big|_{z=0} = k^0 \left( e^{-\frac{\alpha F}{RT}\left(E - E_f^0\right)}[A] - e^{-\frac{(1-\alpha)F}{RT}\left(E - E_f^0\right)}[B] \right) \quad [3.13]$$

Note that in writing the transfer coefficient for the reduction step as $\alpha$ and for the oxidation step as $(1-\alpha)$, the assumption is made that there is no change of electrochemical mechanism across the range of applied potential.

The further simulation assumption is made that the film layer of species $B$ is sufficiently thin that diffusion of the reduced material through the film is so fast that no concentration gradients build up within the film (as represented by condition 3.14).

$$D_{B(film)} = 0 \quad [3.14]$$

The rate of stripping of species $B$ is governed by both Butler-Volmer kinetics (as shown in equation 3.13) and the film thickness, $l$, of species $B$ on the surface of the microdisk in relation to the concentration gradient of species $A$, such that equation 3.15 also applies.

$$l \frac{\partial [B]}{\partial t} = D_A \frac{\partial [A]}{\partial z}\Big|_{z=0} \quad [3.15]$$

### 3.4.3 Dimensionless Parameters

The parameter space used for modelling is transformed from Cartesian three dimensional $xyz$ space to two dimensional cylindrical $rz$ space. Furthermore, the simulation is reduced by its scaling factors into dimensionless units, as per Table 3.1 [35]. Although a further discussion on dimensionless units is provided at the end of this thesis in Appendix A, these transforms allow for independence against the scaling parameters such as the electrode radius.

**Table 3.1 Parameter Space Transformations**

| Parameter | Variable | Dimensionless Variable | Expression |
|---|---|---|---|
| Radial ordinate | $r$ | $R$ | $R = r/r_e$ |
| Normal ordinate | $z$ | $Z$ | $Z = z/r_e$ |
| Normalised time | $t$ | $\tau$ | $\tau = Dt/r_e^2$ |
| Concentration of species B | $[B]$ | $B$ | $B = [B]/[B]_{bulk}$ |
| Concentration of species A | $[A]$ | $A$ | $A = [A]/[B]_{bulk}$ |
| Film thickness | $l$ | $L$ | $L = l/r_e$ |
| Potential | $E$ | $\theta$ | $\theta = F\left(E - E_f^0\right)/RT$ |
| Scan rate | $v$ | $\sigma$ | $\sigma = r_e^2 Fv/DRT$ |
| Rate constant | $k^0$ | $K^0$ | $K^0 = k^0 r_e/D$ |

### 3.4.4 Dimensionless Kinetics

Through application of the transformations shown in table 3.1, Fick's second law becomes equation 3.16, and the Butler-Volmer kinetic equation at the electrode surface becomes equation 3.17.

$$\frac{\partial A}{\partial \tau} = \frac{\partial^2 A}{\partial R^2} + \frac{\partial^2 A}{\partial R^2} + \frac{1}{R}\frac{\partial A}{\partial R} \quad [3.16]$$

$$\left.\frac{\partial A}{\partial Z}\right|_{Z=0} = K^0\left(e^{-\alpha\theta}A_0 - e^{\theta(1-\alpha)}B_0\right) \quad [3.17]$$

The deposition layer boundary condition is also applied in dimensionless form, transforming equation 3.15 to equation 3.18.

$$L\frac{\partial B}{\partial \tau} = \left.\frac{\partial A}{\partial Z}\right|_{Z=0} \quad [3.18]$$

### 3.4.5 Simulation Space

The simulation space used has a finely divided rectangular two-dimensional expanding mesh, as introduced by Gavaghan [39], resulting in a high density of nodes at the symmetric simulation axis, the electrode, and the electrode edge, as shown in the $R$ direction and $Z$ direction in figures 3.2 and 3.3 respectively.

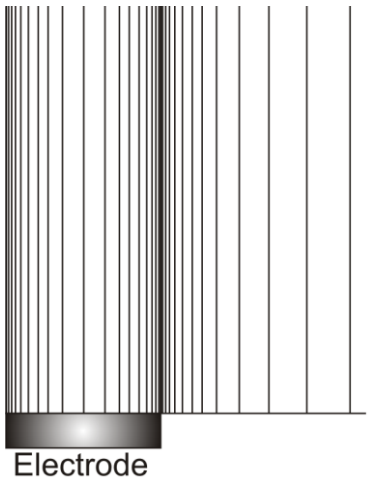



Figure 3.2 – Mesh Spacing representation in the R direction.

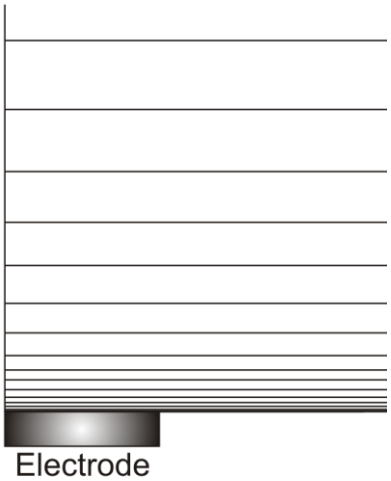



Figure 3.3 – Mesh Spacing representation in the Z direction.

The initial spacing of the mesh ($\gamma$) used in this chapter is equal to $1x10^{-5}$, and the expansion coefficient ($h$) is 1.125 [39]. The limits of the mesh are described by the

diffusional domain constraints. These constraints are set as a distance $6\sqrt{\tau_{max}}$ from the electrode, which is known to sufficiently exceed the diffusion layer in all cases [40].

Solving of equations 3.16-3.18, is achieved through the Alternating Direction Implicit (ADI) algorithm [41], which as described above, is solved for each time step in the $R$ direction, followed by the $Z$ direction.

### 3.4.6 Application of the Expanding Mesh ADI

Expansion of equation 3.16 using an expanding mesh ADI introduces new notation. For a node in the simulation, the distance to the next node in the positive $R$ direction is given as $\Delta R_+$, and in the negative $R$ direction is given as $\Delta R_-$. Similar notation is used in the Z direction ($\Delta Z_+, \Delta Z_-$).

In conjunction with this notation, expansion of equation 3.16 gives:

$$\frac{A^{\tau}-A^{\tau-\frac{1}{2}}}{\Delta\tau} = \frac{\frac{A_{R+1}-A_R}{\Delta R_+}-\frac{A_R-A_{R-1}}{\Delta R_+}}{\frac{1}{2}(\Delta R_+ + \Delta R_-)} + \frac{\frac{A_{Z+1}-A_Z}{\Delta Z_+}-\frac{A_Z-A_{Z-1}}{\Delta Z_+}}{\frac{1}{2}(\Delta Z_+ + \Delta Z_-)} + \frac{1}{R}\frac{A_{R+1}-A_{R-1}}{\Delta R_+ + \Delta R_-} \quad [3.19]$$

For ease of use, equation 3.19 is simplified by 3.20-3.22 to give equation 3.23.

$$\lambda R = \frac{\Delta\tau}{\frac{1}{2}(\Delta R_+ + \Delta R_-)}; R_+ = \frac{\lambda R}{\Delta R_+}; \lambda R_- = \frac{\lambda R}{\Delta R_-} \quad [3.20]$$

$$\lambda Z = \frac{\Delta\tau}{\frac{1}{2}(\Delta Z_+ + \Delta Z_-)}; Z_+ = \frac{\lambda Z}{\Delta Z_+}; \lambda Z_- = \frac{\lambda Z}{\Delta Z_-} \quad [3.21]$$

$$\lambda RR = \frac{1}{R}\frac{\Delta\tau}{(\Delta R_+ + \Delta R_-)} \quad [3.22]$$

$$A^{\tau} - A^{\tau-\frac{1}{2}} = \begin{array}{c} A_{R+1}(\lambda R_+ + \lambda RR) - A_R(\lambda R_+ + \lambda R_-) + A_{R-1}(\lambda R_- - \lambda RR) \\ + A_{Z+1}(\lambda Z_+) - A_Z(\lambda Z_+ + \lambda Z_-) + A_{Z-1}(\lambda Z_-) \end{array} \quad [3.23]$$

As explained in Chapter 3.2, in ADI each direction is calculated on alternate half-time steps. Thus on the first half time step, the $R$ direction is implicitly calculated for $\tau$ using the $Z$ direction at as input values, and vice versa on the subsequent step.

The time step equations that result from equation 3.23 are solved by the Thomas Algorithm for tri-diagonal matrices [42].

### 3.4.7 Boundary Conditions

Application of the deposition layer boundary condition at the electrode surface (equation 3.24 in dimensionless units) and the Butler-Volmer kinetic paradigm (equation 3.25 in dimensionless units) combine into equation 3.26:

$$B_0^{\tau-\frac{1}{2}} = B_0^{\tau} - \lambda(A_1^{\tau} - A_0^{\tau}), \text{ where } \lambda = \frac{\Delta\tau}{L\Delta Z_0} \text{ and } L \neq 0 \quad [3.24]$$

$$B_0 = A_0\left(\lambda^K + e^{-\theta}\right) - A_1(\lambda^K), \text{ where } \lambda^K = \frac{1}{\Delta Z_0 K^0 e^{\theta(1-\alpha)}} \text{ and } K^0 \neq 0 \quad [3.25]$$

$$B_0^{\tau-\frac{1}{2}} = A_0^{\tau}\left(\lambda^K + \lambda + e^{-\theta}\right) - A_1^{\tau}(\lambda^K + \lambda), \text{ where } L \neq 0 \text{ and } K^0 \neq 0 \quad [3.26]$$

Equation 3.26 thus relates $B_0^{\tau-\frac{1}{2}}$ and $A_0^{\tau}$ for the next half time step at the electrode surface. At the simulation boundaries exceeding the diffusion layer, it is assumed that the bulk solution dominates, and the symmetry axis boundary has no change in flux. These boundary conditions in the system are noted in Table 3.2.

**Table 3.2 Other Boundary Conditions**

| Condition | Boundary Region | Equation |
|---|---|---|
| Symmetric Axis | $R = 0$ | $\partial A/\partial R = 0$ |
| Insulator | $Z = 0; R > 1$ | $\partial A/\partial Z = 0$ |
| Exceeding the Diffusion Layer | $R = 6\sqrt{\tau_{max}}$<br>$Z = 6\sqrt{\tau_{max}}$ | $A = A_{bulk}$ |
| Thin film concentration, $t$ = 0 | $Z = 0; R < 1;\ \tau = 0$ | $B = 1$ |
| Solution, $t$ = 0 | $\tau = 0$ | $A = A_{bulk}$ |
| Everywhere except the film, $t$ = 0 | $R \leq 1$ at $Z = 0$<br>All $R$ at $Z > 0$ | $B = 0$ |

### 3.4.8 Electrode Flux

The concentration flux, $j$, to the electrode surface for time $\tau$ is given by equation 3.27:

$$j_\tau = \int_0^1 R \left.\frac{\partial A}{\partial Z}\right|_{Z=0} dR \qquad [3.27]$$

Due to the distribution of nodes at the electrode surface, the derivative of equation 3.27 is approximated as being linear at the electrode surface.

$$\left.\frac{\partial A}{\partial Z}\right|_{Z=0} \approx \frac{A_1 - A_0}{\Delta Z_0} \qquad [3.28]$$

The integral is then evaluated using the trapezium rule:

$$\int_0^1 f(R)dR \approx \sum_{i=0}^{i_{el}} \Delta R_i \frac{f(R_i)+f(R_{i-1})}{2} \qquad [3.29]$$

For a single diffusion domain, the absolute current at time $\tau$ can be calculated according to equation 3.30:

$$i_\tau = 2\pi F D j_\tau [B]_{initial} \qquad [3.30]$$

### 3.4.9 Ultramicroelectrode arrays

In the case of regular ultramicroelectrode arrays, each electrode can be considered in a separate unit cell. The diffusion layers of neighbouring unit cells will overlap given slow voltage scan rates, resulting in comparatively long experiment times relative to inter-electrode distances, as described in Chapter 1.4.3. The diffusion domain is thus modified in the $R$ direction to a no flux boundary condition, as described by Reller et. al. [43].

The diffusion domain approximation takes the ratio of electroactive surface area to total surface area, represented by a fraction as Θ, and is shown in equation 3.31 where $d$ is the interelectrode distance. Figure 3.4 reiterates the explanation in Chapter 1.4.3 and leads to a modification of the R boundary representative of an array cell, such that equation 3.32 applies:

$$\Theta = \frac{\pi r_e^2}{d^2} = \frac{\pi r_e^2}{\pi r_0^2} = \frac{1}{R_0^2} \therefore R_0 = \frac{1}{\sqrt{\Theta}} \quad [3.31]$$

$$\left.\frac{\partial A}{\partial R}\right|_{R=R_{max}} = 0 \quad [3.32]$$

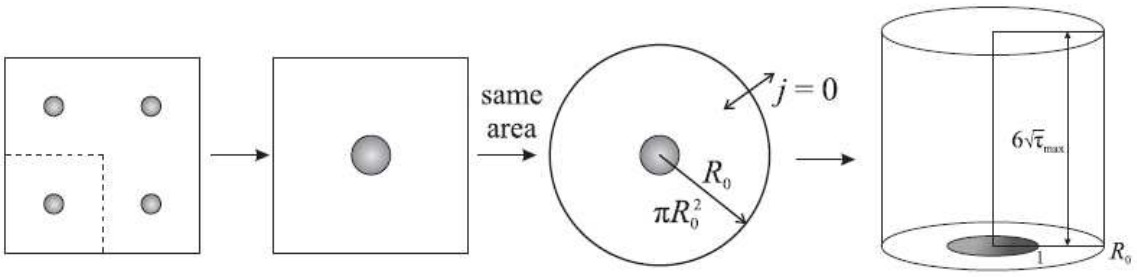



Fig 3.4. Representation of Diffusion Domain Approximation from a regular cubic microarray to cylindrical simulation space.

### 3.4.10. Computer Utilisation

An Intel Core2Quad Q6600 (Kentsfield) running at 3026 MHz with 2GiB of DDR2-800 RAM was used for simulations in this chapter. Microsoft® Visual C++ 2008 Express Edition was used to write and compile the simulation code.

## 3.5 Results and Discussion

### 3.5.1. Modelling Single Diffusion Domains

We consider the effects of changing different variables on the voltammetric signals arising from stripping under different experimentally variable conditions.

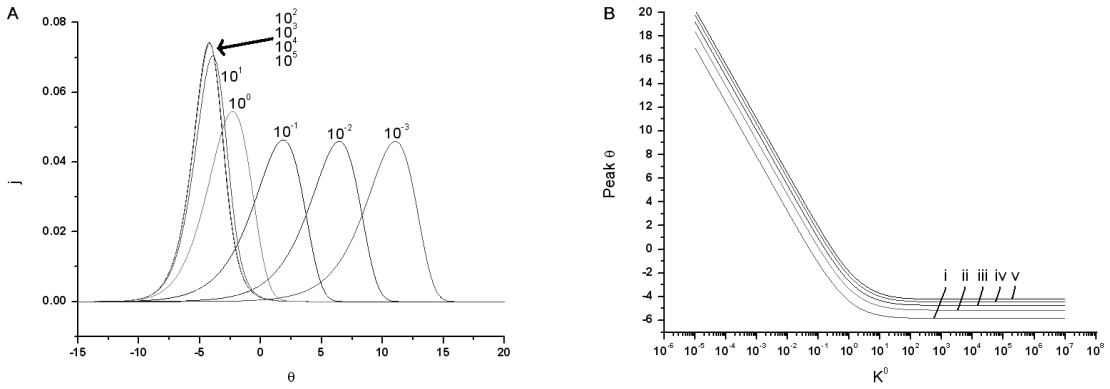



Figure 3.5. (a) Plot showing the variation in voltammetric height and shape as a function of dimensionless rate constant $K^0$ at σ = $10^3$, $L$ = 5x$10^{-4}$. (b) Plot showing the variation of peak $\theta$ as a function of $K^0$ at σ = $10^3$, $L$ = (i) 1x$10^{-4}$, (ii) 2x$10^{-4}$, (iii) 3x$10^{-4}$, (iv) 4x$10^{-4}$, (v) 5x$10^{-5}$.

Figure 3.5 shows a plot of (a) dimensionless flux *j* at the electrode surface as a function of potential and $K^0$ at σ = $10^3$ and $L$ = 5x$10^{-4}$; (b) the variation of peak potential as a function of $K^0$ and $L$ at σ = $10^3$. It can be seen in Figure 3a that as $K^0$ decreases, the peak flux occurs at a higher potential, and the peak flux is reduced; in Figure 3.5b the peak potential follows a negative linear trend as $K^0$ increases until a limit is reached at around $K^0$ = $10^2$. Given a constant scan rate, as $K^0$ decreases, the voltammetric shape flattens

and broadens, but still keeps the same overall charge, as seen in Figure 3.5a. Also, a higher overpotential is required to achieve similar results at lower $K^0$ resulting in a shift in peak potential, as seen in figure 3.5b. The shape of the lines in figure 3.5b is consistent with Butler-Volmer theory, as shown the linear section above $K^0 > 10^1$, and the constant slope of $K^0 < 10^{-1}$.

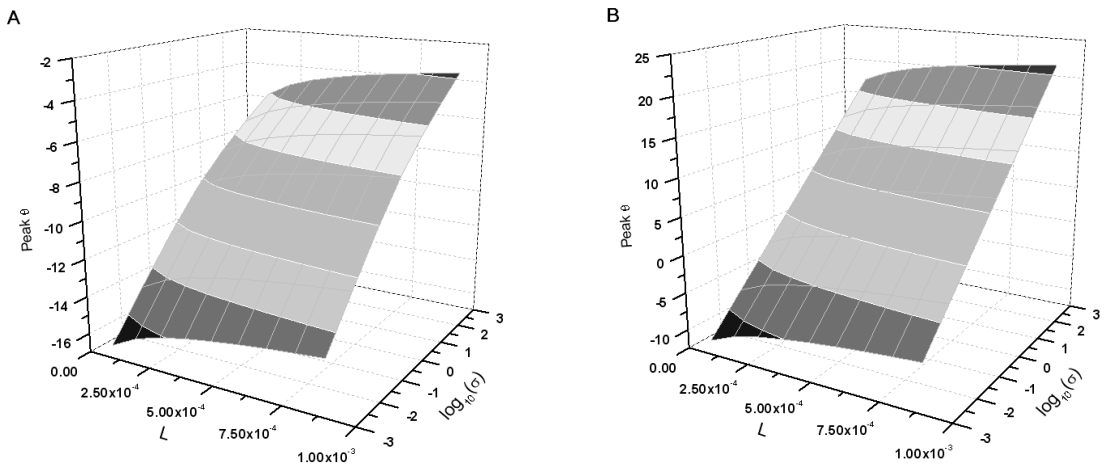



Figure 3.6 Plot showing variation in the dimensionless peak potential as a function of dimensionless film thickness and dimensionless scan rate at $K^0$ = (a) $10^5$, (b) $10^{-5}$.

In figure 3.6, the peak potential is plotted as a function of film thickness and scan rate for low and high rate constants. As film thickness increases, peak potential follows a logarithmic correlation. Also shown is the linear correlation of peak potential as a function of the logarithmic scan rate. At lower scan rates, the total time of the experiment is longer, thus more material is stripped at a lower potential. At high values of dimensionless film thickness L, more time is required to strip the metal from the film, therefore peak $\theta$ occurs at a higher potential.

Figure 3.7 shows a plot of the variation in dimensionless peak width at half height as a function of dimensionless scan rate and dimensionless film thickness where (a) rate constant $K^0$ = $10^5$ and (b) $K^0$ = $10^{-5}$. It can be seen in figure 3.7a that as the scan rate is increased, the peak width at half height increases from a minimum to a maximum. At higher scan rates, the overall experiment time is reduced, and the film is exposed to higher potentials much quicker than at lower scan rates, to a maximum in peak broadening. This is due to stripping becomes diffusion limited. Figure 3.7b shows that at a slower rate constant, the peak width at half height reaches a limit at very slow scan rates.

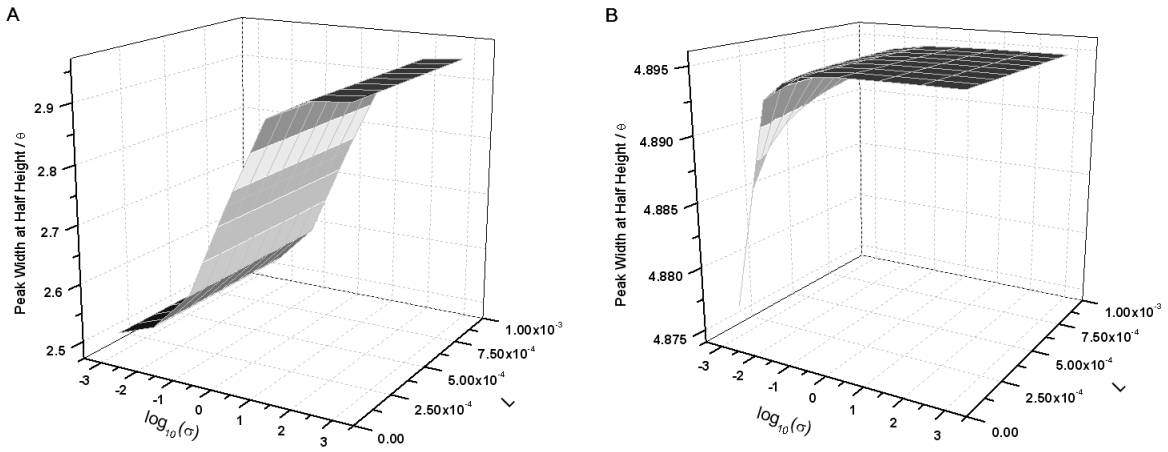



Figure 3.7. Plot showing variation in the dimensionless peak width at half peak height as a function of film thickness and dimensionless scan rate at (a) $K^0$ = $10^5$, (b) $K^0$ = $10^{-5}$.

### 3.5.2. Modelling Arrays of Ultramicroelectrodes

We next turn to results from ultramicroelectrode arrays. The change from an isolated electrode is most prevalent when the diffusion layers of neighbouring domains overlap. This occurs when the unit cell is small, $K^0$ is large, and scan rates are slow. Figure 3.8 shows simulations related to applicable experimental conditions where the electrode radius is 10$\mu$m, a diffusion coefficient of $10^{-5}$ cm $s^{-1}$, and a film thickness of 1nm at three

different scan rates and rate constants. In the figure, the solid line represents an electrode array with unit cell size of 358$\mu$m (such that each electrode can be considered isolated from each other), and the dashed line is of an array with unit cell size 18$\mu$m.

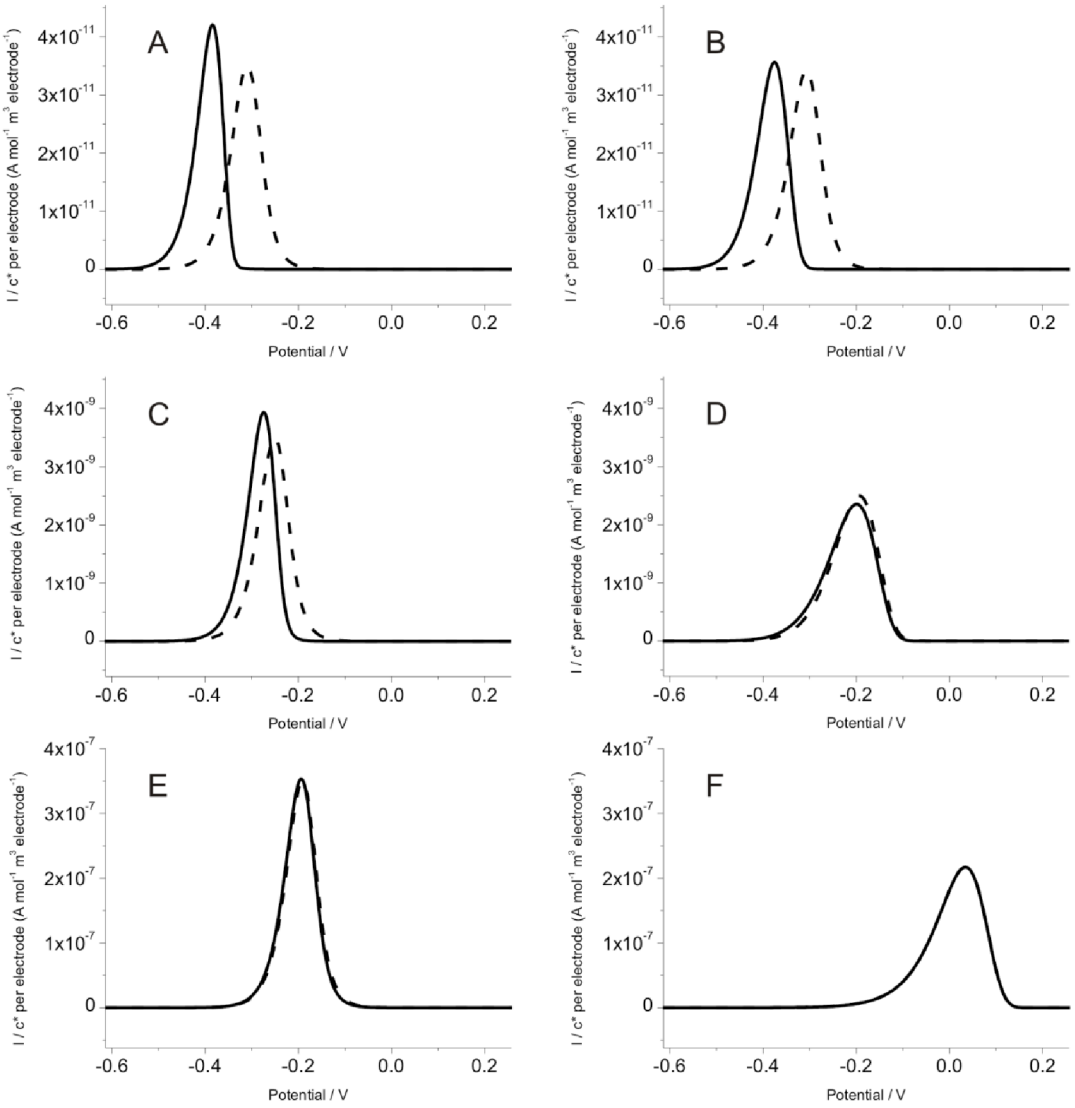



Figure 3.8. Voltammetry showing the effect of the diffusion domain approximation for $l$ =1nm at (A) 1 mV s$^{-1}$, $k^0$ = 1 cm s$^{-1}$, (B) 1 mV s$^{-1}$, $k^0$ = 10$^{-5}$ cm s$^{-1}$, (C) 100 mV s$^{-1}$, $k^0$ = 1 cm s$^{-1}$, (D) 100 mV s$^{-1}$, $k^0$ = 10$^{-5}$ cm s$^{-1}$, (E) 10 V s$^{-1}$, $k^0$ = 1 cm s$^{-1}$, (F) 10 V s$^{-1}$, $k^0$ = 10$^{-5}$ cm s$^{-1}$. The solid line in each voltammogram represents a unit cell width of 358$\mu$m, the dashed line for 18$\mu$m. In (F) both lines fall at the same points for the scale used.

Figure 3.8a shows the situation leading to heavily overlapping diffusion domains between cells – a high rate constant coupled with a slow scan rate. As a result, the smaller unit cell with the overlapping diffusion domain has a lower peak current, a broader peak, and the peak potential occurs later in the scan as compared to isolated electrodes. This is indicative of a lower flux at the electrode surface due to the overlapping diffusion domains. Figure 3.8b shows a smaller peak for the large unit cell, and the gap between peak potentials of the volltamtery is smaller. In figure 3.8b, despite the slower rate constant, the scan rate is still small enough to provide overlapping diffusion domains for the smaller unit cell.

As the scan rate is increased, at higher $k^0$ values (figures 3.8c,e), we expect the diffusion domains of the smaller unit cells overlapping less and less, until they act as independent diffusion domains. Similarly with the slower rate constants (figures 3.8d,f), however the voltammetry between the differing unit cells equate at a lower scan rate.

Figure 3.9 shows concentration profiles of a 10$\mu$m electrode in a 24$\mu$m unit cell at a high $k^0$ (1 cm $s^{-1}$) at differing scan rates at selected points along the voltammogram. From figure 3.9a(i) to figure 3.9c(i) the important observation is the concentration level beyond 10$\mu$m – due to the decreasing scan rate, the diffusion layers of adjacent electrodes overlap to a greater extent.

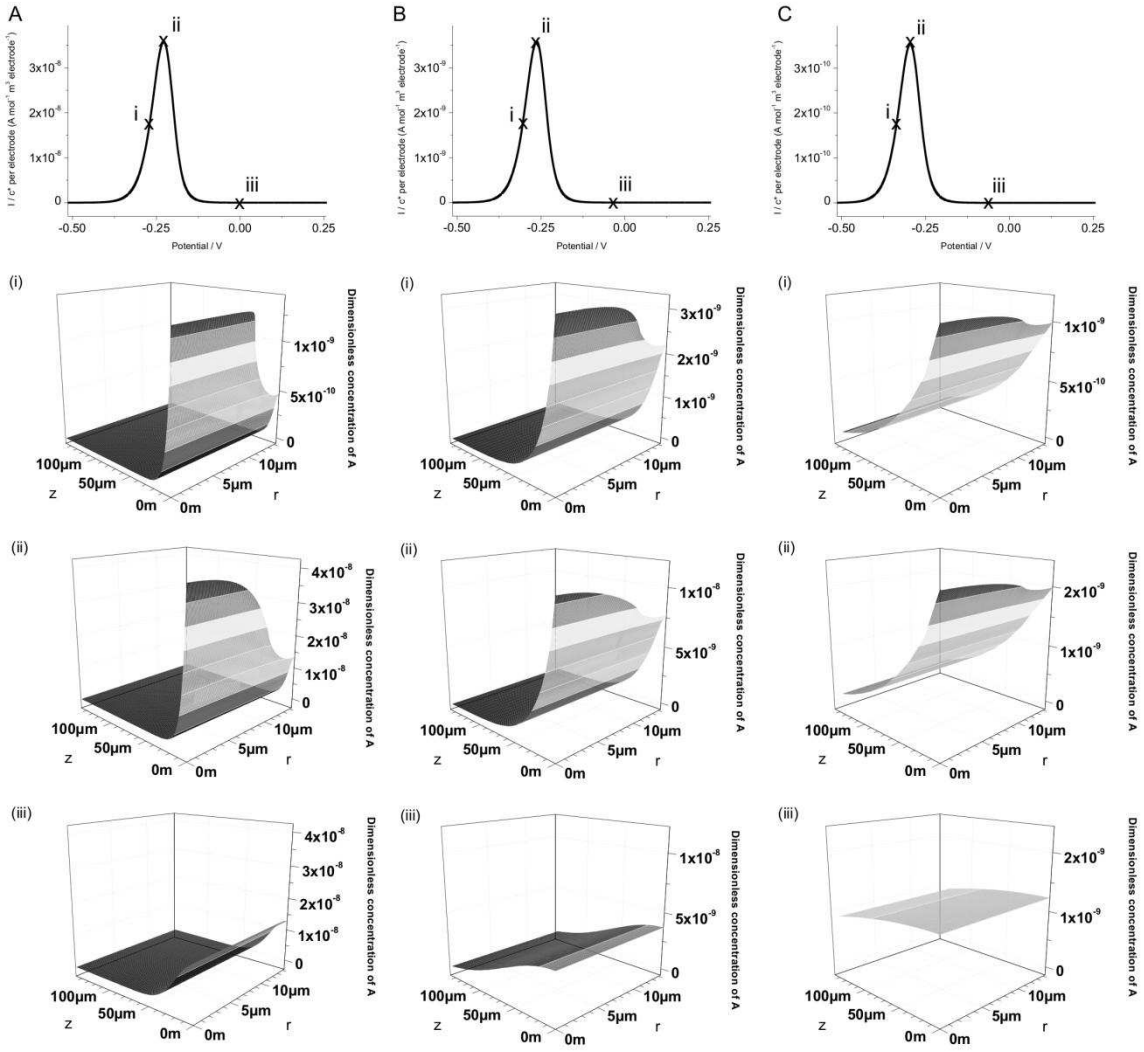



Figure 3.9 Simulated stripping concentration profiles of a 10$\mu$m electrode with a 1nm thick film and D = $10^{-5}$ cm$^2$ s$^{-1}$, $k^0$ = 1 cm s$^{-1}$, $\Theta$ = 0.551 (24$\mu$m unit cell). Scan rates are (A) 1 V s$^{-1}$, (B) 100m V s$^{-1}$ and (C) 10 mV s$^{-1}$, where (i),(ii) and (iii) are the respective points on the voltammogram. I would like to draw attention to the similar scales used in the $z$ direction for better visualisation and comparison – the simulated space often extended beyond the scale as in proportional to experimental time, as described in reference [32]. Also note the (ii) and (iii) graphics for each share the same independent z-axis scale.

Figure 3.10 represents trends in peak potential and peak current as functions of rate constant and coverage at three different scan rates, and confirms trends observed in figure 3.9. Comparison of figure 3.10a(ii), b(ii) and c(ii) is of note, showing the dip in current at higher coverage at lower scan rates. This shows how the overlap of diffusion domains affects peak current per electrode, whereas in 3.10a(ii), the scan is too fast for

diffusion layers to overlap and each electrode acts independently from each other. Figure 3.11 shows the cross section of figure 3.10b(ii), and the increase/decrease in current as coverage increases.

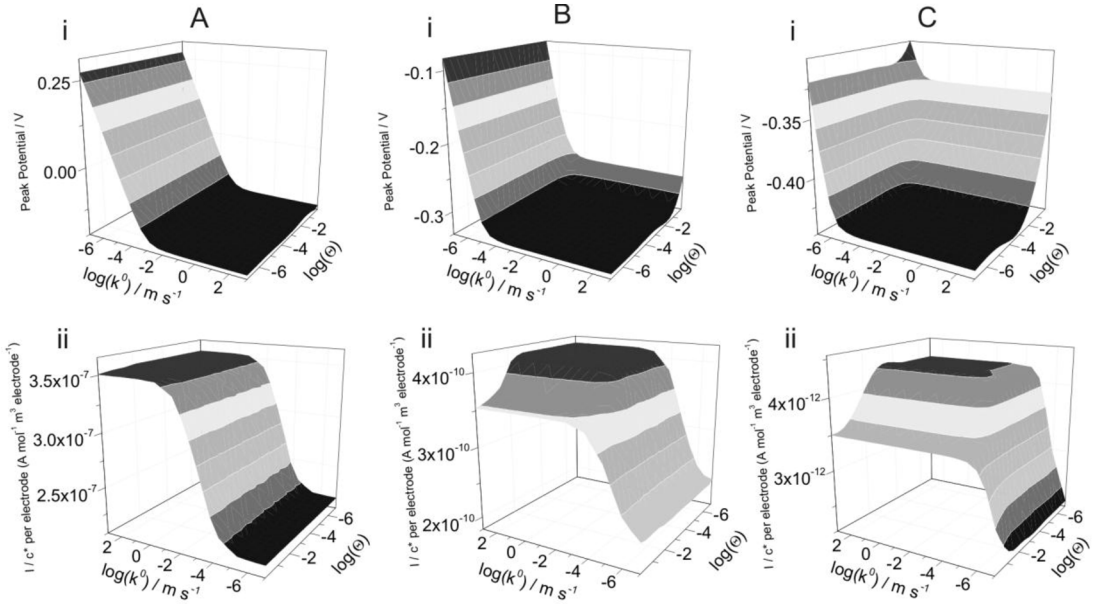



Figure 3.10. Simulated stripping visualisations of trends at $l$ = 1nm for (A) $v$ = 10 V s$^{-1}$, (B) $v$ = 10 mV s$^{-1}$, and (C) $v$ = 1 mV s$^{-1}$, all as a function of $k^0$ and Θ, at an electrode $r_e = 10\mu$m, $D$ = 10$^{-5}$ cm$^2$ s$^{-1}$. Author would like to note that the log(Θ) axes are reversed between (i) and (ii) to show areas of interest.

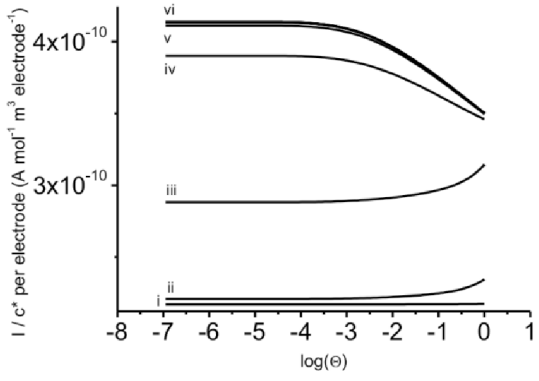



Figure 3.11. Cross sections of Fig 10b(ii), Showing the effect of coverage on current per electrode as a function of $k^0$, where $k^0$=(i) 10$^{-7}$ cm s$^{-1}$, (ii) 10$^{-6}$ cm s$^{-1}$, (iii) 10$^{-5}$ cm s$^{-1}$, (iv) 10$^{-4}$ cm s$^{-1}$, (v) 10$^{-3}$ cm s$^{-1}$, (vi) 10$^{-2}$cm s$^{-1}$to 10$^3$cm s$^{-1}$ (lines of (vi) are equivalent on the scale of the graph).

### 3.6 Conclusions

Two modelling concepts applicable to thin film electrolysis have been considered in this chapter. A two-dimensional approach, from first principles with appropriate modelling techniques and in-depth explanation, has been used to study stripping voltammetry at microelectrodes as one of the main methods of algorithm development. The diffusion domain approximation has been applied to demonstrate diffusional overlapping with adjacent cells in ultramicroelectrode arrays. A major insight for the experimentalist is produced by figures 3.6 and 3.10 which show that the potential of the stripping peak can vary considerably with the experimental parameters. It follows that fingerprinting of species by peak potential, using microelectrodes and microelectrode arrays, can only be undertaken using stripping voltammetry if great care is taken and experimental conditions are reproducible.

### Bibliography


[1] S.E.W. Jones, F.G. Chevallier, C.A. Paddon, R.G. Compton, Anal. Chem. 79 (2007) 4110-4119.
[2] J.C. Ball, R.G. Compton, Electroanalysis 9 (1997) 1305-1310.
[3] J.C. Ball, R.G. Compton, Electroanalysis 9 (1997) 765-769.
[4] K.R. Ward, N.S. Lawrence, R.S. Hartshorne, R.G. Compton, J. Phys. Chem. C 115 (2011) 11204-11215.
[5] J.G. Limon-Petersen, E.J.F. Dickinson, T. Doneux, N.V. Rees, R.G. Compton, J. Phys. Chem. C 114 (2010) 7120-7127.
[6] I.J. Cutress, E.J.F. Dickinson, R.G. Compton, J. Electroanal. Chem. 638 (2010) 76-83.
[7] S.R. Belding, E.J.F. Dickinson, R.G. Compton, J. Phys. Chem. C 113 (2009) 11149-11156.
[8] E.J.F. Dickinson, I. Streeter, R.G. Compton, J. Electroanal. Chem. 625 (2009) 40-46.
[9] S.R. Belding, R. Baron, E.J.F. Dickinson, R.G. Compton, J. Phys. Chem. C 113 (2009) 16042-16050.
[10] E.J.F. Dickinson, I. Streeter, R.G. Compton, J. Phys. Chem. B 112 (2008) 4059-4066.
[11] K.Z. Brainina, E. Neyman: Electroanalytical Stripping Methods, J. Wiley & Sons, New York, 1993.
[12] J. Wang: Analytical Electrochemistry, J. Wiley & Sons, New York, 2000.
[13] F. Wantz, C.E. Banks, R.G. Compton, Electroanalysis 17 (2005) 655-661.
[14] C.H. Goeting, F. Marken, A. Gutierrez-Sosa, R.G. Compton, J.S. Foord, New Diamond Front. Carbon Technol. 9 (1999) 207-228.
[15] R.G. Compton, J.S. Foord, F. Marken, Electroanalysis 15 (2003) 1349-1363.
[16] R.T. Kachoosangi, C.E. Banks, X.B. Ji, R.G. Compton, Anal. Sci. 23 (2007) 283-289.

[17] C.M. Welch, C.E. Banks, S. Komorsky-Lovric, R.G. Compton, Croat. Chem. Acta 79 (2006) 27-32.
[18] C.E. Banks, R.G. Compton, Anal. Sci. 21 (2005) 1263-1268.
[19] R.O. Kadara, N. Jenkinson, C.E. Banks, Sensors and Actuators B-Chemical 138 (2009) 556-562.
[20] S.H. Lieberman, A. Zirino, Anal. Chem. 46 (1974) 20.
[21] R.W. Andrews, D.C. Johnson, Anal. Chem. 48 (1976) 1056-1060.
[22] J.C. Ball, J.A. Cooper, R.G. Compton, J. Electroanal. Chem. 435 (1997) 229-239.
[23] J.C. Ball, R.G. Compton, C.M.A. Brett, J. Phys. Chem. B 102 (1998) 162-166.
[24] C. Agra-Gutierrez, J.C. Ball, R.G. Compton, J. Phys. Chem. B 102 (1998) 7028-7032.
[25] C. Agra-Gutierrez, R.G. Compton, Electroanalysis 10 (1998) 603-612.
[26] C. Agra-Gutierroz, R.G. Compton, Electroanalysis 10 (1998) 204-206.
[27] J.C. Ball, R.G. Compton, J. Phys. Chem. B 102 (1998) 3967-3973.
[28] I.J. Cutress, F. Marken, R.G. Compton, Electroanalysis 21 (2009) 113-123.
[29] M.A. Ghanem, H. Hanson, R.G. Compton, B.A. Coles, F. Marken, Talanta 72 (2007) 66-71.
[30] Y.C. Tsai, B.A. Coles, R.G. Compton, F. Marken, Electroanalysis 13 (2001) 639-645.
[31] O. Ordeig, J. del Campo, F.X. Munoz, C.E. Banks, R.G. Compton, Electroanalysis 19 (2007) 1973-1986.
[32] C. Belmont, M.L. Tercier, J. Buffle, G.C. Fiaccabrino, M. KoudelkaHep, Anal. Chim. Acta 329 (1996) 203-214.
[33] S.P. Kounaves, W. Deng, P.R. Hallock, G.T.A. Kovacs, C.W. Storment, Anal. Chem. 66 (1994) 418-423.
[34] B. Le Drogoff, M.A. El Khakani, P.R.M. Silva, M. Chaker, A.K. Vijh, Electroanalysis 13 (2001) 1491-1496.
[35] E.J.F. Dickinson: Part II. Thesis, Oxford University, Oxford, 2007.
[36] C.E. Banks, R.G. Compton: Understanding Voltammetry, World Scientific, Singapore, 2011.
[37] J.A.V. Butler, Transactions of the Faraday Society 28 (1932) 379-382.
[38] T. Erdey-Grúz, M. Volmer, Z. Phys. Chem. 150 (1930) 203-213.
[39] D.J. Gavaghan, J. Electroanal. Chem. 456 (1998) 1-12.
[40] A.J. Bard, L.R. Faulkner: Electrochemical Methods: Fundamentals and Applications, Wiley, Phoenix, 2001.
[41] D.W. Peaceman, H.H. Rachford, Journal of the Society for Industrial and Applied Mathematics 3 (1955) 28-41.
[42] K.E. Atkinson: Elementary Numerical Analysis, J. Wiley & Sons, New York, 2004.
[43] H. Reller, E. Kirowaeisner, E. Gileadi, J. Electroanal. Chem. 138 (1982) 65-77.

## Chapter 4 - The Era of Parallel Computing

### 4.1 The Importance of Computer Science in Simulation

The interplay of computer simulation and computer science is important in terms of the innovation required to model physical (and electrochemical) phenomena. Rigorous mathematical simulation is historically performed using computers, to obtain results seemingly unfeasible to attempt analytically without severe approximation. Throughout electrochemical simulation literature, the format has remained relatively unchanged: write computer code, perform the simulation, and analyse the results [1-3]. Computer simulation use is often via simulations written many years previously – for the large amount of simple cases (in terms of electrochemistry, a limited about of reactions and species modelled in one dimension) this works well as the simulations are not stressful on modern computational resources. In large and complex situations, often the adaptation and maintenance of large code databases hinders progress into new computational developments.

As computer science develops, in terms of hardware and software, the relevant scientific simulation groups should be aware of what new features are on offer, particularly if the scientific simulation concepts and programs written take a long time to process.

For the hardware side, often upgrading to the latest compatible hardware affords many speed advantages in terms of pure computational throughput, especially if the limiting factor within the simulation is the speed at which calculations are processed (this is not always the case). Moore's Law [4,5], which states that the number of transistors on an integrated circuit doubles approximately every two years, is often a relevant place to

understand where the increased speed over time of a developed computer processor comes from, however in terms of pure speed, a doubling effect is not seen every two years, due to the architecture of a computer processor.

### 4.1.1 The Computer System

In order to describe how a computer processor develops, it is important to understand the jargon used in this area, as well as the basic structure of an integrated computer processor, as well as the subsystem that supports it. Figure 4.1 shows a simplified diagram of a computer subsystem, where the central processing unit is linked to all the vital areas of the system, including the random access memory (a fast store of data ready for computation), the 'chipset' (the system for directing traffic flow between all areas except the processor and memory), a disk drive (for mass data storage), and graphical output.

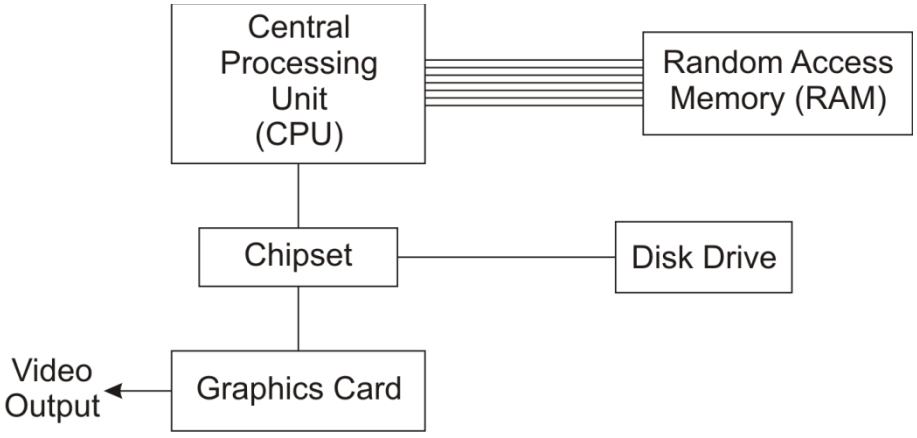



Figure 4.1 The computer subsystem supporting the central processor.

In classical computer simulation, the solution to mathematical operations occurs on the processor. The inputs to a mathematical operation will come from the memory, and likewise the result is also stored in memory (possibly overwriting one of the inputs,

depending on the programming code). If a result is wanted to be saved for use at a later date, after power is lost (for example, the computer is switched off), data can be moved to the disk drive. A simulator will often want to preview the results, and thus the graphics card handles video output.

### 4.1.2 The Computer Processor

In basic terms, the processor is produced through photolithography on electronic grade silicon and the central processor is more complex than a simple 'calculator' – internally the structure combines logic and internal memory storage to process the majority of data that is involved throughout the computer. Figure 4.2 shows the internal structure an Intel Sandy Bridge processor (released 2011), which essentially contains four areas to process calculations (cores), an internal shared memory (L3 cache), a memory controller (for communicating with the RAM, as in figure 4.1), and the addition of an integrated graphics system, which is becoming a recent feature on mainstream processors.

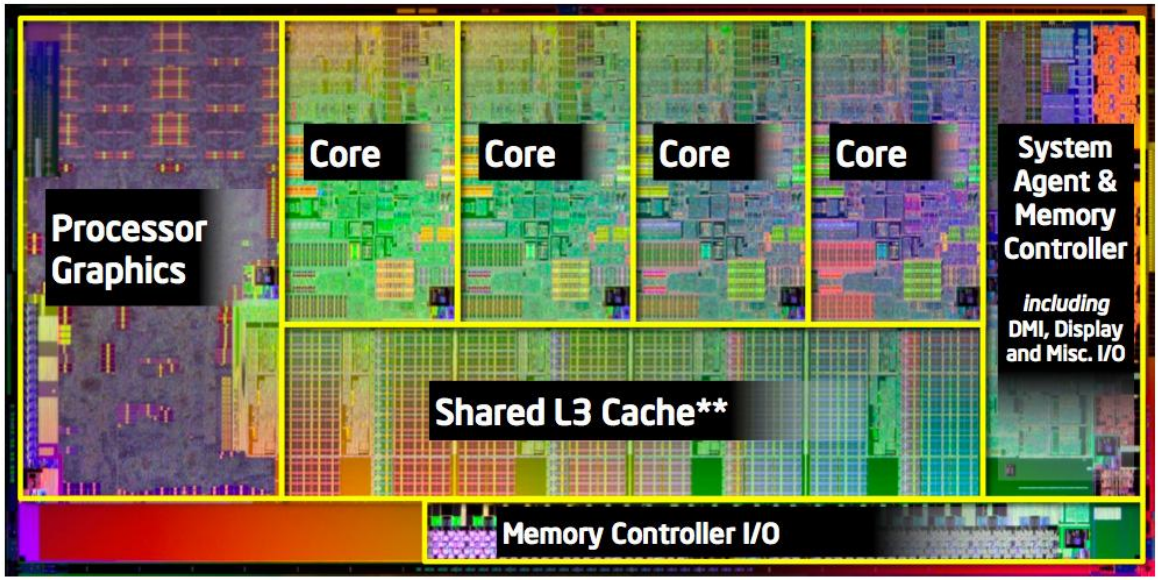



Figure 4.2 Internal structure of a modern computer processor.
Used with permission from [6]

The processor will run at a speed known as the 'clock' rate, measured in MHz. Typical values of the standard clock rate have changed over the past decades (1990: 16 MHz, 1995: 250 MHz, 2000: 1000 MHz, 2005: 3000 MHz); however this in turn increases the power consumption of the processor. In order to keep power consumption at a reasonable consumer and enterprise level, recent improvements have been made in architecture optimisation and reducing the minimum distance between relative transistors on the silicon surface (known as a process). By shrinking the process, from 65nm (2006) to 45nm (2007) and 32 nm (2010), each transistor on the silicon will require less energy to run as it only has to move electrons a shorter distance for the same throughput compared to previous processes, and thus more transistors can be added to keep the processor size the same. The biggest processor manufacturer, Intel, currently run on a 'tick-tock' model[7], whereby every year the technology will develop by a 'tick' (new design processes reducing the minimum distance between relative transistors) or a 'tock' (new processor architecture to optimise power usage and throughput), which has been successfully implemented over the past six years.

The biggest benefit of moving to a smaller process is the increase in transistor count. Recent developments in this area have given the central processor an integrated graphics processor, but it can also afford an increase in the number of 'cores' in a processor. Each core of a processor is designed to compute a certain number of operations per clock, depending on the complexity of the operation. The processor will not automatically use these additional cores in a simulation without a direct command from the software being produced. As is often the case, code passed down from previous simulationists are written in computer languages that do not support multi-core systems (because multi-

core systems were not invented then), or are large code databases which work, albeit slowly, and the research impetus is more on results, rather than the speed of results. The situation of multi-core systems is discussed later in this chapter.

### 4.1.3 The Memory Subsystem

The development of a single processor core is arguably the most important aspect of processor design, in terms of performance, power usage and throughput. Figure 4.3 shows a simplified diagram of the memory layout of a computer core. The processing part of a core, labelled P, has access to a limited number of registers (R, often 32), which each provide a quick store of values in a calculation. If the value needed is not found in the registers, the processor will look at its own level 1 cache (L1 cache), which is almost as quick to access as registers. If the value required is not here, the processor will access its level 2 cache, then the level 3 cache, which is shared between cores (note, modern processors have an L3 cache, but older ones do not – in the future, an L4 cache may evolve), or ultimately search for the value in main memory. Part of the logic of the processor will be to predict which memory values are required and to populate the caches in advance of computation. In essence, the caches are essentially random access memory, but due to their close proximity to the processing part of the core, are a lot quicker to access. They are also more expensive to produce, and require a lot of processor space, and thus increase the power requirements of the processor – thus a processor architect often has a balancing act with cache sizes and access times. Cache memory is often automatically controlled by the processor such that a simulator does not need to worry about it, unless the algorithm being used requires frequent and relatively

slow accesses to main memory such that the memory access speed becomes the limiting factor in the simulation.

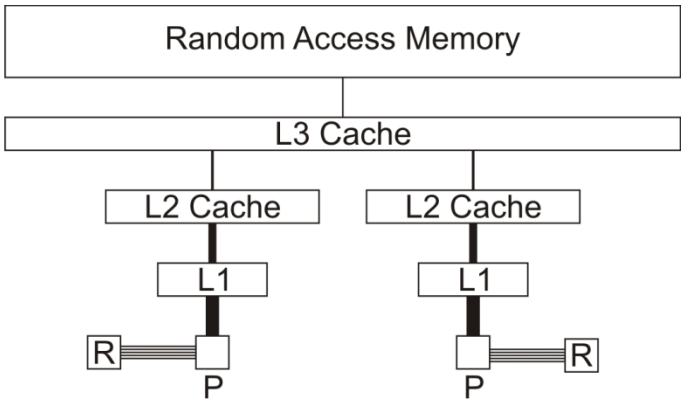



Figure 4.3 The memory subsystem of a dual core processor.

### 4.1.4 Processor Logic

Within a core is a system of logic, designed to break down any incoming operation into component basic operations (addition, multiplication), such as solving a sine function with a Taylor's series, using Gregory's series for an arctan function, or even operations which seem trivial to humans, such as division, require the Newton-Raphson method computationally[8]. There is often a wealth of functions available to the simulator, which takes into account speed of computation as well as accuracy. Part of logic improvements to a processor also include instruction sets, which are groups of operations which can be processed by specialised hardware. Again, these additional logic operations available to the simulator are not automatically used – they have to be directly accessed by the computer code itself.

For a simulator to take advantage of these novel improvements in logic, it is important to keep abreast of literature published by processor manufacturers and decide if it is

relevant to implement in their algorithms in order to upgrade to the relevant hardware. The simulator must also take note of the compiler, the computer program which translates code into machine readable language. The compiler may often have automatic optimisations available for speed, size, or how it treats trigonometric functions, amongst many others, which the simulator should be aware of, for accuracy and speed.

### 4.2 Parallelism

One of the simplest ways of utilising these new features of the processor is to take advantage of the multiple cores available. A modern (2011) computer processor in a home computer is likely to have two to four cores, and individual laboratory workstations can have up to sixteen cores each fairly easily. There are also possibilities to link together many machines to act as a single large multi-core (>128 cores) machine, known as a cluster. There are other features that a computer processor may use to increase throughput – one such example is Intel's 'Hyperthreading', which allows a core to act like two cores by managing the workload. Hyperthreading means that a quad-core processor can act like a quad-core in a worst case scenario, or like an octo-core under an ideal workload. To distinguish between the two, the processor may be said to have four cores, but eight 'threads', where each thread is capable of organising a separate computational operation. (Multi-threaded processors are quickly coming into normal usage, as the technology affords up to double the output with a little increase in power consumption.)

The easiest way to implement parallelism is when a series of simulations are needed for a result, for example a range of scan rates in voltammetry. Each simulation can effectively use one thread, and thus on a computer capable of four threads, four simulations can be run at the same time. This means that apart from code optimisation, the only way to

increase the time taken for a simulation is through pure processor speed. In order to speed up the simulation itself, the code must be optimised for multi-threading (the act of using more than one thread), which is highly dependent on the algorithm used. A branch of computer science is devoted to transforming seemingly serial algorithms to equivalent parallel ones with as little increase in the computational complexity as possible.

### 4.2.1 Serial Algorithms

The easiest way to describe serial and parallel code is with an example. Take, for example, the following pseudo-code, where `x`, `a`, `b` and `c` are variables:

```
(1)   x = 3.1415926535
(2)   a = sin(x)
(3)   b = a + 1
(4)   c = a + b
```

Line 1, where a value is assigned to $x$, is independent of the other lines of code. However, line 2 requires line 1 to be processed first, likewise line 3 requires line 2, and line 4 requires line 3. This is a perfect example of serial code, where the compiler must specifically order these operations for the core to calculate. In actual fact, depending on the compiler used, it may decide to optimise the code. By combining several operations, the speed or memory usage of the program may decrease, depending on if variables are needed later in the code – if `x`, `a` and `b` are not needed, the compiler may reduce this code to the following, which optimises it for memory usage:

```
(1)   c = 2*sin(3.1415926535) + 1
```

Many modern processors can now group a multiplication and an addition into one operation, making the original code reduced into a two step (sine, then multiplication/addition combination) process.

### 4.2.2 Parallelisation of Serial Algorithms

In a similar circumstance, a simple summation of a series of variables can be either serially added, or parallelised. For example, the following summation of the variables `a` to `h` can be calculated such that:

```
(1)   x[0] = a + b
(2)   x[1] = c + d
(3)   x[2] = e + f
(4)   x[3] = g + h
(5)   x[0] = x[0] + x[1]
(6)   x[2] = x[2] + x[3]
(7)   y = x[0] + x[2]
```

In this algorithm, lines 1 to 4 can be computed in any order and are thus parallel. Similarly, lines 5 and 6 are also parallel of each other, but are serial of lines 1 to 4, and must wait for those values to be computed. This way of summation (known as a 'scan addition') reduces what would be seven computational steps on a single thread processor to a three computational step process on any processor capable of four or more threads (the scan addition method also reduces the overall error in the final value). Scan additions are often over several thousand values (such as in large matrix manipulation), and parallelism reduces the number of computation steps appropriately.

### 4.2.3 Parallel Algorithms

In Chapter 3.1, the finite difference method in one dimension using a regular spatial grid was introduced, as the solution to a partial differential equation using centralised point differences. Equation 4.1 shows the solution to Fick's first law for this system:

$$\frac{[C]_x^{t+1}-[C]_x^t}{\Delta t} = D\left(\frac{[C]_{x+1}^t-2[C]_x^t+[C]_{x-1}^t}{(\Delta x)^2}\right) \quad [4.1]$$

The explicit method of the finite difference algorithm takes the concentrations of the current time step, and uses them to find the concentration on the next time step, such that the concentrations at $t$ are used to solve for $t+1$:

$$[C]_x^{t+1} = \frac{D\Delta t}{(\Delta x)^2}([C]_{x+1}^t - 2[C]_x^t + [C]_{x-1}^t) + [C]_x^t \quad [4.2]$$

Over the concentration grid of a finite difference simulation, the solution to each node in the grid is wholly independent of the solution to all the other nodes. What the simulator can do is write this algorithm to solve for as many nodes as there are threads per computational step, and as adjacent nodes use the same input data ($[C]_{x+1}^t, [C]_x^t$ and $[C]_{x-1}^t$), utilise data storage techniques to minimise memory access.

The explicit finite difference method has only been studied in detail for simple systems and in one dimension, due to the relative time to solve such systems in multiple dimensions compared to the implicit finite difference method. Some of this thesis, particularly Chapter 5, is dedicated to exploring explicit finite difference methods in multidimensional situations, using novel simulation techniques explained below.

It should be noted that the ratio of parallel threads to serial steps indicates a theoretical maximum speed up of simulation, as modelled by Amdahl's Law [9], which states that "if

$P$ is the proportion of a program that can be made parallel, and $(1 - P)$ is the proportion that cannot be parallelized, then the maximum speed-up that can be achieved by using $N$ processors is $\frac{1}{(1-P)+\frac{P}{N}}$".

### 4.3 Graphics Cards

So far, this thesis has discussed in terms of the computer processor, as this has been the de facto method of writing computer code to solve simulation problems since the beginning of simulation techniques. The computer processor has acted as the logic and processing unit at the heart of a computer, and is thus relatively easy to utilise advanced software techniques to exploit these resources.

However, there is an area of computer science which arguably bests computer processors in terms of both the speed of development and throughput. Graphics cards and graphic processing units (GPU) were originally designed to output information on a computer monitor. As computer games became popular across the 1990s, there was a consumer interest to see graphics card manufacturers develop more powerful graphics cards in order to render the latest video games with all the visual extras, which became more and more computationally expensive.

Since 2000, using the FLOPS metric (FLoating point OPerations per Second), in full parallel utilisation, the high end computer processors have increased from 6 Giga FLOPs (Pentium 4 3.2 GHz) in 2004 to 130 Giga FLOPS in 2011 (Core i7 2600K) whereas the high end graphics cards have increased from 10 GFLOPS in 2003 (NV30 architecture) to 1581 GFLOPS (GF110 architecture, GTX 580). This is shown in Figure 4.4, using a logarithmic FLOPs scale.

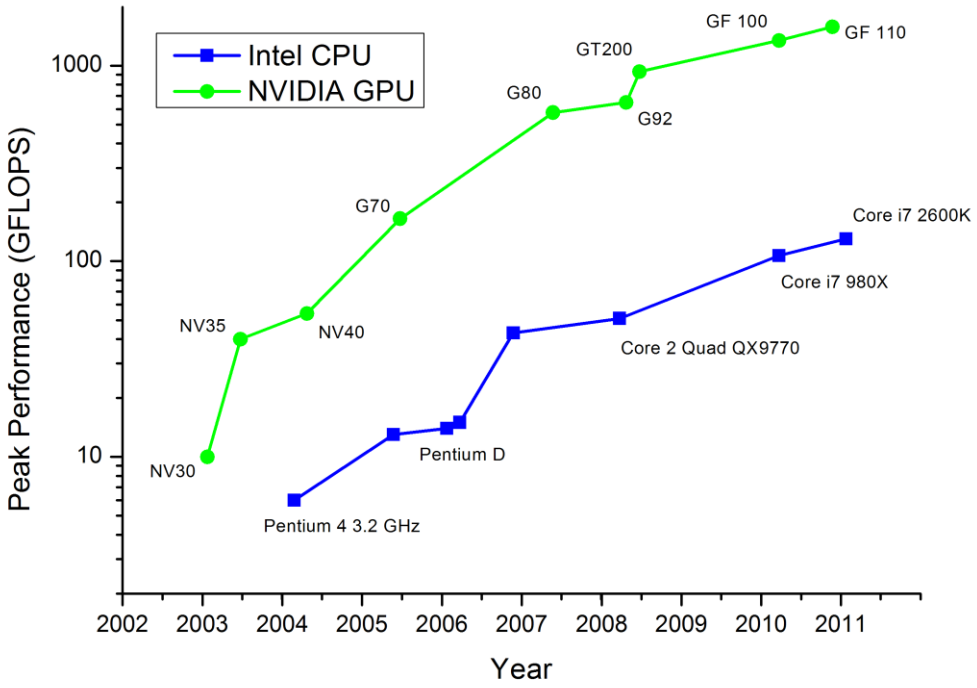



Figure 4.4 Comparison of pure computational throughput of Intel CPUs over NVIDIA GPUs between 2003 and 2011.

A graphics card is able to achieve a high level of throughput due to the architectural design concept used. A lot of the computational throughput in video games is the vector calculations involved for visual representations, as well as any game physics used. These computations are well defined, and often very simple in nature, covering only a few basic operations. Compared to a computer processor, a graphics card is a series of hundreds of computational units with almost zero logic or memory cache, able to take a series of data and apply the same operation to a set of values at once (for example, the movement of a car in a video game is essentially a transform of the polygons that make the object within a physical space).

### 4.3.1 Manufacturers and Programming Languages

As of 2011, there are currently two major graphics card manufacturers – NVIDIA and AMD. Both of these have adapted their gaming graphic products to be more easily used for multi-parallel programming and simulation. However there is no standard way of implementing a vector solver in terms of hardware – both of these companies have taken different approaches in their design and architecture, and thus for the processing method and for each company is somewhat different. Each company has released their own individual software and programming languages for simulators to create the software they need, however there are efforts to produce a programming language which covers both systems (known as OpenCL), but this is still in its infancy. As a result, each manufacturers products are different in terms of the same throughput of similar algorithmic problems – for example, AMD products are superior at cryptographic hashing [10] (the reduction of a complex series of data into an identifier with a 1 in $2^{256}$ (1 in $\sim 10^{17}$) chance of two sets of data having the same identifier) due to their ability to coordinate 3 of the functions into one operation, compared to the NVIDIA architecture which has to process the three functions individually.

One area a simulator must consider is the ease of which this hardware can be programmed. NVIDIA have had their own language for the past three years, and is the most developed. As a result, it is arguably the most successful as well. Their system is known as CUDA (Compute Unified Device Architecture), and is constantly evolving, resulting in a new major release of their standards every six to twelve months, as NVIDIA are strongly moving into the high performance computing markets [11].

The CUDA system also uses a dual language level system, described as high and low level languages. A low level language requires greater knowledge of the underlying hardware in order to manipulate computational data, as well as the time to write code for it, but affords the optimum performance when working well. A high level language provides the simulator with a large set of predefined commands to process the majority of the menial tasks (such as memory copy) which require large inputs in a low level language – there are also distinct compiler differences. But a high level language is arguably easier to understand, quicker to learn, and quicker to write code for.

Most of the important software in use today is written in a low level language for the reason of speed, and the ability of those companies to fund the time into making the products. AMD (and the OpenCL standard to cover both products) currently only offer a low level language, requiring in depth knowledge of the system at hand. As a result, all the work in this thesis is written in the NVIDIA CUDA high level language, and all subsequent discussion regarding GPU architecture and processing is derived from the NVIDIA product perspective.

### 4.3.2 GPU Architecture

The design concept of the graphics processor is substantially different to that of the computer processor. As the operations a graphics processor are simple mathematical functions, there is no need for a large logic structure per core in order to decipher the best route of processing. The result is a large number of lightweight cores suitable for vector calculations.

Figure 4.5 shows the basic layout of the graphics card, which combines a graphics processor with a large amount of on-card memory, and allows for chipset and computer processor communication. For comparison, the current (2011) maximum data rate with which the graphics card can communicate with the chipset and processor is 5 Gbps (gigabits per second), and the graphics processor can communicate with the graphics memory at speeds up to 192 Gbps (under optimal algorithm conditions) [12].

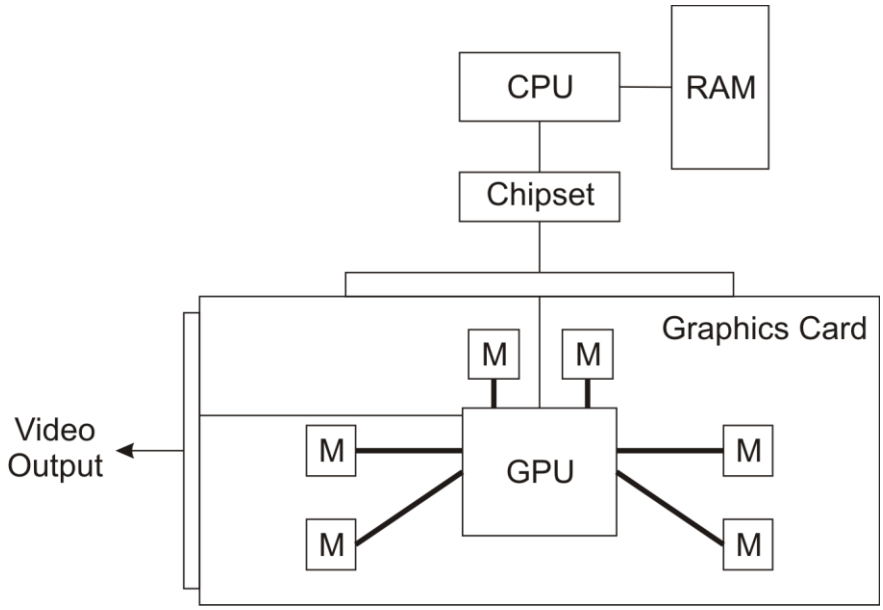



Figure 4.5 – A representation of a graphics card with six memory (M) chips, and the surrounding system.

#### 4.3.2.1 Streaming Multiprocessors

The graphics processor of an NVIDIA graphics card is split into Streaming Multiprocessors (SM), each of which contains 32 cores for processing data and has access to the 'global' graphics memory. Each of these cores will call for data from memory, process it, and either retain it for a future function (as determined by the code), or write the result back to the memory. Data and operations on a GPU are processed in batches, known as 'warps', of 32 threads. The GPU will take 32 different sets of data, perform the same

operation (the fact that it is the same operation is key to making a graphics card work), and then do other actions with the result. Each core would also have access to registers – a set of quick access memory designed for storing values. In a GPU core, there are 16384 or 32768 registers (depends on model of NVIDIA graphics card) that can each hold an integer or a float (single precision) variable (4 bytes).

#### 4.3.2.2 Warps

Take the following example of code, where `x` and `y` are variables in the registers, and `a` and `b` are arrays in memory:

```
(1)   x = sin(a[t])
(2)   y = cos(b[t])
```

Within a simulation involving 2048 nodes (and thus 2048 threads per time step), there will be 64 warps of 32 threads. Warp 1 threads would process line 1 of this code, which requires a memory load to get the value of `a[t]`. A memory read such as this takes a relatively long time (see below for an explanation), so the core will put the variable `x` into registers for the entire warp, with each core using one of its registers. The next warp (warp 2) will come in, and request the same read process. This will also take some time, so that warp is also placed in registers. If the first load request is finished, warp 1 will be engaged to perform the sine function, and then be placed back into registers until the value of `b[t]` is loaded. If the first load request is not finished, warp 3 would be engaged. Warps would continue to be engaged until either one of the warps receives the values it needs, or the registers run out of space, or the hardware limitation of waiting warps has been reached. If the registers run out of space or the warp limit is reached, the streaming processor would essentially be idle, not computing anything, until the

memory requested for one of the warps has returned. As a simple rule, a memory read in this situation takes 400 to 600 times longer than the addition, so the onus is on the simulator to write code which 'covers up' these memory reads with additional useful computation, or to minimise memory reads (and writes) altogether.

#### 4.3.2.3 Memory Cache and Latency

The graphics processing unit of the graphics card has evolved significantly, even since 2009 to 2011. Previous design philosophy was with the lack of a specific memory cache between the cores, resulting in large latency (the time between two events) for memory reads and writes. However, with the latest Fermi architecture from NVIDIA, small amounts of cache have been added, in the form of level 1 and level 2 caches, as shown in figure 5.6.

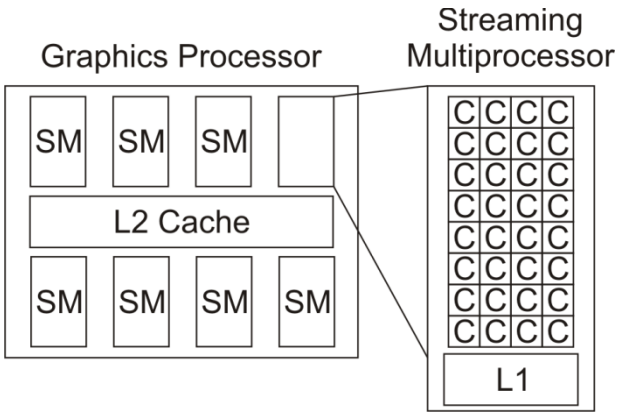



Figure 4.6 – Simplified internal structure of a Fermi graphics processor, showing L1 and L2 cache levels as well as 32 cores (C) per streaming multiprocessor.

These caches offer levels of memory which are faster to access than main memory (as with a CPU), and are managed by the compiler and thus require no user interaction. A small amount of L1 cache can be reserved for user specific inter-warp calculations, providing a quick access to this data. A comparison of time to access these caches is

found in figure 4.7. As these are still not ideal (ideal would be instant), in order to achieve the best throughput efficiency, the algorithm should be coded as to minimise memory accesses and maximise computation. As a result, a simulation may be memory bandwidth limited, rather than compute power limited, and the only solution to a quicker simulation is by code manipulation, rather than faster hardware (however, a new generation of hardware may afford faster memory accesses, speeding up simulation).

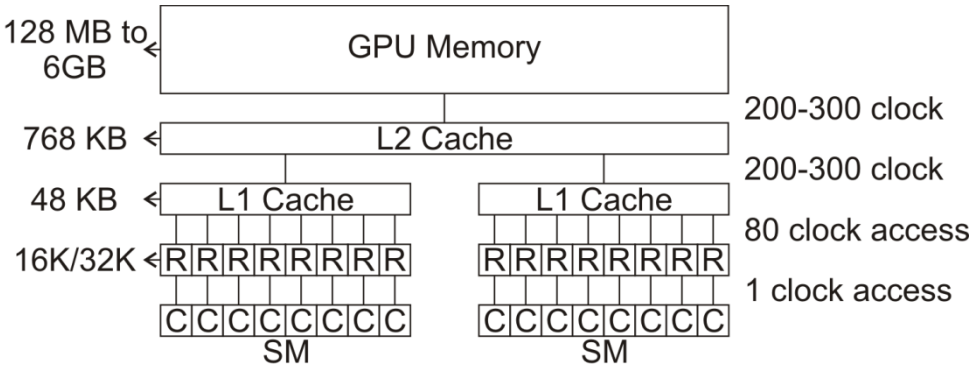



Figure 4.7 – Simplified schematic diagram showing two SMs with 8 cores each, with the sizes of the various caches and access times.

A section of the L1 cache can be set aside to be manipulated by the simulator, by providing a portion of memory to allow threads within a warp to communicate with each other (a useful feature in scan addition).

#### 4.3.2.4 Functions and Precision

As mentioned in Appendix A (Computational Digit Representation, and Reasons for Using Dimensionless Units), the concept of a single and a double precision number was introduced, explaining that while a double precision floating point number (i.e. one with a fractional element) affords many extra significant figures of precision, it comes at the expense of double the memory usage. Given the origins of graphics card development in

the gaming industry, calculating visual vectors, a graphics processor is built with single precision floating points in mind. This is due to the graphics in games do not require a sixth or seventh significant figure of accuracy in terms of their location or colour when dealing with up to 2560 by 1200 pixels on a monitor, or where repeated calculations for physics do not generate significantly different results with repeated calculation. As mentioned in Appendix A, this can be an issue for scientific simulation, where repeated calculation of values happens frequently and inherent errors in the way computer numbers are stored can lead to repetitive errors.

As a result, both of the main graphics card manufacturers develop two different sets of graphics card models. The cheaper models are engineered for gaming scenarios and have a severely hampered double precision application (e.g. double precision calculations are 1/8 of the speed of single precision calculations on the latest Fermi architecture [13]) – there are still applicable for simulation. The other models are for enterprise applications, feature full double precision performance, and other industry standards, such as ECC memory (the ability for memory to detect soft bit errors caused by natural radiation). The downside of these enterprise models is ultimately cost, and to a certain extent speed, as they are run at slower speeds to increase longevity.

Not all algorithms require double precision, but in terms of how the GPU processes functions is also an issue worth noting. In section 4.1.4, it was noted how for more complicated functions (sine, exponential), the hardware will actually use a series of multiplications and additions to calculate the value, and as a result these functions require a significantly longer time to process. In order to speed these calculations up, the architecture of the GPU and the computing languages often offer quicker, less accurate

functions for the simulator to use if the precision of the value is not important. For example, the two functions `sin()` and `sinf()` both compute the sine of a value, but `sinf()` will only calculate the value to enough significant figures to fit the value in a single precision floating point variable, and the `sin()` function will calculate at full double precision accuracy, using more memory and more compute time.

### 4.3.3 When to Use a Graphics Processor

A large part of choosing when to use a graphics processor is if the algorithm for the simulation being solved is applicable. The first requirement is that the simulation is either highly parallel, or that there are lots of very small serial simulations that fit into the GPU memory per thread. This chapter has explained extensively in terms of multi core technologies and parallel processing, as well as memory management and how the design of a graphics processor defines which algorithms are applicable. Table 4.1 gives a handy to use guide into the limitations and exploitable potential of a graphics card, in terms of peak performance and thread density (the amount of memory that can be held per thread).

**Table 4.1. Method Comparison**

| | Single CPU | Multi-core CPU | GPU |
|---|---|---|---|
| **Calculations per clock (threads)** | 1 | 2-16 | 40-1000+ |
| **Peak Performance** | 12.5 GFLOPs | 20-200 GFLOPs | 100 GFLOPs to 2.4 TFLOPs |
| **Maximum Memory** | Any | Any | 6 GB Per GPU<br>< 1 KB per thread |
| **Implementation** | Automatic | Software | Software + Hardware |
| **Thread Density** | Any | Low to High | Low |
| **Suitable Algorithms** | Serial and Parallel | Serial and Parallel | Parallel |
| **Software Implementation** | C/C++ | C/C++ with OpenMP | C and CUDA |

### 4.3.4 Financial Cost

When evaluating whether a graphics card is suitable for the planned simulation, a question of cost is normally relevant. As shown with the graphics sub-system in figure 4.1, a graphics card must have access to a computer processor and computer memory in order to function. However this computer processor does not need to be high powered – just sufficient to pass commands from the compiled code onto the graphics processor, and perform any subsequent post-processing of data.

The graphics cards available for sale often come in a variety of speeds and memory capacities as well, similar to computers and computer processors. Table 4.2 shows a representative sample of current prices of computers, computer processors and graphics card combinations suitable for an individual researcher. The prices involved do not include computer memory, which for enterprise level solutions (16GB or more) can equal, double or even triple the cost of the system – graphics card memory is fixed by the model of graphics card used. Graphics card names in bold represent enterprise models which have the additional features as listed above.

Depending on the computer supporting the graphics cards, a computer may house up to seven low powered graphics cards or four high powered graphics cards for simulation. In these circumstances, it is prevalent that they are sufficiently cooled to prevent overheating – especially with the consumer gaming models that are not designed to deal with the intense stress simulation can cause (computer games rarely cause the graphics card to work at 100% the whole time compared to simulation).

**Table 4.2 – Computer Hardware**

| CPU | CPU Threads | CPU Speed | Power Consumption | Cost (2011) |
|---|---|---|---|---|
| Lab machine | 1-2 | 1 GHz | 100W | £200 |
| Average Simulation | 4 | 2.4 GHz | 200W | £600 |
| Enterprise Simulation | 16 | 2.4 GHz | 600W | £1800 |
| **Graphics Card** | **GPU Threads** | **GPU Memory** | **Power Consumption** | **Cost (2011)** |
| GTX 460 | 336 | 1.0 GB | 150W | £100 |
| GTX 480 | 480 | 1.5 GB | 250W | £300 |
| GTX 580 | 512 | 3.0 GB | 300W | £500 |
| **Quadro FX 580** | 32 | 0.5 GB | 40W | £150 |
| **Tesla C1060** | 240 | 4.0 GB | 190W | £1000 |
| **Tesla C2050** | 448 | 3.0 GB | 240W | £2000 |
| **Quadro 6000** | 448 | 6.0 GB | 204W | £3600 |
| A reasonable GPU simulation setup (2011): | | | | |
| Average Simulation<br>2×GTX 480 | 4 Threads<br>960 Threads | 2.4 GHz<br>1.5 GB each | 700W | £1200 |

### 4.4 Applications of Graphics Cards in this Thesis

In Chapter 5, graphics cards are used to discuss and develop explicit solutions to the partial differential equations describing diffusional mass transport to the surface of an electrode. The algorithm development and manipulation of the equations is explicitly defined and initially applied to the contexts of elliptical / deformed microdisk electrodes. Relevant insights are also gained into limited length rectangular band electrodes, and dual microdisk systems in generator-collector mode (explained in detail in Chapter 5). These systems require a three dimensional approach when using a finite difference algorithm, due to a lack of axial symmetry of the electrode(s) used.

In Chapter 6, the relatively novel electrochemistry concept of the Random Walk methodology simulating individual particle movements is introduced and detailed. An

overview of the history of the electrochemical random walk is discussed, and a superior model is presented with comparison to experimental systems previously published. The random walk concept is also used to probe the concentration limits of cyclic voltammetry at ultra-low concentrations – where the movement of particles introduces a fluctuating response at the electrode. Finally, this thesis applies random walk to a recent experimental system whereby silver nanoparticles are interacting with an electrode, and during the interaction, a layer of thallium is deposited on the nanoparticle – this thesis will explain in depth the simulation to model the residence time of the nanoparticle required for monolayer coverage of different size nanoparticles.

**Bibliography**


[1] R.G. Compton, M.B.G. Pilkington, G.M. Stearn, Journal of the Chemical Society-Faraday Transactions I 84 (1988) 2155-2171.

[2] I.J. Cutress, R.G. Compton, Electroanalysis 21 (2009) 2617-2625.

[3] A. Molina, F. Martinez-Ortiz, E. Laborda, International Journal of Electrochemical Science 4 (2009) 1395-1406.

[4] G.E. Moore, Cramming more components onto integrated circuits, Electronics, 1965.

[5] Intel, Excerpts from A Conversation with Gordon Moore: Moore's Law, ftp://download.intel.com/museum/Moores_Law/Video-Transcripts/Excepts_A_Conversation_with_Gordon_Moore.pdf, 2005.

[6] A.L. Shimpi, The Sandy Bridge Review: Intel Core i7-2600K, i5-2500K and Core i3-2100 Tested, www.anandtech.com/show/4083, 2011.

[7] Intel, Intel's Tick-Tock Model, http://www.intel.com/technology/tick-tock, 2008.

[8] A. Habegger, A. Stahel, J. Goette, M. Jacomet, Electronic Design, Test and Application, 2010. DELTA '10. Fifth IEEE International Symposium on 1 (2010) 183-187.

[9] G. Amdahl, Validity of the single-processor approach to achieving large scale computing capabilities, AFIPS Conference Proceedings 1967, p. 483-485.

[10] Bitcoin.it, Why a GPU mines faster than a CPU, https://en.bitcoin.it/wiki/Why_a_GPU_mines_faster_than_a_CPU#Why_are_AMD_GPUs_faster_than_Nvidia_GPUs.3F, 2011.

[11] NVIDIA, High Performance Computing - Supercomputing with Tesla GPUs, http://www.nvidia.com/object/tesla_computing_solutions.html, 2011.

[12] NVIDIA, GeForce GTX 580, http://www.nvidia.com/object/product-geforce-gtx-580-us.html, 2011.

[13] NVIDIA, NVIDIA's Next Generation CUDA Compute Architecture: Fermi, http://www.nvidia.com/content/PDF/fermi_white_papers/NVIDIA_Fermi_Compute_Architecture_Whitepaper.pdf, 2009.

## Chapter 5 – Explicit Finite Difference for Three Dimensional Systems

From solving the mass transport equations represented by the partial differential equations in Chapter 3, the two dimensional algorithm required to implement the implicit finite difference problem has been studied extensively. However, when a system is three dimensional, implementation of an alternating direction implicit algorithm often brings instability depending on whether one or directions are considered implicit on each time step, as well as considerable simulation time to solve the one or two dimensional linear equations resulting from these algorithms [1-4]. The explicit algorithm is uncommonly used in electrochemical simulation [5-8], often due to time taken to simulate. However the explicit finite difference algorithm has well defined stability conditions, and affords a well defined parallel system in order to reduce computational simulation time when used in conjunction with massively parallel hardware, such as graphics cards and graphics processors as described in Chapter 4.

In this chapter, the explicit finite difference algorithm is detailed and applied to three electrode geometries which produce three-dimensional systems to be solved computationally.

The first of these is the case of elliptical microdisk electrodes, which are in effect deformed microdisks, where the electrode has one short diameter and one long diameter. The effect of the deformity plays a role in the microdisk edge length to microdisk area ratio, causing a greater edge effect and larger concentration gradients due to the large deformity. In this section of the chapter, different ratios are probed under potential step chronoamperometry and the results discussed in relation to the

implication to chronoamperometric results of microdisks which are commonly afforded to kinetic studies.

The second case examined in this chapter looks at square, rectangular and limited length microband electrodes, where the length to width ratio is not considered infinite. Previous simulation of microbands considers the infinite length to width ratio [9-13]; however the case of limited length microbands is an example of a three-dimensional simulation system and is encountered experimentally when examining electrode surfaces with scattered microband active regions.

The final case examined in this chapter deals with dual microdisk electrodes, when used in a collector-generator mode, such that the potential on one microelectrode is sufficient to drive a reversible redox reaction in one direction, and the other microelectrode is set a potential so drive the reverse reaction. The interaction between the diffusion layers is highly dependent on the inter-electrode distance, and this is examined via three dimensional explicit finite difference simulations.

**5.1 Constructing the Explicit Finite Difference Algorithm**

Chapter 3.1 detailed the finite difference approach to solving mass transport partial differential equations, by considering the concentration as the material balance gradient of the surrounding points, such that the concentration gradient of species $C$ at point $x$ in one dimension, where $[C]_x$ describes the concentration of $C$ at $x$:

$$\frac{\partial[C]}{\partial x} \approx \frac{[C]_{x+1}-[C]_{x-1}}{2\Delta x} \qquad [5.1]$$

The second derivative around point $x$ is determined by combining the half-differences from the adjacent half-points next to $x$:

$$\frac{\partial^2[C]}{\partial x^2} \approx \frac{\frac{\partial[C]}{\partial x}\Big|_{x+1/2} - \frac{\partial[C]}{\partial x}\Big|_{x-1/2}}{\Delta x} \approx \frac{\frac{[C]_{x+1}-[C]_x}{\Delta x} - \frac{[C]_x-[C]_{x-1}}{\Delta x}}{\Delta x} \approx \frac{[C]_{x+1}-2[C]_x+[C]_{x-1}}{(\Delta x)^2} \quad [5.2]$$

In this chapter, three dimensional systems are dealt with exclusively, whereby Fick's first law for the rate of diffusional mass transport is applied in three dimensions:

$$\frac{\partial[C]}{\partial t} \approx D_{\mathrm{C}}\nabla^2 = D_{\mathrm{C}}\left(\frac{\partial^2[C]}{\partial x^2} + \frac{\partial^2[C]}{\partial y^2} + \frac{\partial^2[C]}{\partial z^2}\right) \quad [5.3]$$

where $D$ is the diffusion coefficient of the species, and $t$ is the time. For the three-dimensional analyses used in this work, the Laplacian is split over the Cartesian dimensions $x$, $y$ and $z$.

### 5.1.1 Dimensional Transforms

Dimension transformations are often employed to electrochemical simulations, as explained in Appendix A and by many others [14,15], to relieve the simulation against scaling factors. A list of commonly used parameters in this chapter is provided in table 5.1, and subsequent transforms are given in table 5.2.

**Table 5.1 Common Parameters used in this Chapter**

| Parameter | Representation |
|---|---|
| Cartesian Coordinates | $x, y, z$ |
| Time ordinate | $t$ |
| Species concentration | $[C]$ |
| Major/Minor Axes of Ellipse | $a, b$ |
| Flux at Electrode Surface | $j$ |
| Diffusion Coefficient of Species C | $D_C$ |
| Initial Mesh Spacing | $\gamma$ |
| Electrode Radius/Length | $r_e$ |
| Mesh Expansion Coefficient | $h$ |

| Table 5.2 Parameter Transformations | |
|---|---|
| **Parameter** | **Transformation** |
| Cartesian Coordinates | $X = \frac{x}{r_e}, Y = \frac{y}{r_e}, Z = \frac{z}{r_e}$ |
| Time ordinate | $\tau = \frac{Dt}{{r_e}^2}$ |
| Species concentration | $C = \frac{[C]}{[C]_{bulk}}$ |

The expansion of equation 5.3 given the dimension transforms of table 5.2, give equation 5.4 to be solved.

$$\frac{\partial C}{\partial \tau} = \frac{\partial^2 C}{\partial X^2} + \frac{\partial^2 C}{\partial Y^2} + \frac{\partial^2 C}{\partial Z^2} \qquad [5.4]$$

Using similar substitutions to those previously utilised in chapter 3.4.6 regarding two-dimensional finite difference solutions, the parameter substitutions given in table 5.3, equation 5.4 expands explicitly by finite differences to give equation 5.5.

| Table 5.3 Parameter Substitutions | |
|---|---|
| **Parameter** | **Substitution** |
| $\lambda K, \quad K = X, Y, Z$ | $\frac{1}{\frac{1}{2}(\Delta K_+ + \Delta K_-)}$ |
| $\lambda K_\pm, \quad K = X, Y, Z$ | $\frac{\lambda K}{\Delta K_\pm}$ |

$$\frac{C^{t+1} - C^t}{\Delta \tau} = \lambda X_+ (C_{X+1}) + \lambda X_- (C_{X-1}) + \lambda Y_+ (C_{Y+1}) + \lambda Y_- (C_{Y-1}) + \lambda Z_+ (C_{Z+1}) +$$

$$\lambda Z_+ (C_{Z-1}) - C_{X,Y,Z} (\lambda X_+ + \lambda X_- + \lambda Y_+ + \lambda Y_- + \lambda Z_+ + \lambda Z_-) \qquad [5.5]$$

Equation 5.5 represents a series of concentrations that can be calculated independently from each other – each concentration can now be solved for each timestep ($t$) by the Explicit Algorithm for $t+1$ from $t$ on a GPU due to this level of independence. Note, $\Delta\tau$ is left out of the parameter substitutions in table 5.3 (unlike in chapter 3.4.6) due to the expanding time grid implemented in this chapter.

The explicit algorithm is uncommonly used in electrochemical simulation [5], often due to stability constraints and time taken to simulate. However, it offers complete parallelisation and low thread density – ideal for graphics cards – such that these issues should be overcome.

### 5.1.2 Explicit Algorithm Stability

The explicit algorithm is stable when, for equation 5.4, equation 5.6 holds.

$$\frac{\Delta\tau_{max}}{\Delta K_{min}{}^2} \leq \frac{1}{2} \ , \ K = X, Y, Z \qquad [5.6]$$

thus the upper bound on $\Delta\tau$ is fixed given the minimum grid spacing used.

There are variations of the explicit algorithm to improve this stability, such as the Dufort-Frankel [16] method of considering the concentration at each point as the linear function of time, such that:

$$C^t = \frac{1}{2}(C^{t+1} + C^{t-1}) \qquad [5.7]$$

However, this method requires knowledge of the concentrations of two steps before the current step, therefore doubling memory usage.

## 5.2 Elliptical Disk Electrodes

### 5.2.1. Application of Computational Theory

The majority of electrochemical experiments are simulated under approximation to reduce the computational complexity, often with mathematical simplification, with reliably accurate results [17,18]. However, for some specific applications, such as the simulation of odd-shaped two-dimensional microelectrodes rather than microdisk electrodes, previous approximations are no longer possible due to the three dimensional nature of the system. Electrochemical simulation underpins most current electrochemical literature describing rigorous physical investigations - the combination of current theory with experimental verification, or experimental results with simulation verification, is an essential technique with electrochemists. Seeking more than empirical progress, by adapting theory and utilising simulation, the electrochemist can predict a wide variety of results in common situations – coupled homogeneous and adsorption kinetics [5,19], hydrodynamic electrodes [20,21], weakly-supported electrolysis [22-24], modified electrodes [25,26], rough surfaces [27] electrodes of various geometry [18], and electrode arrays [28]. An electrochemist is seemingly only limited by his or her intuition into the mechanistic insights behind mass transport processes involving diffusion, convection and migration.

### 5.2.2 Microdisk Electrodes in Electrochemistry

Microelectrodes, i.e. electrodes with at least one dimension of the order of microns, have been used in voltammetric analysis since the 1980s. The benefit over planar macroelectrodes allows the exploration of significantly faster kinetic processes than was

previously thought possible. As mentioned previously in Chapter 1, microelectrodes exhibit this increase in sensitivity due to the convergent diffusion regime: at short periods of time, diffusion is effectively planar, and a Cottrellian $\left(1/\sqrt{t}\right)$ response is observed; at longer times, the decay of current is less rapid, such that the diffusion layer has spread beyond the edge of the electrode, and a steady state current is observed, consistent with convergent diffusion. The magnitude of the flux observed under steady-state conditions is proportional to the size and shape of the microelectrode - thus, the smaller the microelectrode, the larger the current density and the faster material can diffuse to (and from) the electrode surface. Chronoamperometry of such electrodes can lead to the solution of diffusion coefficients, species concentrations, electrons transferred and electrode radii. A thorough explanation can be found in reference. [19].

### 5.2.3 Ellipses and Deformed Microdisks

The equation of an ellipse is that of equation 5.8, where $a$ is the major axis, and $b$ is the minor axis. Equation 5.9 defines the eccentricity ($\varepsilon$) of an ellipse, and equation 5.10 shows the circumference (C) of an ellipse, as defined by the complete elliptic integral of the second kind. Equation 5.11 gives the area of an ellipse.

$$\frac{x^2}{a^2} + \frac{y^2}{b^2} = 1 \qquad [5.8]$$

$$\varepsilon = \sqrt{1 - \left(\frac{b}{a}\right)^2} \quad , \quad 0 < \varepsilon < 1 \qquad [5.9]$$

$$C = -2\pi a \sum_{n=0}^{\infty} \frac{\varepsilon^{2n}}{2n-1} \prod_{m=1}^{n} \left(\frac{2m-1}{2m}\right)^2 \qquad [5.10]$$

$$A = \pi ab \qquad [5.11]$$

In current literature, elliptical electrodes have been avoided experimentally and only briefly touched upon via simulation [29] or via analytical derivation [30]. However, elliptical electrodes, also described as deformed microdisks, are present in experimental results – for example in the paper by Whelan et al. [31], figure 4 shows typical electron micrographs of two 'typical' platinum microdisk electrodes of radius 13.06$\mu$m and 37.05$\mu$m. On closer inspection reveals two ellipses, with $a/b$ ratios of 1.085 and 1.034 respectively.

Ellipses are created, if not on purpose, by defects in the manufacturing process, or sub-standard cleaning, of microdisk electrodes. The experimental effect of such an electrode, on the microscale, is the increase in circumference to area ratio, introducing a larger edge effect into the system. This ultimately distorts the outcome of the voltammetry and the analysis if a regular microdisk is assumed - more so if the distortion effect is propagated through an array.

Simulation of a distorted microdisk has been briefly touched upon by Fulian et al. [29] in the presentation of their boundary element simulation technique. In reference [29], the current density at steady state was analysed as a function of the minor axis to provide figure 7 of that publication. The method used to obtain this data approximates what would be a classic finite-difference three-dimensional problem to a less computationally expensive two-dimensional type system. To the best of the authors' knowledge, no other simulation has been performed on elliptical electrodes under the monikers of 'elliptical electrodes' or 'distorted microdisks'.

### 5.2.4 Theory

For this work we have considered the transport limited one electron reduction:

$$A(aq) + e^{-} \rightleftharpoons B(aq) \qquad [5.12]$$

Under conditions where diffusion is dominant, and migratory effects can be ignored due to the presence of sufficient electrolyte present as defined by Dickinson et al. [24], solution of Fick's second law enables the prediction of the electrolysis current as a function of time:

$$\frac{\partial[A]}{\partial t} = D_{\mathrm{A}}\nabla^2[A], \qquad [5.13]$$

for typical voltammetric and chronoamperometric measurements, where $D$ is the diffusion coefficient of the species (A), and $t$ is the time. For the three-dimensional analyses used in this work, the Laplacian is split over the Cartesian dimensions $x$, $y$ and $z$.

**Table 5.4 Parameters used in this work**

| Parameter | Representation |
|---|---|
| Cartesian Coordinates | $x, y, z$ |
| Time ordinate | $t$ |
| Species concentration | $C = [A]$ |
| Major/Minor Axes of Ellipse | $a, b$ |
| Length Scaling Factor | $s_f$ |
| Flux at Electrode Surface | $j$ |
| Diffusion Coefficient of Species C | $D_C$ |
| Initial Mesh Spacing | $\gamma$ |
| Mesh Expansion Coefficient | $h$ |
| Normalised Circumference | $C_N$ |

Dimension transformations are often employed to electrochemical simulations, as represented by this author previously and by many others [14,15], to relieve the

simulation against scaling factors. A list of specifically used parameters in this section of this chapter is provided in Table 5.4, and subsequent transforms are given in Table 5.5.

**Table 5.5 Parameter Transformations**

| Parameter | Transformation |
|---|---|
| Cartesian Coordinates | $X = \frac{x}{s_f}, Y = \frac{y}{s_f}, Z = \frac{z}{s_f}$ |
| Time ordinate | $\tau = \frac{Dt}{{s_f}^2}$ |
| Species concentration | $c = \frac{C}{C_{bulk}}$ |
| Major/Minor Axes of Ellipse | $A = \frac{a}{s_f}, B = \frac{b}{s_f}$ |

#### 5.2.4.1 Mesh Generation

Simulation of an ellipse is reduced in complexity due to the axes of symmetry. As a result, one-quarter of the ellipse is simulated only, and appropriate boundary conditions (chapter 5.2.4.2) are applied. Mesh generation for this work combines the expanding mesh principles devised by Gavaghan [17] combined with those used by Streeter et al. [32] to deal with spherical electrodes. To mimic the curvature of an elliptical electrode, high density rectangular grids are used over the electrode surface in the $XY$ plane, and Gavaghan-style expanding grids thereafter. The $Z$-direction follows a directly expanding grid. Equations 5.14-5.16 explain the expansion, and figures 5.1-5.3 represent mesh points in each plane for an elliptical electrode with a b/a ratio of 0.5. Values of $\gamma = 0.01$ and $h = 1.125$ were used throughout this work. Mesh limits are set at $6\sqrt{\tau_{max}}$ from the electrode, which is known to sufficiently exceed the diffusion layer in all cases [17].

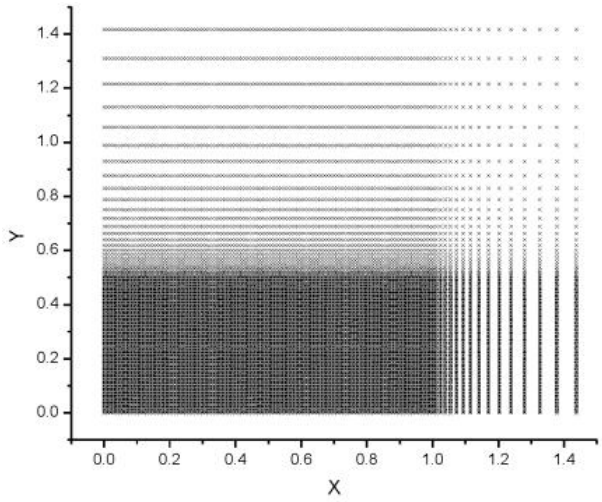



Figure 5.1 – Mesh Points in the $XY$ Plane

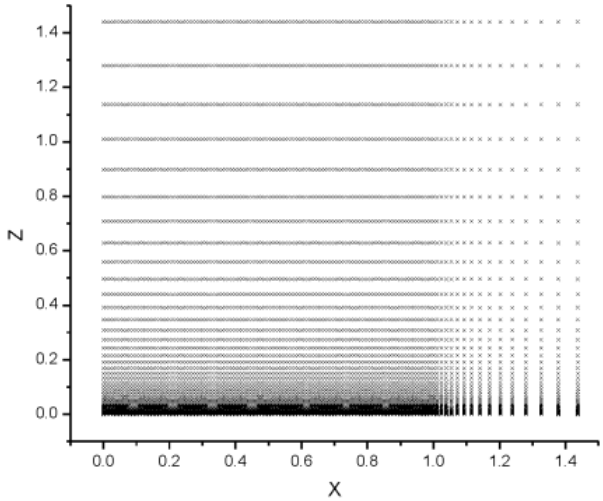



Figure 5.2 – Mesh Points in the $XZ$ Plane

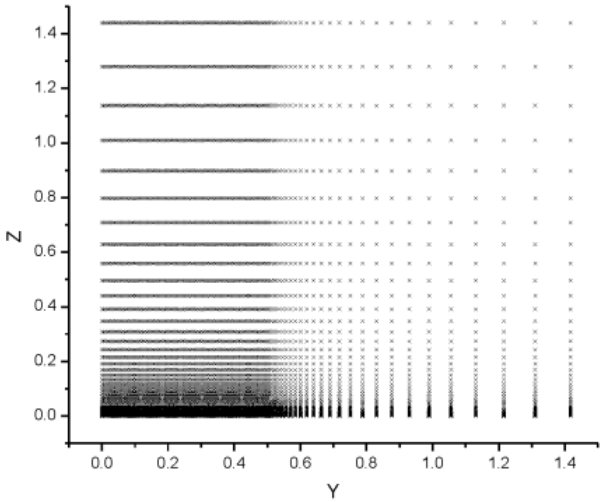



Figure 5.3 – Mesh Points in the $YZ$ Plane

$$\Delta X_i = \gamma; \qquad i < a$$
$$\Delta X_i = \gamma * h^{(i-\frac{a}{h})}; \qquad i > a \qquad [5.14]$$
$$\Delta Y_i = \gamma; \qquad i < b$$
$$\Delta Y_i = \gamma * h^{(i-\frac{b}{h})}; \qquad i > b \qquad [5.15]$$
$$\Delta Z_i = \frac{\gamma}{10} * h^i \qquad [5.16]$$

#### 5.2.4.2 Boundary Conditions

Three-dimensional simulation of an ellipse affords four spacial boundaries (electrode, insulator, symmetry, infinite) and two time boundaries. The boundaries of the $XY$ plane are shown in figure 5.4 (insulator, electrode), the $X = 0$ and $Y = 0$ planes give symmetry boundaries and the $X = X_{max}$, $Y = Y_{max}$ and $Z = Z_{max}$ give the infinite boundaries. At $\tau = 0, c_{X,Y,Z} = 1$, and at $\tau > 0, c_{Z=0} = 0$ at the electrode. These boundary conditions are also given in table 5.6.

**Table 5.6 Boundary Conditions**

| Boundary | Location | Expression |
|---|---|---|
| **Electrode** | Electrode Surface, $Z = 0$ | $\tau = 0, c_{X,Y,Z} = 1$<br>$\tau > 0, c_{Z=0} = 0$ |
| **Insulator** | Insulator, $Z = 0$ | $\frac{\partial c}{\partial Z} = 0$ |
| **Symmetry** | $X = 0, Y = 0$ | $\frac{\partial c}{\partial X} = 0, \frac{\partial c}{\partial Y} = 0$ |
| **Infinite** | $X = X_{max}$,<br>$Y = Y_{max}$,<br>$Z = Z_{max}$ | $c = 1$ |
| $\boldsymbol{\tau = 0}$ | | $c_{X,Y,Z} = 1$ |
| $\boldsymbol{\tau > 0}$ | | $c_{Z=0} = 0$ |

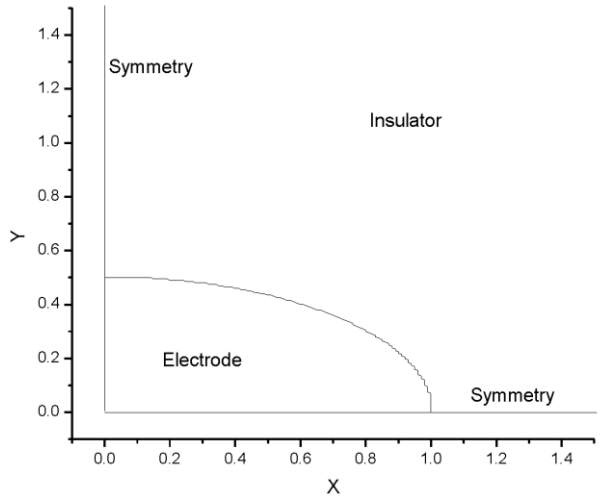



Figure 5.4 An $a$ = 1, $b$ = 0.5 ellipse shown in the $XY$ plane at $Z = 0$, representing boundary elements.

#### 5.2.4.3 Time Grid

For this work, the chronoamperometric response is calculated at an elliptical surface. An expanding time grid was implemented until $\Delta\tau_{max}$ (as defined in equation 5.6) was reached, and subsequent steps were $\Delta\tau_{max}$. The time grid shown in equations 5.17 to 5.20 was applied to all simulations in this paper.

$$\gamma_\tau = \frac{\gamma}{10} \qquad [5.17]$$

$$\Delta\tau_0 = \gamma_\tau{}^3 \qquad [5.18]$$

$$\Delta\tau_i = \tau_{i-1} * \sqrt{\gamma_\tau}, \qquad \tau_{i-1} * \sqrt{\gamma_\tau} < \Delta\tau_{max} \qquad [5.19]$$

$$\Delta\tau_i = \Delta\tau_{max}, \qquad \tau_{i-1} * \sqrt{\gamma_\tau} > \Delta\tau_{max} \qquad [5.20]$$

#### 5.2.4.4 Flux Calculation

The concentration flux to the electrode surface for time $t$ is given by equation 5.21.

$$j_\tau = \iint \left.\frac{\partial C}{\partial Z}\right|_{Z=0} dXdY \qquad [5.21]$$

Shown in equation 5.22, a two-point flux calculation is utilised to calculate the approximated derivative as being linear at the electrode surface, due to the increased level of discretisation compared to the various multipoint analyses used and compared by Gavaghan [33].

$$\left.\frac{\partial C}{\partial Z}\right|_{Z=0} = \frac{C_{Z=1} - C_{Z=0}}{\Delta Z_0} \qquad [5.22]$$

The double integral is then evaluated, as shown in equation 5.23.

$$\iint f(R)dXdY = \sum_{Y=0}^{Y_{el}} \sum_{X=0}^{X_{el}} \left[ f\left(R_{X,Y}\right) * \frac{1}{2}(\Delta X_+ + \Delta X_-)(\Delta Y_+ + \Delta Y_-) \right] \qquad [5.23]$$

#### 5.2.4.5 Multithreading and Graphics Cards

Explicit finite difference simulations, as described in chapter 5.1, were performed on two NVIDIA graphics cards – one low powered model commonly found in lab machines, and a mid-power model which is expected to become more mainstream (future models, as in reference. [34], promise many factors of speed increase). Comparative CPU simulations were performed in single CPU mode (which has been used previously for simulation, however not with the explicit algorithm) and multiple CPU mode (known as multithreading), using a machine with 16 available threads.

Graphics card implementation occurs through the CUDA development kit for NVIDIA GPUs, which is similar to the CPU language C, with a few minor code extensions. CPU code was written in C/C++.

### 5.2.6 Computer Utilisation

For CPU analyses, a dual Xeon E5520 workstation (maximum 16 threads) running at 2.4 GHz with 2.43GB RAM was used. GPU analyses were between an NVIDIA Quadro FX 580 (512MB RAM) on the dual Xeon workstation, and an nVidia GTX 280 (1GB RAM) on an AMD X2-5050e (2.5 GHz) with 3.24GB RAM. The NVIDIA Quadro has CUDA Compute 1.1, the NVIDIA GTX 280 has CUDA Compute 1.3. Simulations were compiled in Microsoft Visual C++ 2008, using CUDA Visual Studio Wizard (for GPU) and speed optimisations.

### 5.2.7. Results and Discussion

#### 5.2.7.1 Comparison vs. Microdisks

Analysis of a new method requires verification against a known standard. As the utilisation of the GPU is new in digital electrochemical simulation, initial verification of the method with the explicit algorithm was performed on an ellipse whereby $a$ = 1 and $_b$ = 1, i.e. a microdisk. For a microdisk, the Shoup-Szabo equation, as shown by equation 5.24, is known to give an accurate flux to within 0.6% for all $\tau$ [35].

$$j_\tau = 0.7854 + 0.4431\tau^{-\frac{1}{2}} + 0.2146e^{-0.39115\tau^{-\frac{1}{2}}} \qquad [5.24]$$

Figure 5.5 shows a plot of flux against $\tau$ for the simulation of a microdisk in three-dimensions. The flat line before $\tau = 10^{-5}$ is due to the maximum value of the flux in the simulation, which is limited by $\Delta Z$ – thus on the timescale $\tau < 10^{-5}$, diffusion is not measureable between $c_{Z=1}\ and\ c_{Z=0}$. As $\tau$ extends to long $\tau$, convergent diffusion due to

edge effects is observed. The simulation is within 1% for all $\tau > 10^{-5}$, often well within 1%.

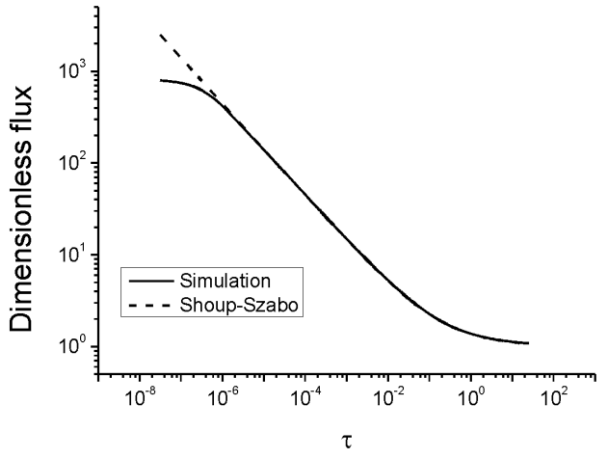



Figure 5.5. Simulation of a microdisk ($a$ = 1, $b$ = 1) against the Shoup-Szabo equation.

#### 5.2.7.2 Modifying the $a$:$b$ Ratio

Simulation in this work proceeded to simulate a series of elliptical microelectrodes for which the ratio $a$:$b$ changes, but where the area, as given in equation 5.11, is constant. Thus, a comparison with other microdisks of the same area can be made. Table 5.7 describes the results obtained for a series of $a$:$b$ ratios as a function of the flux at the surface of the ellipse divided by the flux expected at a microdisk of the same area as defined by equation 5.21 ($j_e/j_{SS}$) . The table shows the normalised circumference ($\boldsymbol{C_N}$), i.e. the circumference of the ellipse as defined in equation 5.10 divided by the circumference of a microdisk of the same area, and that as the electrode becomes more elliptical, the flux increases – a 1% deviation from the expected microdisk current from a true disk of the same area occurs at an a:b ratio of approximately 0.75. A plot of the flux transients, as seen in figure 5.6, shows that even a large degree of eccentricity in the

microelectrode, an $a{:}b$ ratio of 0.50, is seemingly identical by eye to a normal microelectrode of the same area.

**Table 5.7 $a{:}b$ Ratio Simulations**

| $a{:}b$ Ratio | $a$ | $b$ | $C_N$ | $j_e/j_{SS}$, $\tau = 1$ |
|---|---|---|---|---|
| **0.995** | 0.9974969 | 1.0025094 | 1.00001 | 1.003 |
| **0.990** | 0.9949874 | 1.0050378 | 1.00003 | 1.003 |
| **0.980** | 0.9899495 | 1.0101525 | 1.00010 | 1.003 |
| **0.970** | 0.9848858 | 1.0153462 | 1.00023 | 1.003 |
| **0.960** | 0.9797959 | 1.0206207 | 1.00042 | 1.003 |
| **0.950** | 0.9746794 | 1.0259784 | 1.00066 | 1.003 |
| **0.850** | 0.9219544 | 1.0846523 | 1.00660 | 1.005 |
| **0.750** | 0.8660254 | 1.1547005 | 1.02062 | 1.010 |
| **0.650** | 0.8062258 | 1.2403473 | 1.04605 | 1.018 |
| **0.550** | 0.7416198 | 1.3483997 | 1.08816 | 1.032 |
| **0.500** | 0.7071068 | 1.4142136 | 1.11803 | 1.041 |
| **0.450** | 0.6708204 | 1.4907120 | 1.15590 | 1.054 |
| **0.350** | 0.5916080 | 1.6903085 | 1.26625 | 1.090 |
| **0.250** | 0.5000000 | 2.0000000 | 1.45774 | 1.157 |
| **0.150** | 0.3872983 | 2.5819889 | 1.84617 | 1.293 |

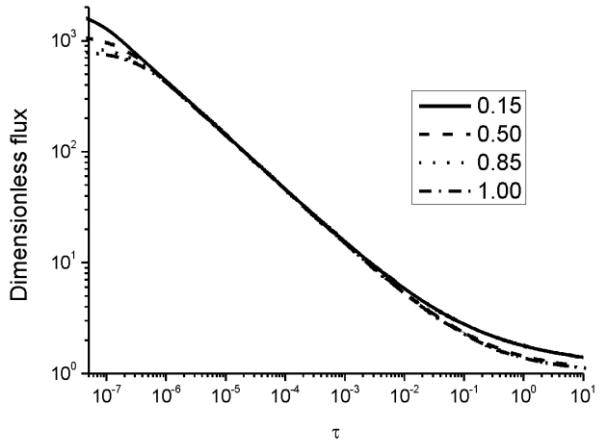



Figure 5.6 Flux transients of a:b ratios from 1.00 to 0.15.

A plot of % flux deviation against normalised circumference yields a directly proportional relationship. Given the simulation error as shown in chapter 5.2.7.1, the fitting of a linear equation of this line results in equation 5.25 with an $R^2$ fitting coefficient of 0.999. These

results confirm the increased edge effect of the ellipsicity of the electrodes, resulting in a greater proportion of convergent diffusion over a microdisk.

$$\%\ Flux\ Deviation\ at\ (\tau = 1) = 34.64\ (C_N - 1) \qquad [5.25]$$

These results afford two main points of discussion. We see from table 5.7 and figure 5.6 that a microelectrode with a moderate degree of elliptical nature (~0.75) will still act within 1% of a microdisk with a similar nature. However a problem arises with measuring electrode size by steady state studies – 1% in an acceptable error in an experiment, but could suggest, for example, that the microelectrode is of 10$\mu$m radius, or an ellipse with axes of 11.54$\mu$m and 8.66$\mu$m. An array of 10 x 10 of these elliptical electrodes, created under conditions of diffusional independence [36], each with a 1% difference to a microelectrode, may cause the experimentalist to believe that they have actually an array of microelectrodes of double the area.

#### 5.2.7.3 Comparing Simulation Times

For a simulation time comparison, an ellipse of $a$ = 1, $b$ = 1 and $\tau_{max}$ = 1 was used. All speed optimisations in the development environment were enabled. Table 5.8 shows the results between using a single-CPU simulation (S-CPU), a multi-threaded CPU simulation (M-CPU), a GPU simulation on a low-end graphics card (FX 580) and a GPU simulation on a mid-range consumer graphics card (GTX 280). The GTX 280 simulation offers a 26.93x speed-up over previous explicit simulation methods, and a 5.23x increase over CPU code utilising 16 threads.

| Table 5.8 Simulation Time Comparison | | | |
|---|---|---|---|
| **Method** | **Threads Capable** | **Time to Simulate to $\tau_{max}$ = 1 / s** | **Speed-up over old S-CPU method** |
| **S-CPU** | 1 | 107271 | 1.00x |
| **M-CPU** | 16 | 19862 | 5.40x |
| **FX 580** | 128 | 33012 | 3.25x |
| **GTX 280** | 960 | 3760 | 26.93x |

The actual flux calculations of the low-end FX 580 are of concern. The FX 580 model has support for single precision only, unlike the CPU methods and the GTX 280 which can use double precision. Single precision limits stored data to seven significant figures, whereas double precision stores data to 15 significant figures. Thus if concentration changes at a mesh point between time steps is less than the seventh significant figure, the FX 580 does not record it, and error is introduced into the simulation. Figure 5.7 shows a transient calculation, and at approximately $\tau$ = 1, the simulation fails to record further diffusion, thus nullifying the FX 580 results. Double precision, as found on the GTX 280 and CPU, does not have this issue. Double precision on the GPU is enabled for certain models at the manufacturers' discretion, often only on their mid-range to high-range products. It is also stated by nVidia that only a part of the GTX 280 performs double precision, specifically one-eighth, reducing the maximum FLOP potential, previously mentioned as 933 GFLOPS in section 1.1.1, to 116.6 GFLOPs [37]. This one-eighth figure is expected to be removed with new graphic card products [34]. However, a 5.23x speed-up was still observed for the GTX 280 over M-CPU.

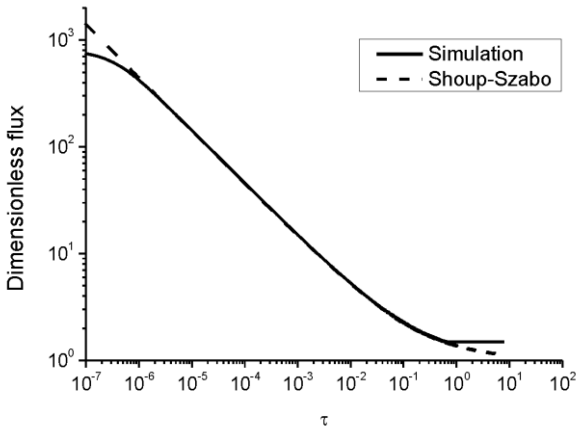



Figure 5.7 Flux transient on a Quadro FX 580

Also of note is the cost comparison between these methods. At the time of writing, a dual Xeon workstation offering 16-threads is approximately eight times the cost of a GTX 280 GPU. The GPU requires an appropriate computer to utilise its capabilities, but many GPUs may be fitted to one computer. Overall, a GPU such as the GTX 280 can offer an approximate 42x increase in performance/cost analysis over an M-CPU arrangement.

### 5.2.8 Conclusions

The three-dimensional explicit calculation of elliptical microelectrodes using GPGPU has been described, showing a new route in performing electrochemical simulation. The GPGPU method takes clear advantage of parallel algorithms to offer an efficient, powerful and cost-effective tool for the electrochemist. By exploiting graphics cards, new inroads into fundamental analysis via parallel algorithms, such as the explicit algorithm, can be made. This new method has shown the theoretical analysis of elliptical electrodes – the difficulty in direct calculation of electrode size from voltammetric analysis, and problems arising from analyses using these electrodes, have been discussed.

### 5.3 Square, Rectangular and Microband Electrodes

#### 5.3.1 Microband Electrodes

The use of microelectrodes within electrochemistry over the past decade has remodelled the landscape of voltammetric analysis [38-41]. Microelectrodes provide an enhanced rate of mass transport, over macroelectrodes, to give mechanistic information for the electrochemist studying fast electrochemical processes, previously too rapid to be probed by voltammetric analysis. This enhanced rate of mass transport reflects the types of diffusion to the various electrodes– for macroelectrodes, planar diffusion dominates; whereas for microelectrodes, where edge effects are a major contributor to the diffusion, convergent diffusion is observed. In addition, microelectrodes have a low capacitance, facilitating fast voltage sweep rates in voltammetry and the low currents drawn greatly reduces ohmic distortions compared to macroelectrodes.

Microband electrodes are of interest to electrochemists for their ease of fabrication to specific measurements [41,42], and application within channel systems [43-50]. Utilising an array of band electrodes, of alternate 'generator-collector-generator' modes utilising anodic and cathodic potentials, allows the 'redox cycling' of target analytes by measuring diffusion from the generator microbands to the collector microbands [51-56]. A pair of band electrodes solely for generator-collector experiments is also a common electroanalytical setup [54,57] for 'titration' measurements.

Fabrication methods for microbands have included screen printing processes [41,57], photolithographic techniques [58], and laminated microelectrodes [42]. A considerable problem with fabrication is to make the electrode flush with the insulator material – for example, the laminated microelectrode technique described by Welford et al. [42] are

observed to be slightly recessed but afford good agreement with the relevant analytical theory as well as being cheap and easy to produce, without special technical skills or services.

### 5.3.2 Microband Theory and Simulation

#### 5.3.2.1 History of Two Dimensional Simulation

Present microband theory relates to the situation where the band is substantially longer in one dimension than the other (by several orders of magnitude) such that the diffusional contribution at the ends of the long dimension is insignificant compared to the bulk of the band. This leads to a distinct two dimensional system, which is solved in terms of diffusion by evaluating equation 5.26.

$$\frac{\partial[C]}{\partial t} = D\left(\frac{\partial^2[C]}{\partial x^2} + \frac{\partial^2 C}{\partial z^2}\right) \qquad [5.26]$$

where $c$ is the concentration of the species of interest, $t$ is the time, D is the diffusion coefficient of species $c$, and $x$ and $z$ are the Cartesian co-ordinates of interest. Current theoretical literature on microbands is split in the use of dimensionless co-ordinates – some, such as Aoki et al. [59-62] and Streeter et al. [44], use equation 5.27, whereas Britz et al. [63] and Strutwolf [1] use equation 5.28. Ultimately equations 5.27 and 5.28 differ simply by a factor of 4 (as shown in equation 5.29), however this can still lead to potential confusion when interpreting the results – in this paper, to simplify the explanations, equation 5.27 will be used exclusively, and references using equation 5.28 will be reduced to use the dimensionless time in equation 5.27.

$$\tau = \frac{Dt}{w^2} \quad [5.27]$$

$$T = \frac{4Dt}{w^2} \quad [5.28]$$

$$4\tau = T \quad [5.29]$$

where $w$ is the width of the electrode in the $x$ direction.

Aoki et al. simulated equation 5.26 to develop two solutions for the microband – one at short times shown as equation 5.30 [59-61], and one at long times shown as equation 5.31 [60,62], which were subsequently conjoined into one expression shown in equation 5.32 [60]. These were confirmed by experiment on bands with length to width ratios of 20.4, 92.6, 160 and 375. This latter expression is reported by Aoki to be accurate to within 0.8% for all $\tau$, however Britz et al. [63] find this solution deviates from their own simulations at $\tau > 100$.

$$f(\tau) = \frac{1}{\sqrt{\pi\tau}} + 1 - \cdots \text{ for } \tau < 2 \text{ within } 5\% \quad [5.30]$$

$$f(\tau) = 2\pi\left(\frac{1}{ln\tau+3} - \frac{0.577}{(ln\tau+3)^2} - \frac{1.312}{(ln\tau+3)^3} - \cdots\right) \text{ for large } \tau \quad [5.31]$$

$$f(\tau) = \frac{1}{\sqrt{\pi\tau}} + 0.97 - 1.10exp^{\left(\frac{-9.90}{|\ln(12.37\tau)|}\right)} \text{ for } \tau < 10^8 \text{ within } 0.8\% \quad [5.32]$$

Coen et al. [64] provided comparison to the work of Aoki using integral equation methods, resulting in equation 5.33, agreeing with 1% for $7.5 < \tau < 25$.

$$f(\tau) = \frac{5.553}{\ln 4\tau} - \frac{6.791}{(\ln 4\tau)^2} \quad [5.33]$$

Szabo et al. [65] have also calculated a solution for all t in two parts – the first part, for $\tau <$ 0.4, is the short time solution similar to Aoki et al. in equation 5.27; the second, shown in equation 5.34, is reported as accurate within 1.3% for $\tau > 0.4$.

$$f(\tau) = \frac{\pi \exp\left(-0.4\sqrt{\pi\tau}\right)}{4\sqrt{\pi\tau}} + \frac{\pi}{\ln\left(e^{5/3} + \sqrt{64e^{-\gamma}\tau}\right)} \qquad [5.34]$$

where $\gamma = 0.577216$, the Euler constant.

Conformal mapping techniques utilised by Britz et al. [63] were used to compare the solutions found in equations 5.32-5.34 to find a general solution accurate to within 0.1% for all $\tau$. The conclusions of that paper result in suggesting the use of the Aoki short and long time approximations of equations 5.30 and 5.31, and linear interpolation of a set of values provided in that paper for $0.125 < \tau < 175$.

Alden et al. [66] use the Strongly Implicit method (SIP) alongside the Backwards Implicit method (BI) to simulate channel microbands with and without convection and discussed the merits of each method.

#### 5.3.2.2 Three Dimensional Simulation and Theory

Rectangular microbands have three main modes of diffusion – linear diffusion over the electrode, convergent edge diffusion at the long and the short edges, and point diffusion at the vertices of the electrode – this is shown in figure 5.8. Two dimensional simulations assume that the short edge diffusion and the point diffusion are insignificant compared to the long edge and linear diffusional modes – this is assumed when the length of the electrode is much greater than the width, or the electrode is flush against a channel wall.

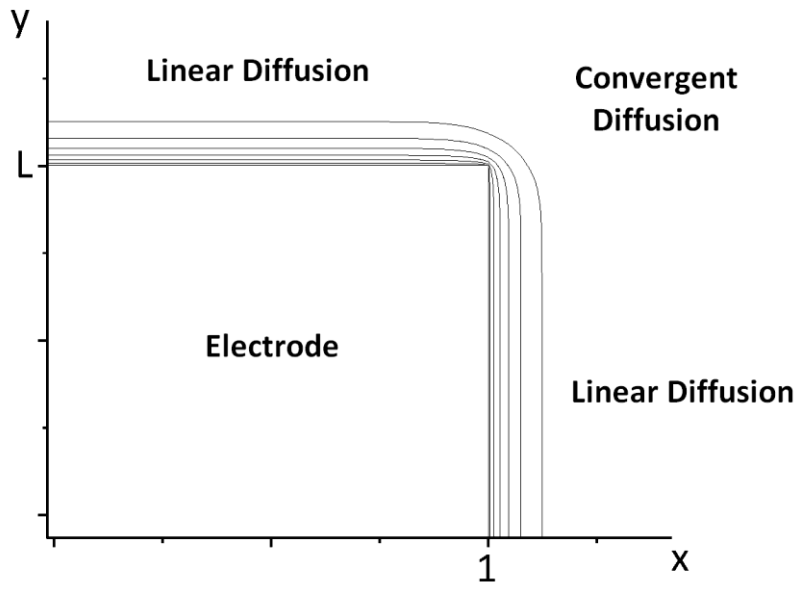



Figure 5.8 One-quarter representation, in the $XY$ plane, of a rectangular electrode showing the modes of diffusion.

Three-dimensional simulation assumes that the point and the side edge diffusion play a significant role in the voltammetry occurring at the electrode. Simulation of such a system is reduced in complexity due to the two axes of symmetry afforded by a rectangular electrode.

To this author's knowledge, only two works of three-dimensional analysis of rectangular (non pseudo-infinite) microband electrodes exist – that of Strutwolf [1], and of Woodvine et al. [67].

Strutwolf uses a finite difference simulation incorporating the Douglas-Gunn ADI (alternating direction implicit) algorithm to simulate one quarter of a rectangular electrode of different lengths. The conclusion of that work for the experimentalist is ultimately in figure 5 of ref [1], where Strutwolf has considered the deviations from the

Aoki equation for an infinite band which arise for bands of certain finite lengths and for particular times following a potential step.

Woodvine et al. use commercial finite element software to simulate one quarter of a square electrode, although it has recently been advised that this software, for three-dimensional electrochemical simulation, be approached with extreme caution [68]. The results of interest to this paper can be seen in figure 8 of ref [68], where the commercial software gives varying results depending on the mesh accuracy used with reference to the square electrode. Despite the distinct absence of the standard electrochemical procedure of assuming a semi-infinite boundary at a distance $6\sqrt{\tau_{max}}$ from the electrode surface, which has been extensively used successfully [14,69-74] and is known to sufficiently exceed the diffusion layer in all cases [17], comparison on various mesh parameterisations is also shown for both a square electrode and a microdisk. For the microdisk, an error of less than 5% is observed at high mesh parameters and less than 1% for very high mesh granularity (the exact figure is difficult to decipher from the paper and not provided in the text). However the goal of the work in [67] is the development and characterisation of square electrode systems, and especially arrays of these, rather than development of square electrode theory.

### 5.3.3 Theory and Simulation

This work considers the transport limited potential step chronoamperometry for a one electron reduction:

$$A(aq) + e^- \rightleftharpoons B(aq) \qquad [5.35]$$

When diffusion is solely dominant, and migratory effects are ignored due to the presence of sufficient electrolyte, quantified by Dickinson et al. [24], solution of Fick's second law enables the prediction of the electrolysis current as a function of time:

$$\frac{\partial[A]}{\partial t} = \mathrm{D_A}\nabla^2[A], \qquad [5.36]$$

The finite difference system used in this work is identical to that found in ref [75] which we have used previously to great effect. Parameters unique to this work are noted in the text.

The geometry of the simulated microband electrode is shown in figure 5.9, where length, $l$, is of the electrode size in the y direction, and width, $w$, is the electrode size in the x direction. The simulated quarter-electrode has dimensionless half-width $X = 1$, and dimensionless half-length $L = l/w$.

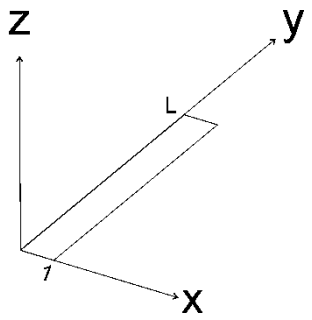



Figure 5.9 Geometry of the simulated system

The system will simulate three-dimensional potential step chronoamperometry at the electrode surface and diffusion within the semi-infinite boundary of length $6\sqrt{\tau_{max}}$ from the electrode by solving the dimensionless three-dimensional form of equation 5.36, as shown in equation 5.37. The finite difference form of equation 5.37 which is solvable computationally is shown in equation 5.38, utilising expanding mesh techniques as

described in section 5.3.3.1. Boundary conditions of the system for the potential step are provided in table 5.9.

$$\frac{\partial c}{\partial \tau} = \frac{\partial^2 c}{\partial X^2} + \frac{\partial^2 c}{\partial Y^2} + \frac{\partial^2 c}{\partial Z^2} \quad [5.37]$$

$$\frac{c^{t+1} - c^t}{\Delta \tau} = \frac{\frac{c_{X+1} - c_X}{\Delta X_+} - \frac{c_X - c_{X-1}}{\Delta X_-}}{\frac{1}{2}(\Delta X_+ + \Delta X_-)} + \frac{\frac{c_{Y+1} - c_Y}{\Delta Y_+} - \frac{c_Y - c_{Y-1}}{\Delta Y_-}}{\frac{1}{2}(\Delta Y_+ + \Delta Y_-)} + \frac{\frac{c_{Z+1} - c_Z}{\Delta Z_+} - \frac{c_Z - c_{Z-1}}{\Delta Z_-}}{\frac{1}{2}(\Delta Z_+ + \Delta Z_-)} \quad [5.38]$$

**Table 5.9 Boundary Conditions**

| Boundary | Location | Expression |
|---|---|---|
| **Electrode** | Electrode Surface, $Z = 0$ | $\tau = 0, C_{X,Y,Z} = 1$<br>$\tau > 0, C_{Z=0} = 0$ |
| **Insulator** | Insulator, $Z = 0$ | $\frac{\partial C}{\partial Z} = 0$ |
| **Symmetry** | $X = 0$, $Y = 0$ | $\frac{\partial C}{\partial X} = 0$, $\frac{\partial C}{\partial Y} = 0$ |
| **Infinite** | $X = X_{max}$, $Y = Y_{max}$, $Z = Z_{max}$ | $c = 1$ |
| $\boldsymbol{\tau = 0}$ | All $X, Y, Z$ | $c_{X,Y,Z} = 1$ |
| $\boldsymbol{\tau} > 0$ | Electrode Surface, $Z = 0$ | $c_{Z=0} = 0$ |

#### 5.3.3.1 Mesh Generation and Time Grid

Previous work [75] using the GPU method required the use of a constant grid due to the electrode geometry. The nature of the rectangular electrode system is optimal for directly expanding grids, focusing mesh points on the edges of the electrode. The expanding mesh principles used were introduced by Gavaghan [17]. As in [75], values of $\gamma = 0.01$ and $h = 1.125$ were used, and mesh limits are set at $6\sqrt{\tau_{max}}$ from the electrode, as discussed above. Figure 5.10 represents the mesh used in the $XY$ plane for a square electrode, showing the density of points at the electrode edge.

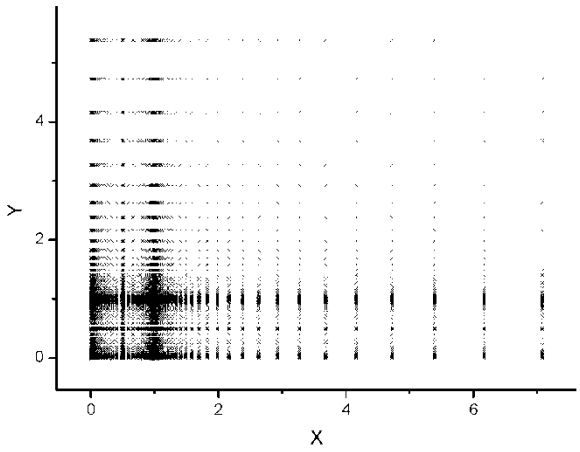



Figure 5.10 Mesh points in the $XY$ plane on a square electrode

The time grid used in this paper is identical to that validated in chapter 5.2.4.3.

#### 5.3.3.2 Flux Calculation

The normalised current at the electrode surface for time $\tau$ is given by the double integral of the flux gradient over the electrode surface, as shown in equation 5.39:

$$\frac{i(\tau)}{nFDc^*l} = f(\tau) = \frac{2}{L}\iint \frac{\partial c}{\partial Z}\Big|_{Z=0} dXdY \qquad [5.39]$$

The concentration flux at the electrode surface is given by equation 5.40:

$$j_\tau = \iint \frac{\partial c}{\partial Z}\Big|_{Z=0} dXdY \qquad [5.40]$$

A two-point flux calculation is utilised to calculate the approximated derivative as being linear at the electrode surface, as shown in equation 5.41.

$$\frac{\partial C}{\partial Z}\Big|_{Z=0} = \frac{C_{Z=\Delta Z_0} - C_{Z=0}}{\Delta Z_0} \qquad [5.41]$$

The double integral is then evaluated, shown in equation 5.42.

$$\iint g(R)dXdY = \sum_{Y=0}^{Y_{el}} \sum_{X=0}^{X_{el}} \left[ f\left(R_{X,Y}\right) * \frac{1}{4}(\Delta X_{+} + \Delta X_{-})(\Delta Y_{+} + \Delta Y_{-}) \right] \quad [5.42]$$

### 5.3.4 GPU improvements

From the previous GPU publication [75] (and Chapter 5.2), improvements to GPU approach of the explicit algorithm have been made to increase throughput. A point that was not elaborated on in [75] is the use of blocks – a GPU executes threads in batches, known as blocks. Block size is determined by the coder, and in this case is a grid of $X * Y$ threads. Within a block, each thread has access to a quick memory bank called shared memory - thus threads within a block can share values. In the previous sub-chapter, each element in the 3D mesh was a thread – due to the explicit algorithm, it would call from main memory each of the concentrations from the previous timestep it required (in this case, 7). As main memory accesses are slow, the algorithm was improved such that within a block, each thread would load one concentration into shared memory from main memory, then request the others required from shared memory. This reduces the action to ~1 main memory load per thread (the edges of the block require another element to load, thus a 14*18 size block requires a 16*20 load), but requires more registers per thread (thus potentially limiting the number of threads processed at once). This change has increased the speed (with reference to table (**7**) of ref [75]) of the simulation by 15% for high precision calculation. This is more emphatically explained in Table 5.10, which compares the rate of calculation in terms of $10^6$ points calculated per second.

| Table 5.10 GPU Speedup Through Shared Memory Accesses | | | | | |
|---|---|---|---|---|---|
| | Main Memory Loads | Texture + Shared Memory Loads | FLOPs per thread | $10^6$ Points / s (peak) | Speedup |
| CPU Explicit, 1 thread [75] | -- | -- | 50 | 21 | -- |
| CPU Explicit, 16 threads [75] | -- | -- | 50 | 108 | 5.1x |
| GPU Without Using Shared Memory | 7 | 7 + 0 | 50 | 565 | 26.9x |
| GPU Using Shared Memory | 1.5 | 7 + 6 | 50 | 640 | 30.5x |

### 5.3.5 Computer Utilisation

GPU analyses were using an NVIDIA GTX 280 (1GB RAM) on an AMD Athlon 3000+ (1.8 GHz) with 2GB RAM. The NVIDIA GTX 280 has CUDA Compute 1.3. Simulations were compiled in Microsoft Visual C++ 2008, using CUDA Visual Studio Wizard (for GPU) and speed optimisations under CUDA 2.2.

### 5.3.6 Results and Discussion

#### 5.3.6.1 Agreement with Previous Theory

The Aoki all-$\tau$ equation 5.32 describes microbands where $L \to \infty$; these, in agreement with results published by Strutwolf [1], were accurately replicated for $L$ = 1000, as shown in figure 5.11, to within 1%.

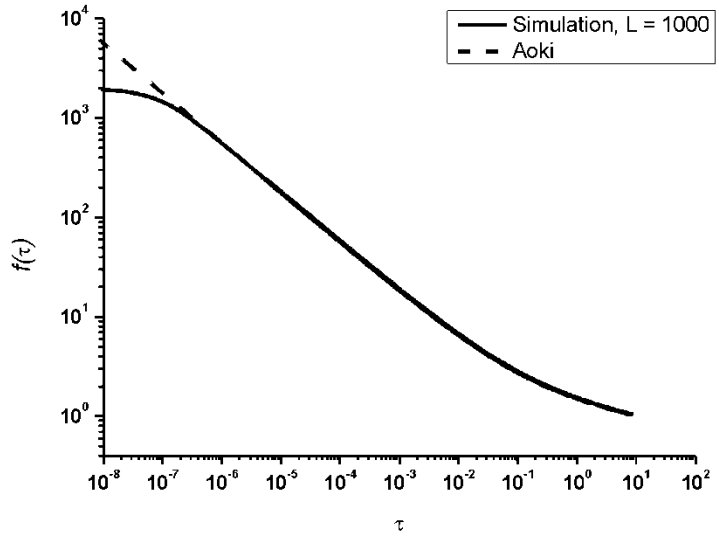



Figure 5.11 Comparison of simulation where $L$ = 1000 to the Aoki all-$\tau$ equation.

As noted in previously in [75] and shown by equation 5.41, the maximum value of $f(\tau)$ is limited by the $\Delta Z$ spacing at the electrode surface, giving the resultant limited maximum of $f(\tau)$ when $\tau < 10^{-6}$.

A plot of the flux at the electrode surface, as shown in figure 5.12 for an electrode of $L$ = 1 at $\tau = 1.25$, shows the simulation accurately finds that the point of the rectangular electrode encourages a larger flux density than the rectangular electrode edges, which in turn provides a larger flux density than the main electrode surface.

The effect of the rectangle point on diffusion can be graphically represented by comparison of the concentration change at various values of $\tau$, as shown in figure 5.13 for a square electrode. At $\tau = 1$, a microdisk nature of diffusion is observed around the point, leading to a pseudo-hemispherical shaped diffusion around the whole electrode at large distances compared to the contorted edge diffusion near the electrode point.

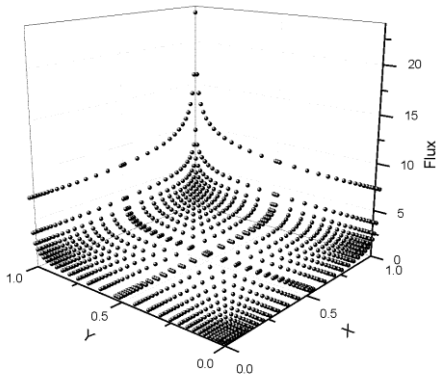



Figure 5.12 Calculated flux at each point of the mesh on the electrode surface, showing the peak flux occurring at the corner of the rectangular electrode.

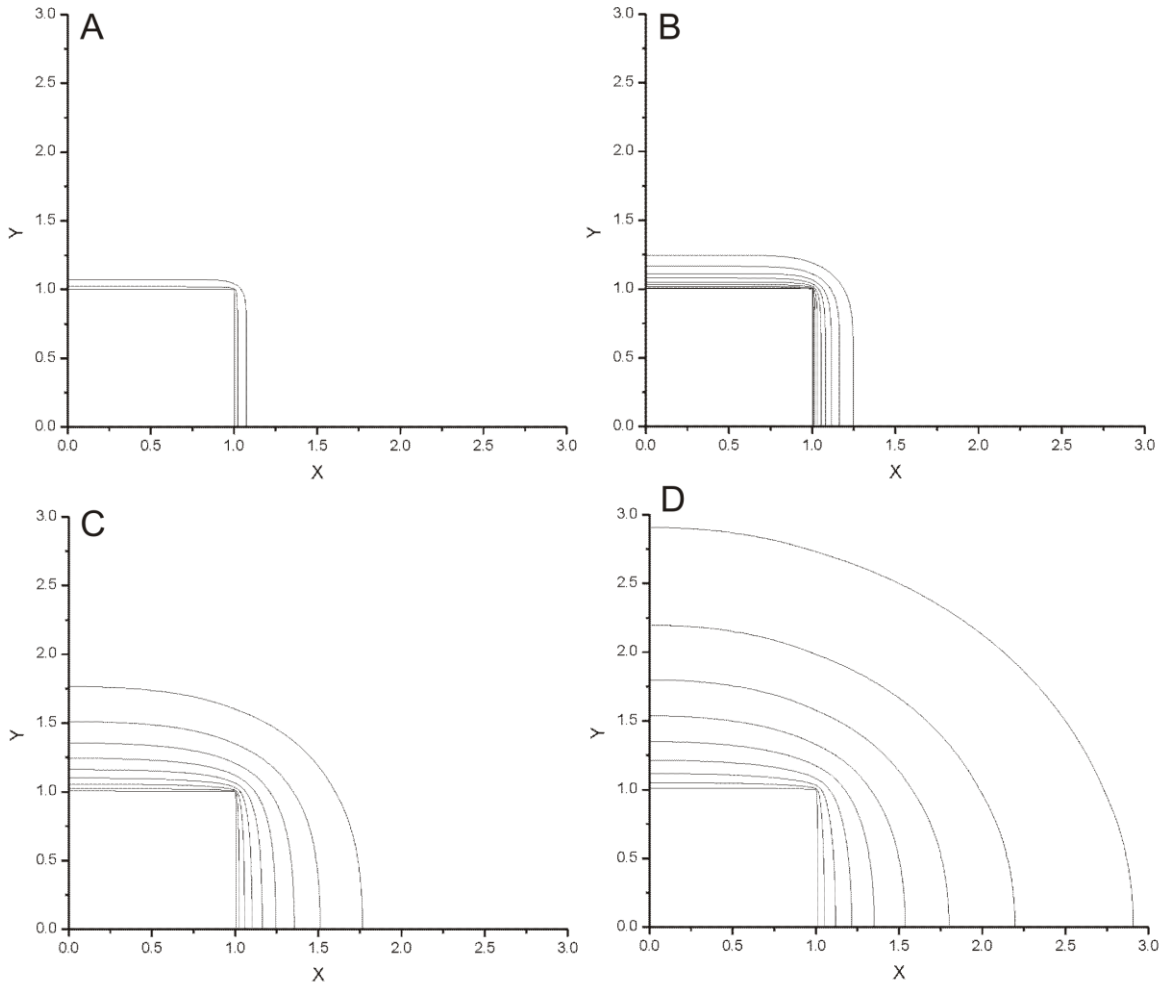



Figure 5.13 Contour representations of diffusion at $\tau =$ (A) 0.0025, (B) 0.01, (C) 0.1 and (D) 1 for a square electrode.

### 5.3.6.2 Extension to Values of $L$ Other Than Infinity

Simulations were run also for microbands of $L$ = 1 (a square electrode), 5, 10, 50, 100 and 1000. The transients, shown in figure 5.14, show that as the electrode becomes shorter, essentially from microband to rectangle to square, the presence of the side and point diffusion lead to the formation of a true steady state current-like representation.

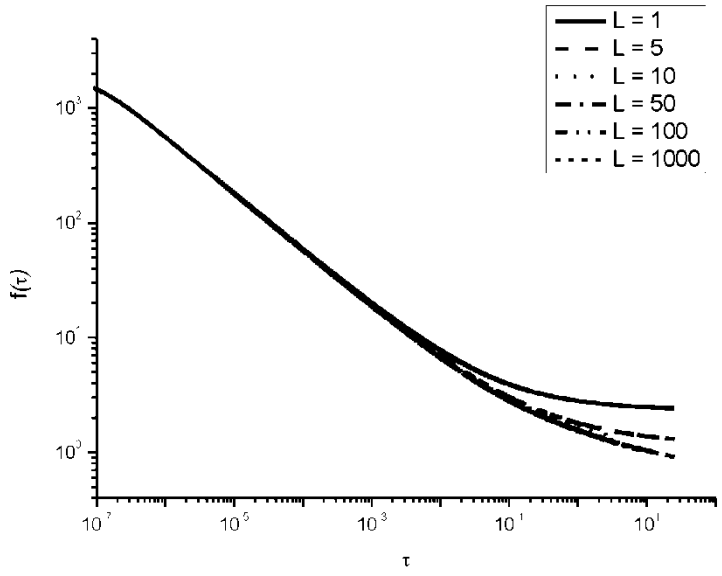



Figure 5.14 Transients of $f(\tau)$ of microbands of $L$ = 1, 5, 10, 50, 100 and 1000.

Each of the transients for the differing $L$ values was fitted to a generalised Aoki-type equation of the form shown in equation 5.43:

$$f(\tau) = \frac{1}{\sqrt{\pi\tau}} + A + Bexp\left(\frac{C}{|ln(D\tau)|}\right), f(\tau) < 1\% \; for \; \tau < 25 \qquad [5.43]$$

where coefficients A, B, C and D are to be determined. Using in-house analysis software which combines user input with least squares calculations, each of the plots in figure 5.14 were fitted to equation 5.43, as shown in figure 5.15, to within 1% for $10^{-6} < t < 25$.

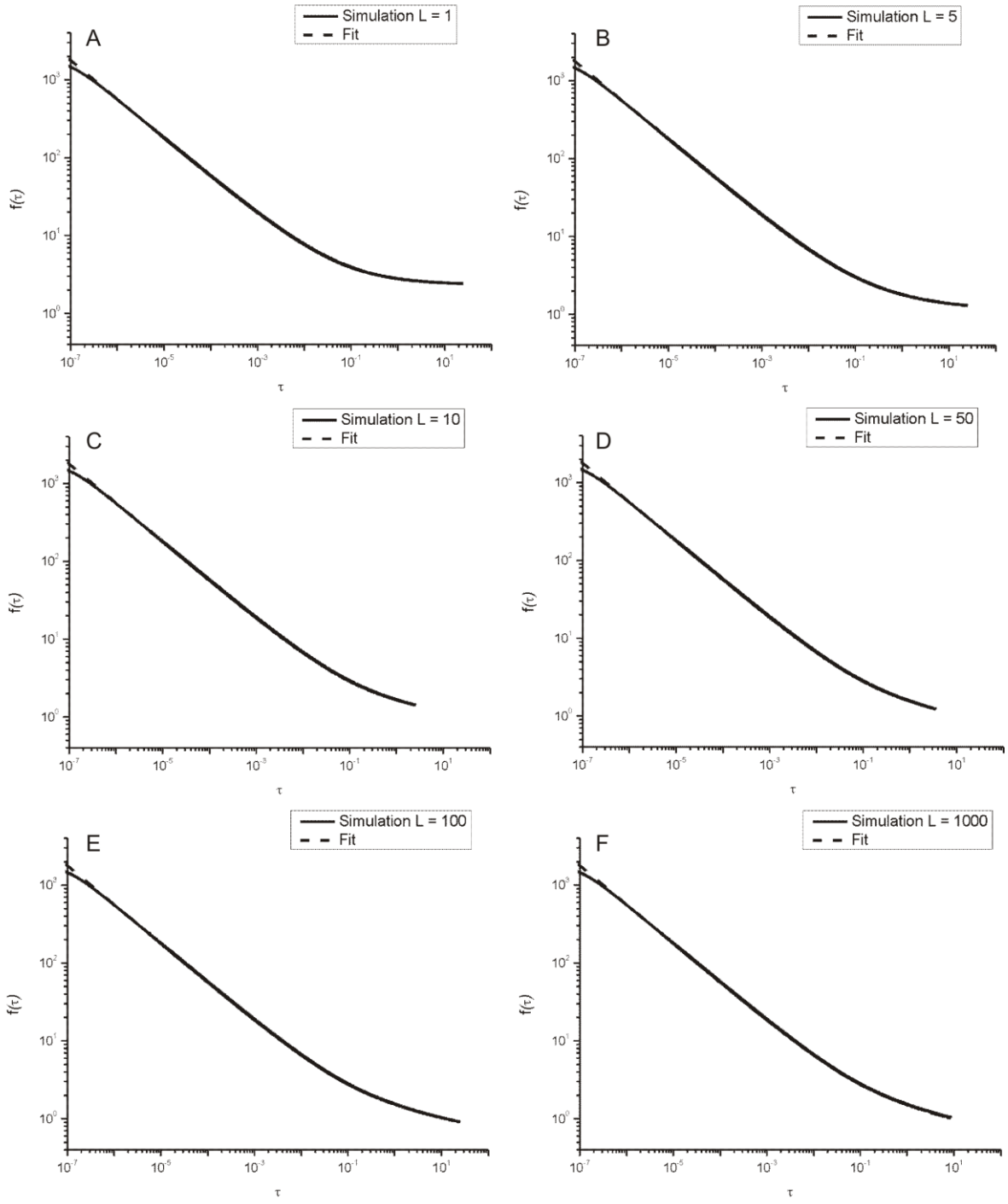



Figure 5.15 Transient plots of simulations and fitted Aoki-type equations for $L$ = (a) 1, (b) 5, (c) 10, (d) 50, (e) 100 and (f) 1000.

#### 5.3.6.3 Fitting of coefficients

Table 5.11 shows the fitted values of each of the coefficients A, B, C and D from equation 5.43 for each of figures 5.15 a-f, along with the Aoki value of each coefficient. Each of the values of A, B, C and D follow a trend as $L$ increases. These trends were plotted and fitted, as shown in figure 5.16. Each of the coefficients A, B, C and D can be expressed in terms of $x = \log_{10} L$ to give equations 5.44 to 5.47:

$$A = 1.004 + 1.3exp(-2.53x) \quad [5.44]$$

$$B = -1.133 + \frac{0.59}{1+exp\left(\frac{x-1}{0.3415}\right)} \quad [5.45]$$

$$C = -9.85 \quad [5.46]$$

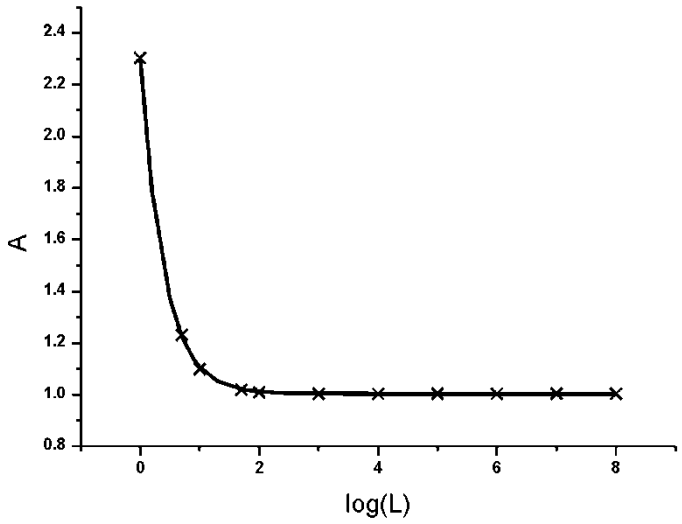
$$D = 17.22 - \frac{18.04}{1+exp\left(\frac{x-1.113}{0.367}\right)} \quad [5.47]$$

The functions of these coefficients have been calculated using Origin 8 fitting.

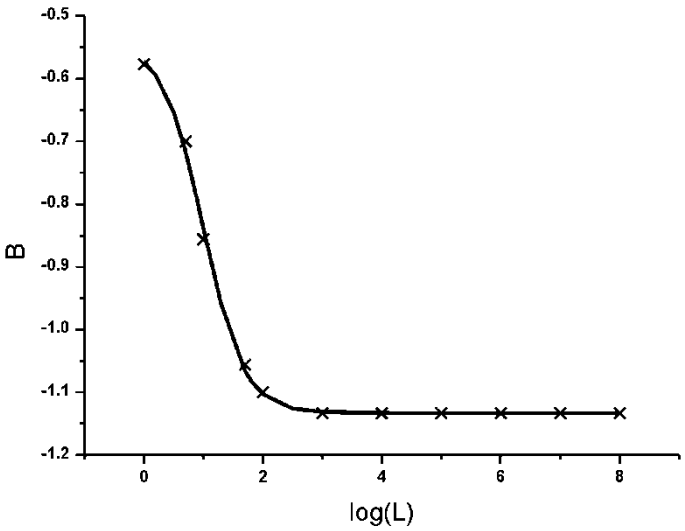
**Table 5.11 Values of coefficients A, B, C and D from figure 5.14**

| L | 1 | 5 | 10 | 50 | 100 | 1000 | Aoki |
|---|---|---|---|---|---|---|---|
| **A** | 2.303 | 1.232 | 1.099 | 1.02 | 1.009 | 1.004 | 0.97 |
| **B** | -0.577 | -0.7 | -0.856 | -1.056 | -1.1 | -1.133 | -1.1 |
| **C** | -9.85 | -9.85 | -9.85 | -9.85 | -9.85 | -9.85 | -9.9 |
| **D** | 0.0063 | 2.45 | 8.0015 | 13.92 | 15.23 | 17.22 | 12.37 |

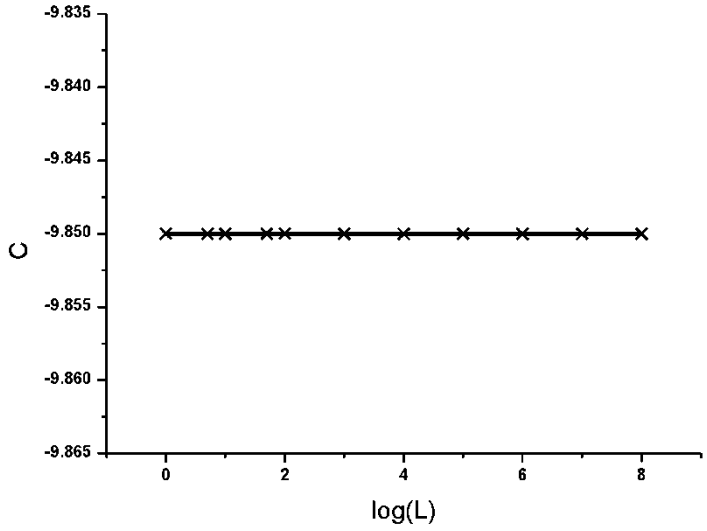
(a)


A
log(L)
2.4
2.2
2.0
1.8
1.6
1.4
1.2
1.0
0.8
0
2
4
6
8

(b)


B
log(L)
-0.5
-0.6
-0.7
-0.8
-0.9
-1.0
-1.1
-1.2
0
2
4
6
8

(c)


C
log(L)
-9.835
-9.840
-9.845
-9.850
-9.855
-9.860
-9.865
0
2
4
6
8

(d)

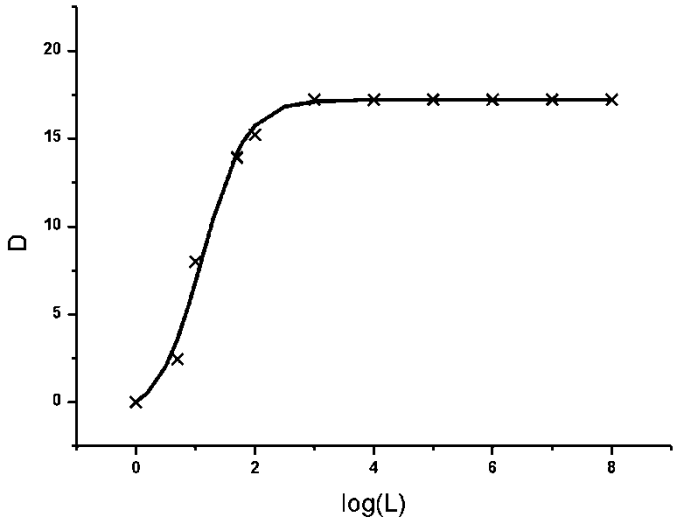



Figure 5.16 Values and fits of the coefficients of (a) A, (b) B, (c) C, and (d) D for the fitted equations shown in figure 5.15 against the Aoki-type equation 5.32.

By combining equation 5.43 with the coefficient functions of equations 5.44 to 5.47, a generalised equation for the transient at a rectangular microband, of any L, can be provided, as shown in equation 5.48. A plot of equation 5.48 against $\log_{10}L$ and $\log_{10}\tau$, shown in figure 5.17, shows how, at low $\tau$, each rectangular microelectrode experiences linear diffusion, and how edge diffusion contributes a larger factor earlier for small electrodes than large electrodes.

$$f(\tau) = \frac{1}{\sqrt{\pi\tau}} + 1.004 + 1.3exp(-2.53x) + \left(-1.133 + \frac{0.59}{1+exp\left(\frac{x-1}{0.3415}\right)}\right) exp\left(\frac{-9.85}{\left|ln\left(\tau\left(17.22-\frac{18.04}{1+exp\left(\frac{x-1.113}{0.367}\right)}\right)\right)\right|}\right) \quad [5.48]$$

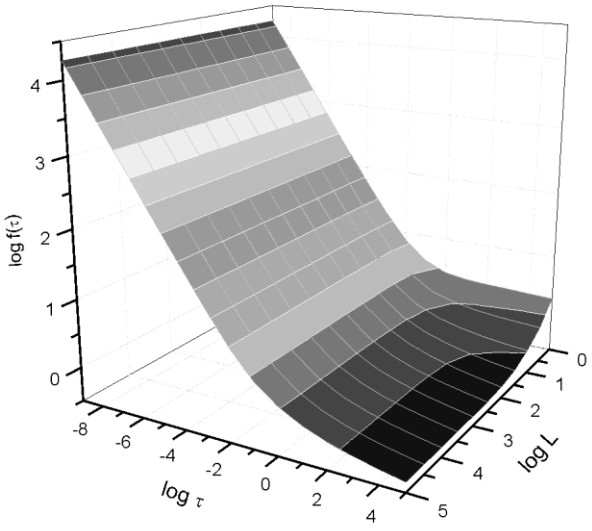



Figure 5.17 General equation for the transient at a rectangular microband, equation 5.43, plotted against $\log_{10}L$ and $\log_{10}\tau$

### 5.3.7 Conclusions

The three-dimensional GPU explicit calculation of square, rectangular and microband electrodes has been described in this chapter. From comparison to Aoki's all-$\tau$ equation [60] for the chronoamperometric transient of long microband electrodes to ensure simulation accuracy, this principle was expanded to microbands of various lengths. As a result, a general expression to describe the chronoamperometric transient of microband electrodes has been produced, based on Aoki's equation, whereby the numerical coefficients have been replaced with functions fitted from the results of this work. In addition, the use of explicit calculations aligned to GPU usage is again seen to promote a useful approach to electrochemical simulations.

### 5.4 Dual Microdisk Electrodes in Transient Generator-Collector Mode

Arrays of microdisk electrodes have been studied in great detail[18,28,76], including partially blocked arrays and malformed microdisks exhibiting elliptical geometries[75]. The use of single electrodes, both micro- and macro-, are employed in the study of electrode processes and reaction mechanisms, as well as analytically and for the measurement of diffusion coefficients. In addition, the use of double electrodes has been considered to offer additional information and insights, starting with the invention of the ring-disk electrodes by Nekrasov[77,78], subsequently much developed by Bard[79-82], and by Albery[83,84]. These experiments are generally operated in a 'generator-collector mode', with species made on one 'generator' electrode being detected on the second 'collector' electrode. In this way, intermediates in the electrochemical process occurring at the first electrode can be identified and their lifetimes inferred. The collector experiment was conducted under steady-state conditions with the 'collection efficiency' defined as the ratio in equation 5.49:

$$N = \frac{i_{collector}}{i_{generator}} \qquad [5.49]$$

which reflects the amount of material arriving at to the collector after diffusion[85,86]. Such measurements can also be operated in transient mode, where 'time of flight' experiments reflect, in a highly sensitive manner, both the diffusion coefficient and the homogeneous chemical reaction rates of electro-generated species; such experiments are normally conducted with a potential step at the generator electrode with the collector current measured at a potential corresponding to a transport limited detection of the product or intermediate of interest.

These systems have been studied extensively in terms of two dimensional diffusion, where the collector and generator are both parallel microbands [49,50,87-90], channel electrodes [47,49,91], wall-jet electrodes [92,93], or at a ring-disk electrode [54,73,94-100]. To a lesser extent, due to the stimulatory three-dimensional nature of the system, the dual-microdisk generator collector system has also been probed [85,86,95,101], as well as dual micro-hemispheroidal electrode systems [86,102,103].

### 5.4.1 Dual Microdisk Systems

Marken et al. have investigated dual-microdisk systems in generator-collector mode, including transport junctions [103-106] and microwave enhanced analytical processes [102,107]. Zhang et al. [108] have published fabrication methods, and analysis of films on dual microdisk electrode systems. Phillips et al.[95] have developed numerical analysis for dual microdisk collector-generator steady-state system for which a heterogeneous first-order reaction may consume the species during travel from the generator to the collector.

Simulation of dual microdisk systems using a random walk algorithm has been primarily attempted by Baur et al. [85]. Using a computational method devized by Nagy et al.[109] and based on Brownian motion [110], Baur et al. simulate individual elements (compounds or ions) within a domain, and when a particle interacts with an electrode surface, given the current state of the particle and the potential of the surface, electron transfer may occur. The electron transfers per step can be used to output a current. Despite the transient nature of the simulation, results to experiment were compared only at steady state. Baur et al. assumed a semi-infinite boundary at a distance of 12 radii divided by the diffusional distance moved each step – for example, where a semi-infinite

boundary is set at 60 diffusional distances from the electrode surface, and thus after 60 steps, an element could have hit the electrode, then a side boundary, and re-enter the other side, much like an array, despite the simulation being run over 10,000 steps. This is in contrast to common simulation practice, where a diffusion domain size proportional to the experimental time has been extensively used successfully [14,69-73,111] and is known to sufficiently exceed the diffusion layer in all cases [17]. Also of note, is the method by which Brownian diffusion was modelled – a pseudo pyramidal diffusion from the electrode is observed due to tri-axial random movement being considered, rather than an easily implementable spherical diffusion model.

Qiu et al. have utilized the boundary element method (BEM) effectively to simulate the collection efficiency at steady state between two microdisk electrodes[86], and the transient of both the generator and collector at specified distances[101]. The work presented in this chapter section expands on that done by Qiu et al.[86] by altering the diffusion coefficient ratio (shown to be very significant in the experimental system shown below), and providing general expressions for the transient collection efficiency at such a system of equal diffusion coefficients, using the simplistic yet computationally intensive explicit finite difference method, which has been successfully applied to elliptical electrodes[75] and finite band electrodes[112] by way of relatively novel computer hardware utilisation in the form of a graphics processor. The results are compared to the transient time of flight experiments made using the $Ru(NH_3)_6^{2+/3+}$ system, and the relative diffusion coefficients of $Ru(NH_3)_6^{2+}$ and $Ru(NH_3)_6^{3+}$ compared.

### 5.4.2 Theoretical Model

Effective simulation of the chronoamperometric response of a redox couple allows characterization of dual-electrode systems for the experimentalist, and provides a basis for more complex and three-dimensional orientations.

In this chapter section, the theory of potential step chronoamperometry at a diffusion-only dual-microelectrode system is tackled by way of the redox couple of a species over two electrodes. We consider a system whereby two microdisk electrodes are embedded in the insulator and flush against the insulator surface. Material is considered to diffuse within in the simulation space, where migratory [24] and convection effects are negligible, by way of Fick's second law to solve the electrolysis current as a function of time:

$$\frac{\partial C}{\partial t} = \mathrm{D_C}\nabla^2 C, \quad C = [A], [B] \qquad [5.50]$$

for typical voltammetric and chronoamperometric measurements, where $C$ is the concentration of species A or B, $D_C$ is the diffusion coefficient of the species of $C$, and $t$ is time.

The finite difference system used in this work is similar to that found in references [75] and [112], as well as chapters 5.2 and 5.3, which have been used previously to great effect. Parameters are noted in table 5.12 and dimensionless parameter transforms are noted in table 5.13.

| Table 5.12 Parameters used in this work | |
|---|---|
| **Parameter** | **Representation** |
| Cartesian Coordinates | $x, y, z$ |
| Time Ordinate | $t$ |
| Species Concentration | $C = [A], [B]$ |
| Diffusion Coefficient of Species C | $D_C$ |

| Table 5.13 Dimensionless Parameter Transforms | |
|---|---|
| **Parameter** | **Transform** |
| Cartesian Coordinates | $X = \frac{x}{r_e}, Y = \frac{y}{r_e}, Z = \frac{z}{r_e}$ |
| Time Ordinate | $\tau = \frac{D_A t}{{r_e}^2}$ |
| Species Concentration | $c = \frac{[C]}{[A]_{bulk}}$ |
| Inter-Electrode Distance | $2d = \frac{edge-to-edge}{r_e}$ |
| Half-gap to radius ratio | $d = \frac{1}{2} * \frac{edge-to-edge}{r_e}$ |

Due to system symmetry, only half the system need be modelled with the appropriate boundary conditions, as shown in Figure 5.18, where *2d* is the shortest dimensionless separation distance from the generator electrode edge to the collector electrode edge. Also, a dimensionless diffusion coefficient ratio $D_R$ is defined as the ratio of the diffusion coefficients between species B and A, and thus equation 5.51 applies, and the time ordinate transform is relative to $D_A$.

$$D_R = \frac{D_B}{D_A} \qquad [5.51]$$

Potential step chronoamperometry was simulated at the electrode surfaces by solving the dimensionless three-dimensional form of equation 5.50, as shown in equation 5.52a for species $A$ and equation 5.52b for species $B$. The finite difference form of equation 5.52a, which is solvable computationally, is shown in equation 5.53, utilising expanding mesh techniques, variables defined in table 5.14, and computational GPU methods as

described in reference [75]. Boundary conditions of the system for the potential step are provided in table 5.15.

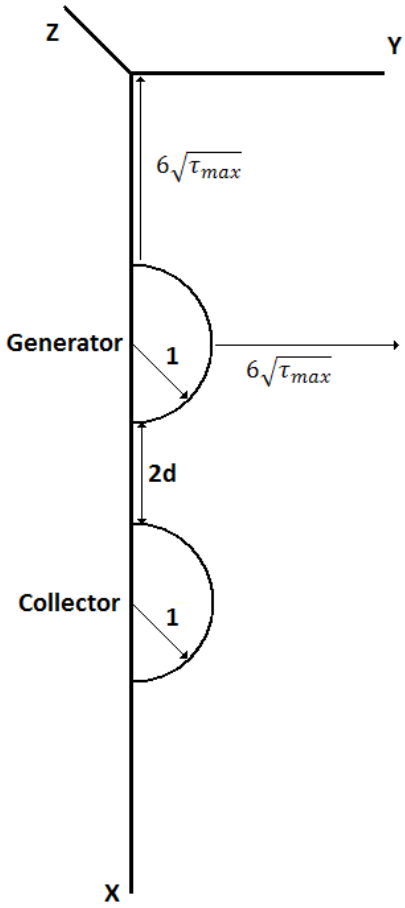



Figure 5.18. Dimensionless electrode system modelled by simulation

$$\frac{\partial a}{\partial \tau} = \frac{\partial^2 a}{\partial X^2} + \frac{\partial^2 a}{\partial Y^2} + \frac{\partial^2 a}{\partial Z^2} \qquad [5.52a]$$

$$\frac{\partial b}{\partial \tau} = D_R \left[\frac{\partial^2 b}{\partial X^2} + \frac{\partial^2 b}{\partial Y^2} + \frac{\partial^2 b}{\partial Z^2}\right] \qquad [5.52b]$$

$$\frac{a^{t+1} - a^t}{\Delta\tau} = \frac{\frac{a^t{}_{X+1} - a^t{}_X}{\Delta X_+} - \frac{a^t{}_X - a^t{}_{X-1}}{\Delta X_-}}{\frac{1}{2}(\Delta X_+ + \Delta X_-)} + \frac{\frac{a^t{}_{Y+1} - a^t{}_Y}{\Delta Y_+} - \frac{a^t{}_Y - a^t{}_{Y-1}}{\Delta Y_-}}{\frac{1}{2}(\Delta Y_+ + \Delta Y_-)} + \frac{\frac{a^t{}_{Z+1} - a^t{}_Z}{\Delta Z_+} - \frac{a^t{}_Z - a^t{}_{Z-1}}{\Delta Z_-}}{\frac{1}{2}(\Delta Z_+ + \Delta Z_-)} \qquad [5.53]$$

| Table 5.14 Parameter Substitutions | |
|---|---|
| **Parameter** | **Substitution** |
| $\lambda K, \quad K = X, Y, Z$ | $\frac{1}{\frac{1}{2}(\Delta K_+ + \Delta K_-)}$ |
| $\lambda K_\pm, \quad K = X, Y, Z$ | $\frac{\lambda K}{\Delta K_\pm}$ |

#### 5.4.2.1 Mesh Generation and Time Grid

The simulation space mesh is similar to that used in [75], whereby the mesh is regular in the $XY$ plane over the curved electrode surface in order to focus mesh points on the edges of the electrode. For all other parts of the simulation space, the mesh uses the expanding mesh principles introduced by Gavaghan [17]. As in reference [75], values of $\gamma = 0.01$ and $h = 1.125$ were used, and mesh limits are set at $6\sqrt{\tau_{max}}$ from the electrode, which is known to sufficiently exceed the diffusion layer in all cases [17]. Figure 5.19 represents the mesh used in the $XY$ plane for a dual-microdisk electrode, showing the density of points at the electrode edges.

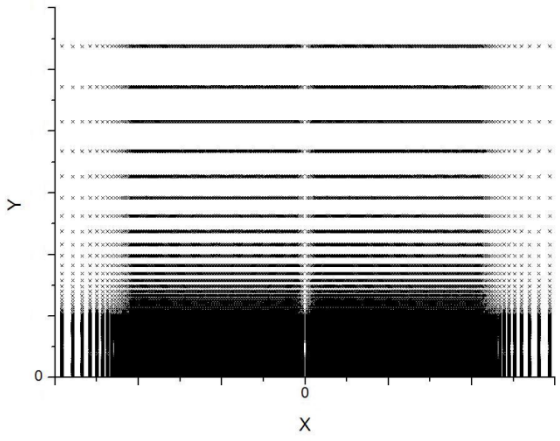



Figure 5.19. Representation of the $XY$ mesh points calculated in simulation. A regular grid is applied over the electrode surface, with expanding grids towards the semi-infinite boundaries and between the electrodes.

The time grid and flux calculations used in this work are identical to those validated in previous publications [75,112] and chapters 5.2 and 5.3.

**Table 5.15. Boundary Conditions**

| Boundary | Location | Expression |
|---|---|---|
| **Generator** | Electrode Surface | $\tau > 0$<br>$a_{Z=0} = 0$<br>$\frac{\partial a}{\partial Z} = -D_R \frac{\partial b}{\partial Z}$ |
| **Collector** | Electrode Surface | $\tau > 0$<br>$b_{Z=0} = 0$<br>$\frac{\partial a}{\partial Z} = -D_R \frac{\partial b}{\partial Z}$ |
| **Insulator** | Insulator, $Z = 0$ | $\frac{\partial c}{\partial Z} = 0$ |
| **Simulation Symmetry** | $X = 0$ | $\partial c / \partial X = 0$ |
| **Simulation Edge** | $X = X_{max}$, $Y = Y_{max}$, $Z = Z_{max}$ | $a = 1$ |
| $\boldsymbol{\tau = 0}$ | All $X, Y, Z$ | $a_{X,Y,Z} = 1$<br>$b_{X,Y,Z} = 0$ |

### 5.4.3 Computer utilisation

GPU analyses were performed using an NVIDIA GTX 460 on an AMD X2 Athlon 5050e (2.6 GHz) with 2GB RAM. The NVIDIA GTX 460 has CUDA Compute 2.0. Simulations were compiled in Microsoft Visual C++ 2008, using CUDA Visual Studio Wizard (for GPU) and speed optimisations under CUDA 2.2. Each explicit digital simulation was performed on the GPU and took on the order of 1-3 days depending on complexity. This is compared to a simulation time of several months (estimated) using code written for a single thread of a CPU.

### 5.4.4 Theoretical Results and Siscussion

Simulations for $d = 0.05 - 2.50$ and $D_R$ = 1 were run, with both the generator and collector currents recorded, as shown in Figures 5.20 and 5.21. As expected from the generator in Figure 5.20, at large inter-electrode gaps, the generator behaves like an isolated electrode and follows the Shoup-Szabo equation. As the half inter-electrode distance, $d$, decreases to within an electrode radius, deviation from the Shoup-Szabo equation is observed. This is due to the effect of the collector regenerating the species consumed at the generator, and lowering the localized concentration gradient - an increase in generator current is observed.

The collector currents in Figure 5.21 show that systems in which the electrodes are close together show the shortest delay, from the start of the simulation, for a collector current to be observed, as expected. As the distance between the electrodes increase, the peak collector current decreases as well as an increase in the delay of the initial current observed.

By dividing the absolute collector current by the generator current, or the absolute collector flux by the generator flux, the collection efficiency, $N$, is obtained, as shown in equation 5.49. By plotting the collection efficiency against the generator-collector half inter-electrode distance $d$, and dimensionless time $\tau$, the surface shown in Figure 5.22 is obtained. Figure 5.22 shows the descending nature of the steady state collection efficiency as $d$ decreases, as well as the initial delay in significant collection efficiencies being observed.

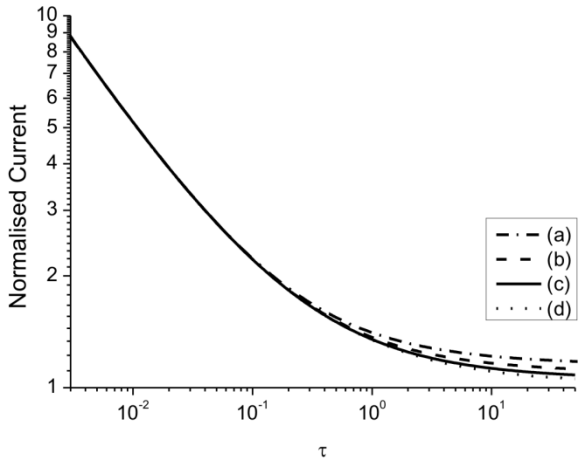



Figure 5.20 Normalized current at the generator electrode as a function of dimensionless time, whereby the half inter-electrode distance, $d$, is (a) 0.10, (b) 0.25, (c) 0.50, (d) 1.00

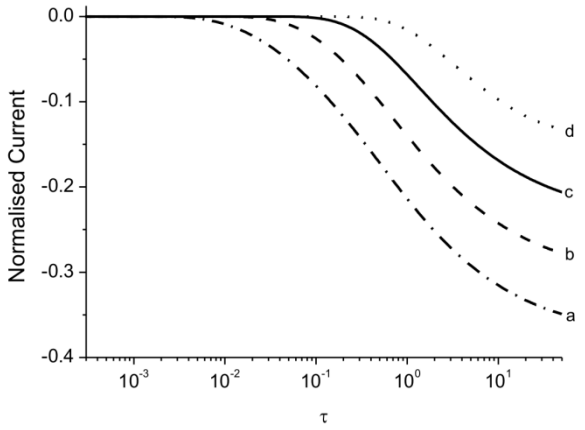



Figure 5.21. Normalized current at the collector electrode as a function of dimensionless time, whereby the half inter-electrode distance, $d$, is (a) 0.10, (b) 0.25, (c) 0.50, (d) 1.00

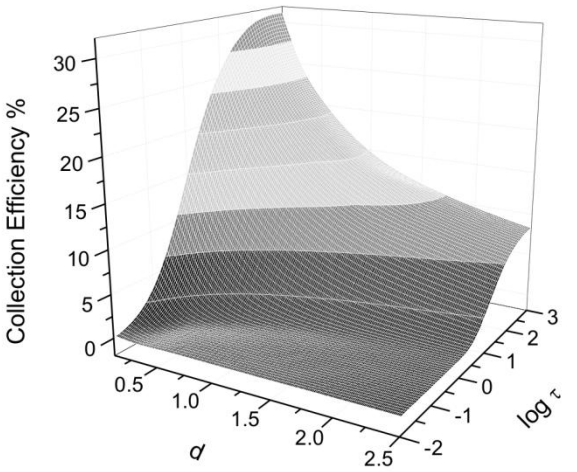



Figure 5.22. Transient Collection Efficiency of the generator-collector couple as a function of half inter-electrode distance, as calculated through simulation.

Also through the simulation, detailed three-dimensional plots of the concentration of species A and B are possible, showing how the diffusion layer distorts around the collector electrode. The concentration profile in the $XY$ plane is shown in Figure 5.23 for a collector-generator system, where the half inter-electrode distance $d = 0.05$, at various $\tau$ through the simulation, where the distortion at longer times is clearly visible.

To provide a general equation to describe the transient nature of the collection efficiency as a function of the half inter-electrode distance, each of the simulated results were fitted to an equation and the coefficients compared. Each simulation was fitted to a Gompertz equation, as shown in equation 5.54, which represents an asymmetric sigmoid with: asymptotes at 0 and $A$ and thus a steady state collection efficiency at $A$; $k$ is indicative the distance between the asymptotic regions; $x_c$ is the horizontal $\tau$-displacement; and the sigmoid inflection point is at ($x_c$, $A * {}^{k}/_{e}$).

$$N_{eff} = Ae^{-e^{-k(log(\tau)-x_c)}} \qquad [5.54]$$

Fitting of the simulations resulted in agreement with results previously reported by Qiu et al [86] for the value of the steady state collection efficiency, $A$. With consequent fitting via Origin 8 software of all the simulations performed, values for the other coefficients $k$ and $x_c$ (to within an $R^2$ fit of 0.995) were obtained for half inter-electrode distances between 0.05 and 2.50, as shown in Table 5.16.

| Table 5.16. Simulation fits to equation 5.54. | | | |
|---|---|---|---|
| **d** | $A$ | $x_c$ | $k$ |
| **0.05** | 33.74 | -0.45610 | 0.95223 |
| **0.10** | 31.24 | -0.30545 | 1.24378 |
| **0.25** | 26.16 | -0.07015 | 1.37122 |
| **0.50** | 20.90 | 0.24203 | 1.47588 |
| **1.00** | 14.91 | 0.59280 | 1.59716 |
| **1.75** | 10.71 | 0.88348 | 1.71385 |
| **2.50** | 8.384 | 1.09391 | 1.80982 |

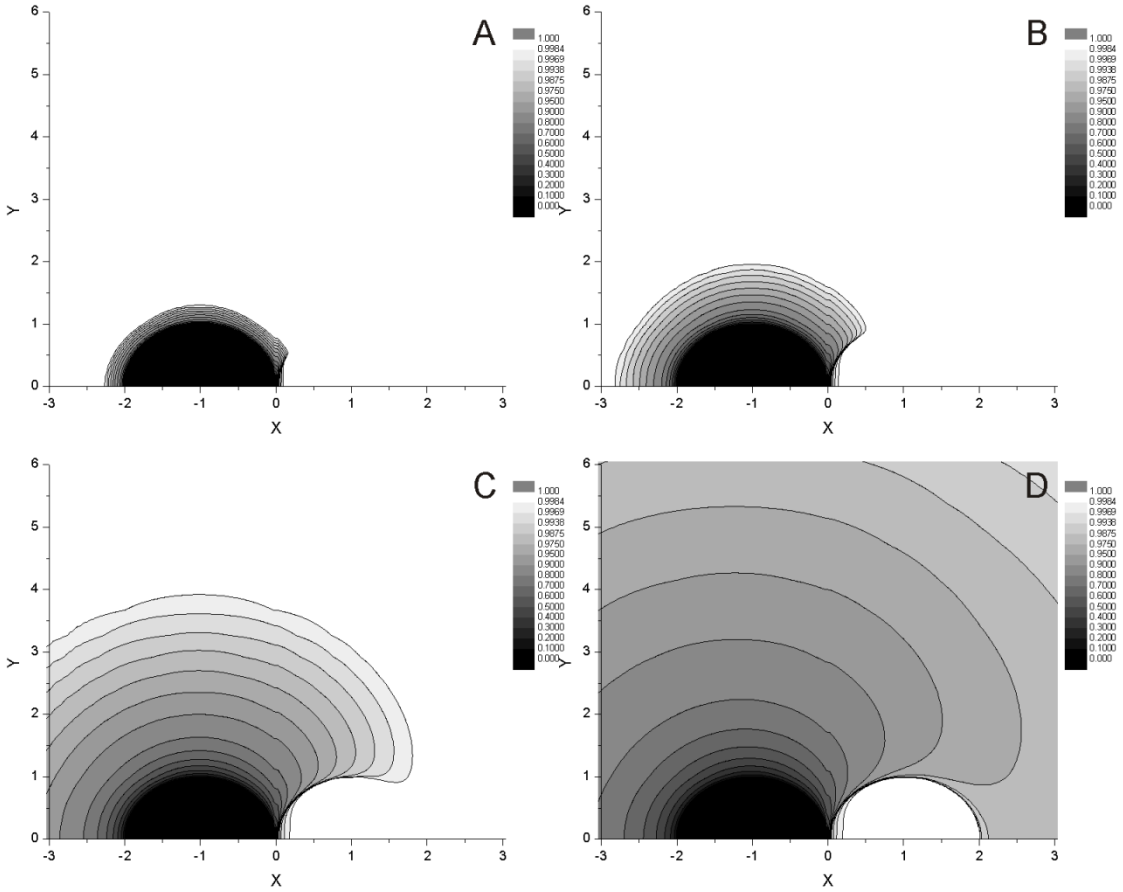



Figure 5.23. Concentration profiles of species A over the generator-collector couple where $d$ = 0.05 at $\tau$ = (a) 0.005, (b) 0.05, (c) 0.5, and (d) 5.

The variations of $A$, $k$ and $x_c$ with respect to half inter-electrode distances were also fitted to relevant equations 5.55-5.57, and fed back into equation 5.54 to give equation 5.58. Equation 5.58 thus is a complete equation describing the transient collection

efficiency dependant on the half inter-electrode between two microdisk electrodes of equal size where both species in solution have equal diffusion coefficients.

$$A = \frac{36.6444}{1+\left({}^{d}/_{0.674}\right)^{0.927}} \quad [5.55]$$

$$k = 1.2189 - 0.0008d^2 + \frac{0.3782d - 0.00644}{\sqrt{d}} \quad [5.56]$$

$$x_c = 0.6206 * ln(2.1811d + 0.3759) \quad [5.57]$$

$$N_{eff} = \left(\frac{36.6444}{1+\left({}^{d}/_{0.674}\right)^{0.927}}\right) e^{-e^{-\left(1.2189-0.0008d^2+\frac{0.3782d-0.00644}{\sqrt{d}}\right)\left(log(\tau)-\left(0.6206*ln(2.1811d+0.3759)\right)\right)}} \quad [5.58]$$

To compliment Figure 5.22, the surface plot showing transient collection efficiency against half inter-electrode distance by way of simulation, Figure 5.24 shows the result of plotting the general equation 5.58. Errors of general equation 5.58 over the half inter-electrode distance $d = 0.05 - 2.50$, are within 1% collection efficiency of the simulations performed.

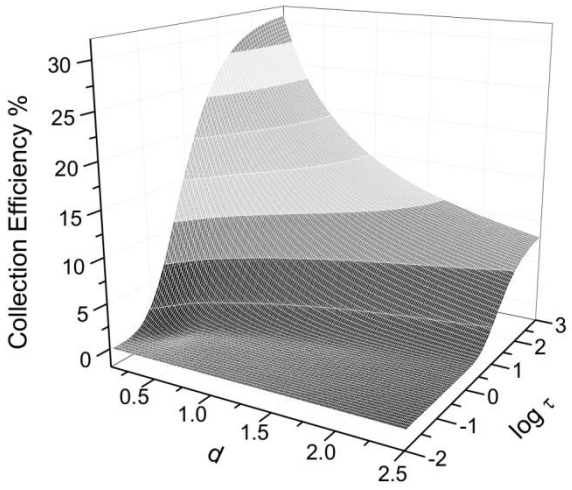



Figure 5.24. Transient Collection Efficiency of the generator-collector couple as a function of half inter-electrode distance, as calculated through equation 5.58.

### 5.4.5 Experimental

Special note should be given to Justin Wang (Oxford), Juan Limon-Peterson (Bath), Liza Rassaei (Bath), Sara Dale (Bath) and Frank Marken (Bath) who have all contributed to the experimental section of this part of the thesis. The work shown in section 5.4.5 and subsections is purely from these colleagues and used to assist the verification of the theory of this chapter.

#### 5.4.5.1 Electrode Radius Measurements

3 mM Hexaammineruthenium(III) chloride ($Ru(NH_3)_6Cl_3$, Aldrich, 98%) and 0.1 M potassium nitrate (Aldrich, >99%) were used without further purification. All solutions were prepared with ultrapure water with a resistivity >18.2 MΩ cm (at 298 K) and degassed with $N_2$ at the start of each experiment. The temperature was maintained constant and controlled by a thermostated Faraday cage. A three electrode cell was used. A dual microdisk electrode, fabricated as described below, with each microdisk tested independently, was used as the working electrode. Gold foil was used as a counter electrode, and a saturated calomel electrode (SCE) was used as a reference electrode. The electrochemical experiments were recorded using a PGSTAT20 potentiostat (Autolab, Netherlands).

#### 5.4.5.2 Double electrode analysis

Optical microscopy was performed using a Leitz Wetzlar (Germany) microscope fitted with a Nikon Fi1 camera on the dual microelectrode, and is shown in figure 5.25. Using the image, a half-gap to generator radius ratio of 0.20 was observed. Both electrodes fall within the 1% steady-state ellipsity as described by chapter 5.2.

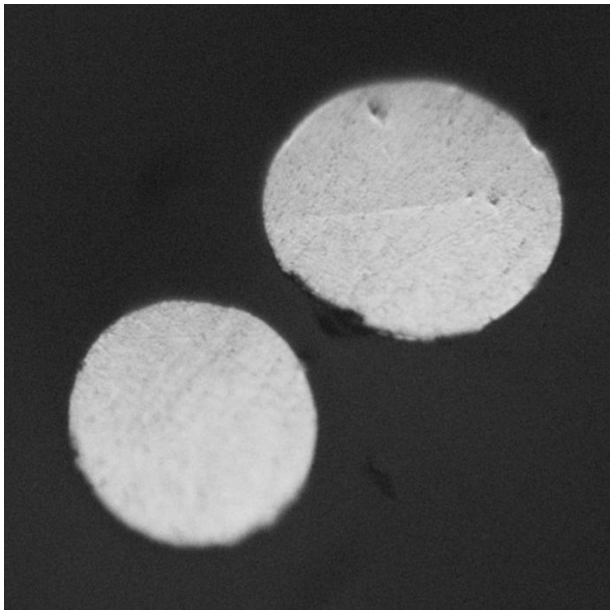


Figure 5.25 Optical microscopy of the dual microelectrode system used.

#### 5.4.5.3 Chronoamperometry on Dual Microdisk Electrodes

Potential step chronoamperometry was applied in a solution, degassed with $N_2$, containing 3 mM hexaammineruthenium(III) chloride and 0.1 M potassium nitrate on the dual electrode system as shown in Figure 5.25. Experiments were performed at 298 ($\pm$2) K with a four electrode glass cell (saturated calomel reference, SCE, platinum wire counter, two adjacent glass-sealed platinum disk working electrodes) and controlled with a Autolab PGSTAT20 (Ecochemie, Netherlands) bipotentiostat. To create the dual microdisk electrode, two platinum wires (100 μm diameter) were sealed into glass with approximately 20μm total separation. The separation is controlled by carefully sealing the two end-twisted Pt wires into glass and then back-polishing the glass to the desired gap [113]. The electrode surfaces were polished with an aqueous slurry of 1 μm alumina (Buehler) on a Buehler microcloth. The collector was potentiostatted at a voltage sufficient to drive the oxidation of $Ru(NH_3)_6^{2+}$ to $Ru(NH_3)_6^{3+}$ at a diffusion limited rate

(+0.2 V vs. SCE). The generator was stepped to a voltage corresponding to a step from zero current to the transport controlled reduction of $Ru(NH_3)_6^{3+}$ (-0.6 V vs. SCE).

### 5.4.6 Experimental results and discussion

Potential step chronoamperometry was performed on each of the electrodes of the dual microdisk electrode shown in figure 5.25 to find exact values of the radii of both electrodes. By fitting with the Shoup-Szabo equation, the active radius of the generator, $r_g$, is equal to 58 μm. The active radius of the collector, $r_c$, is 52 μm, giving a radius ratio $(r_c/r_g)$ of 0.90.

#### 5.4.6.1 Comparison of Theory with Experiment

The experiment, as described in section 5.4.5.3, utilized the dual microelectrode as shown in Figure 5.25 with a 3 mM concentration of $Ru(NH_3)_6Cl_3$ and 0.1 M $KNO_3$ supporting electrolyte. The potential of the collector was held at a sufficient voltage to drive the diffusion limited oxidation of $Ru(NH_3)_6^{2+}$, and the generator was stepped to a voltage resulting in a step from zero current to the transport controlled reduction of $Ru(NH_3)_6^{3+}$.

A simulation, as described in section 5.4.2 and modified using the radius ratio given from single electrode potential-step chronoamperometry, was run using the diffusion coefficient ratio, $D_R = 1.39$, as measured by the double potential-step chronoamperometry by Wang et al.[114], and the half-gap ratio, $d = 0.20$, as obtained from the optical microscopy on the experimental dual microelectrode.

Comparisons between simulation and experimental results show agreement within experimental error with the experimental results, as shown in figure 5.26, demonstrating applicability of the theory to the dual-microdisk experiment.

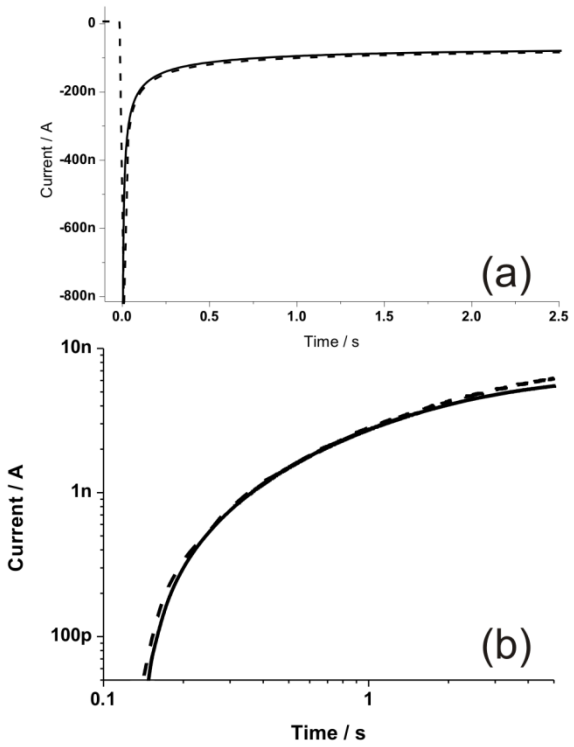



Figure 5.26 Comparison of simulation with experiment, showing (a) the generator and (b) the collector. Solid lines are theory, dashed lines are experiment.

### 5.4.7 Conclusions

The three-dimensional explicit GPU calculation of equal-sized dual microdisk electrodes in generator-collector mode has been described. We have provided a general chronoamperometric transient equation for experimentalists to use in equal-sized dual microdisk collector-generator experiments which fall under diffusion only conditions. Experiments were performed to confirm and demonstrate the utilisation of the technique, and agreed well with theory.

## 5.5 Analysis of Explicit Finite Difference on Graphic Cards

This chapter has described in detail the application of the explicit finite difference algorithm to solve various diffusional mass transport systems where accuracy in three dimensions is required.

Analysis of elliptical electrodes (or deformed microdisks) through this method showed that heavily distorted microdisks can still act as perfect microdisks to great accuracy. In terms of square, rectangular and limited microband electrodes, a general purpose equation for square ended band electrodes of known length has been provided. Similarly, for dual microdisk electrodes, agreement with experiment has shown that in ideal circumstances, when a fully reversible redox couple are in chronoamperometric generator collector mode, the collection efficiency can be accurately predicted.

Through the application of parallel programming, and the use of graphics cards to facilitate high speed digital simulation of these algorithms, three distinct three-dimensional systems have been analysed through finite difference methods.

In the next chapter, we discuss the concept of Random Walk simulation methods, whereby each particle in solution is modelled independently, which was described by D. Britz in 'Digital Simulation in Electrochemistry', as 'this is interesting but hardly recommended, having no obvious advantages over other methods and being presumably somewhat time-consuming'. Chapter six will use the power of graphics cards to dispel this quote, and also some distinct algorithmic improvements on three dimensional diffusional modelling (as well as some important computational aspects worthy of a simulator's attention) are presented.

## Bibliography


[1] J. Strutwolf, Electroanalysis 17 (2005) 1547-1554.
[2] J. Douglas, Numerische Mathematik 4 (1962) 41.
[3] J. Douglas, J.E. Gunn, Numerische Mathematik 6 (1964) 428-453.
[4] J. Noye, in J. Noye (Ed.), Computational Techniques for Differential Equations. Elsevier, Amsterdam 1984, p. 95-354.
[5] C. Amatore, J. Pinson, J.M. Savéant, A. Thiebault, J. Electroanal. Chem. 107 (1980) 59-74.
[6] S.W. Feldberg, J. Electroanal. Chem. 127 (1981) 1-10.
[7] S.W. Feldberg, J. Electroanal. Chem. 290 (1990) 49-65.
[8] B.M. Dasilva, L.A. Avaca, E.R. Gonzalez, J. Electroanal. Chem. 269 (1989) 1-14.
[9] M.V. Mirkin, A.J. Bard, J. Electroanal. Chem. 323 (1992) 1-27.
[10] B.K. Jin, W.J. Qian, Z.X. Zhang, H.S. Shi, J. Electroanal. Chem. 411 (1996) 19-27.
[11] R. Ferrigno, P.F. Brevet, H.H. Girault, J. Electroanal. Chem. 430 (1997) 235-242.
[12] R. Ferrigno, P.F. Brevet, H.H. Girault, Electrochim. Acta 42 (1997) 1895-1903.
[13] J.A. Alden, R.G. Compton, J. Phys. Chem. B 101 (1997) 8941-8954.
[14] S.R. Belding, E.I. Rogers, R.G. Compton, J. Phys. Chem. C 113 (2009) 4202-4207.
[15] I. Svir, A. Oleinick, K. Yunus, A.C. Fisher, J.D. Wadhawan, T.J. Davies, R.G. Compton, J. Electroanal. Chem. 578 (2005) 289-299.
[16] E.C. Du Fort, S.P. Frankel, Math. Tables Aid Comput. 7 (1953) 135.
[17] D.J. Gavaghan, J. Electroanal. Chem. 456 (1998) 1-12.
[18] E.J.F. Dickinson, I. Streeter, R.G. Compton, J. Phys. Chem. B 112 (2008) 4059-4066.
[19] C.E. Banks, R.G. Compton: Understanding Voltammetry, World Scientific, Singapore, 2011.
[20] S.P. Sullivan, M.L. Johns, S.M. Matthews, A.C. Fisher, Electrochem. Commun. 7 (2005) 1323-1328.
[21] I.E. Henley, A.C. Fisher, R.G. Compton, C.E. Banks, J. Phys. Chem. B 109 (2005) 7843-7849.
[22] O.V. Klymenko, C. Amatore, I. Svir, Anal. Chem. 79 (2007) 6341-6347.
[23] C. Amatore, K. Knobloch, L. Thouin, J. Electroanal. Chem. 601 (2007) 17-28.
[24] E.J.F. Dickinson, J.G. Limon-Petersen, N.V. Rees, R.G. Compton, J. Phys. Chem. C 113 (2009) 11157-11171.
[25] M.L. Longmire, M. Watanabe, H. Zhang, T.T. Wooster, R.W. Murray, Anal. Chem. 62 (1990) 747-752.
[26] C.P. Andrieux, B. Limoges, D. Marchal, J.M. Savéant, Anal. Chem. 78 (2006) 3138-3143.
[27] D. Menshykau, I. Streeter, R.G. Compton, J. Phys. Chem. C 112 (2008) 14428-14438.
[28] T.J. Davies, R.G. Compton, J. Electroanal. Chem. 585 (2005) 63-82.
[29] Q. Fulian, A.C. Fisher, J. Phys. Chem. B 102 (1998) 9647-9652.
[30] S. Bruckenstein, J. Janiszewska, J. Electroanal. Chem. 538 (2002) 3-12.
[31] D.P. Whelan, J.J. Odea, J. Osteryoung, K. Aoki, J. Electroanal. Chem. 202 (1986) 23-36.
[32] I. Streeter, R.G. Compton, J. Phys. Chem. C 111 (2007) 18049-18054.
[33] D.J. Gavaghan, J. Electroanal. Chem. 420 (1997) 147-158.
[34] http://www.nvidia.com/object/fermi_architecture.html.
[35] D. Shoup, A. Szabo, J. Electroanal. Chem. 140 (1982) 237-245.
[36] D. Menshykau, X.J. Huang, N.V. Rees, F.J. del Campo, F.X. Munoz, R.G. Compton, Analyst 134 (2009) 343-348.
[37] http://www.ddj.com/hpc-high-performance-computing/210102115.
[38] J. Wang: Microelectrodes VCH, New York, 1990.
[39] D. Pletcher, E.M. Valdes, Anal. Chim. Acta 246 (1991) 267-273.
[40] J. Heinze, Angewandte Chemie-International Edition in English 32 (1993) 1268-1288.
[41] D.H. Craston, C.P. Jones, D.E. Williams, N. Elmurr, Talanta 38 (1991) 17-26.

[42] P.J. Welford, J. Freeman, S.J. Wilkins, J.D. Wadhawan, C.E.W. Hahn, R.G. Compton, Anal. Chem. 73 (2001) 6088-6092.
[43] R.G. Compton, A.C. Fisher, R.G. Wellington, P.J. Dobson, P.A. Leigh, J. Phys. Chem. 97 (1993) 10410-10415.
[44] I. Streeter, N. Fietkau, J. Del Campo, R. Mas, F.X. Munoz, R.G. Compton, J. Phys. Chem. C 111 (2007) 12058-12066.
[45] F. Prieto, J.A. Alden, M. Feldman, B.A. Coles, R.G. Compton, M. Oyama, S. Okazaki, Electroanalysis 11 (1999) 541-545.
[46] C. Amatore, N. Da Mota, C. Sella, L. Thouin, Anal. Chem. 79 (2007) 8502-8510.
[47] C. Amatore, N. Da Mota, C. Lemmer, C. Pebay, C. Sella, L. Thouin, Anal. Chem. 80 (2008) 9483-9490.
[48] I.A. Arkoub, C. Amatore, C. Sella, L. Thouin, J.S. Warkocz, J. Phys. Chem. B 105 (2001) 8694-8703.
[49] C. Amatore, M. Belotti, Y. Chen, E. Roy, C. Sella, L. Thouin, J. Electroanal. Chem. 573 (2004) 333-343.
[50] C. Amatore, C. Sella, L. Thouin, J. Electroanal. Chem. 593 (2006) 194-202.
[51] P. Tomcik, M. Krajcikova, D. Bustin, I. Skacani, Electrochem. Commun. 3 (2001) 191-194.
[52] P. Tomcik, D. Bustin, Fresenius J. Anal. Chem. 371 (2001) 562-564.
[53] P. Tomcik, L. Mrafkova, D. Bustin, Microchimica Acta 141 (2003) 69-72.
[54] I.B. Svir, A.I. Oleinick, R.G. Compton, J. Electroanal. Chem. 560 (2003) 117-126.
[55] C. Amatore, N. Da Mota, C. Sella, L. Thouin, Anal. Chem. 82 (2010) 2434-2440.
[56] P. Tomcik, M. Krajcikova, D. Bustin, Talanta 55 (2001) 1065-1070.
[57] H. Rajantie, J. Strutwolf, D.E. Williams, J. Electroanal. Chem. 500 (2001) 108-120.
[58] F.G. Chevallier, N. Fietkau, J. del Campo, R. Mas, F.X. Munoz, L. Jiang, T.G.J. Jones, R.G. Compton, J. Electroanal. Chem. 596 (2006) 25-32.
[59] K. Aoki, K. Tokuda, H. Matsuda, Denki Kagaku 54 (1986) 1010-1017.
[60] K. Aoki, Electroanalysis 5 (1993) 627-639.
[61] K. Aoki, K. Tokuda, H. Matsuda, J. Electroanal. Chem. 230 (1987) 61-67.
[62] K. Aoki, K. Tokuda, H. Matsuda, J. Electroanal. Chem. 225 (1987) 19-32.
[63] D. Britz, K. Poulsen, J. Strutwolf, Electrochim. Acta 51 (2005) 333-339.
[64] S. Coen, D.K. Cope, D.E. Tallman, J. Electroanal. Chem. 215 (1986) 29-48.
[65] A. Szabo, D.K. Cope, D.E. Tallman, P.M. Kovach, R.M. Wightman, J. Electroanal. Chem. 217 (1987) 417-423.
[66] J.A. Alden, R.G. Compton, J. Electroanal. Chem. 402 (1996) 1-10.
[67] H.L. Woodvine, J.G. Terry, A.J. Walton, A.R. Mount, Analyst 135 (2010) 1058-1065.
[68] I.J. Cutress, E.J.F. Dickinson, R.G. Compton, J. Electroanal. Chem. 638 (2010) 76-83.
[69] I.J. Cutress, R.G. Compton, Electroanalysis 21 (2009) 2617-2625.
[70] S.R. Belding, R. Baron, E.J.F. Dickinson, R.G. Compton, J. Phys. Chem. C 113 (2009) 16042-16050.
[71] F.W. Campbell, S.R. Belding, R. Baron, L. Xiao, R.G. Compton, J. Phys. Chem. C 113 (2009) 14852-14857.
[72] S.R. Belding, E.J.F. Dickinson, R.G. Compton, J. Phys. Chem. C 113 (2009) 11149-11156.
[73] D. Menshykau, A.M. O'Mahony, F.J. del Campo, F.X. Munoz, R.G. Compton, Anal. Chem. 81 (2009) 9372-9382.
[74] D. Menshykau, A.M. O'Mahony, M. Cortina-Puig, F.J. del Campo, F.X. Munoz, R.G. Compton, J. Electroanal. Chem. 647 (2010) 20-28.
[75] I.J. Cutress, R.G. Compton, J. Electroanal. Chem. 643 (2010) 102-109.
[76] D. Menshykau, R.G. Compton, Electroanalysis 20 (2008) 2387-2394.
[77] L.N. Nekrasov, Faraday Discuss. (1973) 308-316.
[78] L.N. Nekrasov, T.N. Khomchenko, K. Almualla, S.A. Martemyanov, B.M. Grafov, Russ. J. Electrochem. 32 (1996) 1185-1190.

[79] K.B. Prater, A.J. Bard, J. Electrochem. Soc. 117 (1970) 207.
[80] K.B. Prater, A.J. Bard, J. Electrochem. Soc. 117 (1970) 1517.
[81] K.B. Prater, A.J. Bard, J. Electrochem. Soc. 117 (1970) 335.
[82] A.J. Bard, J.A. Crayston, G.P. Kittlesen, T.V. Shea, M.S. Wrighton, Anal. Chem. 58 (1986) 2321-2331.
[83] W.J. Albery, M.G. Boutelle, P.J. Colby, A.R. Hillman, Journal of the Chemical Society-Faraday Transactions I 78 (1982) 2757-2763.
[84] W.J. Albery, P.N. Bartlett, A.E.G. Cass, D.H. Craston, B.G.D. Haggett, Journal of the Chemical Society-Faraday Transactions I 82 (1986) 1033-1050.
[85] J.E. Baur, P.N. Motsegood, J. Electroanal. Chem. 572 (2004) 29-40.
[86] F.L. Qiu, N.A. Williams, A.C. Fisher, Electrochem. Commun. 1 (1999) 124-127.
[87] B. Fosset, C.A. Amatore, J.E. Bartelt, A.C. Michael, R.M. Wightman, Anal. Chem. 63 (1991) 306-314.
[88] C. Amatore, C. Sella, L. Thouin, Journal of Electroanalytical Chemistry 547 (2003) 151-161.
[89] H. Rajantie, D.E. Williams, Analyst 126 (2001) 1882-1887.
[90] H. Rajantie, D.E. Williams, Analyst 126 (2001) 86-90.
[91] M. Thompson, E.V. Klymenko, R.G. Compton, J. Electroanal. Chem. 576 (2005) 333-338.
[92] T. McCreedy, P.R. Fielden, Analytical Proceedings 32 (1995) 499-501.
[93] W.J. Albery, B.G.D. Haggett, C.P. Jones, M.J. Pritchard, L.R. Svanberg, J. Electroanal. Chem. 188 (1985) 257-263.
[94] G. Zhao, D.M. Giolando, J.R. Kirchhoff, Anal. Chem. 67 (1995) 1491-1495.
[95] C.G. Phillips, H.A. Stone, J. Electroanal. Chem. 437 (1997) 157-165.
[96] P. Liljeroth, C. Johans, C.J. Slevin, B.M. Quinn, K. Kontturi, Electrochem. Commun. 4 (2002) 67-71.
[97] P. Liljeroth, C. Johans, C.J. Slevin, B.M. Quinn, K. Kontturi, Anal. Chem. 74 (2002) 1972-1978.
[98] S.L.R. Harvey, P. Coxon, D. Bates, K.H. Parker, D. O'Hare, Sensors and Actuators B-Chemical 129 (2008) 659-665.
[99] S.L.R. Harvey, K.H. Parker, D. O'Hare, J. Electroanal. Chem. 610 (2007) 122-130.
[100] D. Menshykau, F.J. del Campo, F.X. Munoz, R.G. Compton, Sensors and Actuators B-Chemical 138 (2009) 362-367.
[101] F.L. Qiu, A.C. Fisher, Electrochem. Commun. 5 (2003) 87-93.
[102] L. Rassaei, R.W. French, R.G. Compton, F. Marken, Analyst 134 (2009) 887-892.
[103] R.W. French, F. Marken, J. Solid State Electrochem. 13 (2009) 609-617.
[104] R.W. French, Y.H. Chan, P.C. Bulman-Page, F. Marken, Electrophoresis 30 (2009) 3361-3365.
[105] M.Y. Vagin, A.A. Karyakin, A. Vuorema, M. Sillanpaa, H. Meadows, F.J. Del Campo, M. Cortina-Puig, P.C.B. Page, Y.H. Chan, F. Marken, Electrochem. Commun. 12 (2010) 455-458.
[106] L. Rassaei, F. Marken, Anal. Chem. 82 (2010) 7063-7067.
[107] M.A. Ghanem, M. Thompson, R.G. Compton, B.A. Coles, S. Harvey, K.H. Parker, D. O'Hare, F. Marken, J. Phys. Chem. B 110 (2006) 17589-17594.
[108] C.G. Zhang, X.Y. Zhou, J. Electroanal. Chem. 415 (1996) 65-70.
[109] G. Nagy, Y. Sugimoto, G. Denuault, J. Electroanal. Chem. 433 (1997) 167-173.
[110] Brownian Motion, Faraday Discussions of the Chemical Society, no. 83, 1987.
[111] D. Menshykau, R.G. Compton, J. Phys. Chem. C 113 (2009) 15602-15620.
[112] I.J. Cutress, R.G. Compton, J. Electroanal. Chem. 645 (2010) 159-166.
[113] R.W. French, S.N. Gordeev, P.R. Raithby, F. Marken, J. Electroanal. Chem. 632 (2009) 206-210.
[114] Y. Wang, J.G. Limon-Petersen, R.G. Compton, J. Electroanal. Chem. (2010) In press.

## Chapter 6 Random Walk Simulation

This chapter introduces individual particle simulation. By modelling each particle in solution, an accurate representation of net molecular movement can be established. Historically, this is a slow simulation due to the number of particles or timesteps required – this chapter details a 4000x speed increase through use of graphics cards and algorithm modification, as well as a pure three-dimensional diffusional model. This chapter will discuss individual particle simulation and creation, and then apply it to low concentration, low volume systems with small numbers of molecules. This chapter also probes the stochastic limit at which cyclic voltammetry can be performed, and investigates the deposition of a monolayer on a nanoparticle in contact with an electrode.

### 6.1 Modelling Individual Particles

Every electron transfer between an electroactive reagent and a surface is discrete – a specific amount of charge is passed in a short space of time. In experimental voltammetry, these quantized events are summed over a time domain, and the total charge passed in that time becomes the result of interest. At ultra-low concentrations, very few electron transfers occur, which results in quantized current readings over the time domain used. However, these quantized events are often obscured by the limitations of the equipment used, so that one stimulus for simulation is the identification of 'true' quantized behaviour over experimental system 'noise'.

In the chapter, we utilise the concept of random-walk simulations (*vide infra*) to probe voltammetric systems from the standpoint that under diffusion only conditions, a particle

in solution will move an average distance per unit time, but in a randomly changing direction, until the particle interacts with a boundary.

A random walk is one where an object moves in a random direction at each step. It is often used as a model for true time-dependent processes, such as diffusion of molecules and colloids, and can be modified to consider weighted directions, or weighted distances moved.

Within electrochemistry and voltammetry, Nagy et al.[1] have reported in a landmark paper the creation of a random-walk electrochemical simulation, detailing concepts of movement along axial directions, simulation space dimensions and boundary conditions. This specific method has been used to explore dual microdisk chronoamperometry[2], finite volume spherical electrodes[3], microdisk electrode arrays in two dimensions[4,5], neurodynamics[6,7], deposition[8-10], and ionic relaxation at a biomembrane[11] and at an electrode[12]. Amatore et al. also probed moving boundary diffusion with aggregation[13] and protein cluster formation[14] in two dimensions using a Gaussian method[15]. We now propose an improved methodology by applying true three dimensional spherical diffusion (in which a molecule can move in any direction, rather than only parallel to the $x$, $y$ and $z$ axes[1]) and open volume semi-infinite boundaries to model true potential step chronoamperometry.

There are two main differences between a random-walk simulation and the finite difference method based on Fick's Laws. The first difference is that, in a random walk simulation, each particle in the simulation is modelled as a point particle, completely independent in its movement with respect to other particles in the simulation (the diffusion coefficient in this context is indirectly indicative of the solvent and supporting

electrolyte used in the experiment). With point particles, the interactions at boundaries can be quantized with respect to the time in simulation. From the perspective of independent particles, we can utilise various simulation techniques designed for parallel problems, such as those brought forward by Cutress et al. in explicit finite difference simulation of elliptical electrodes and finite band electrodes[16,17], but converted for multi-particle simulation. To an extent, this negates various commentaries[18] which criticise the random-walk method for being exceedingly slow.

The second difference is the applicability to three dimensional simulations. Previous methods for tackling three dimensional Fickian diffusion have either relied on explicit finite difference calculation of uncomplicated geometries, which result in large simulation times due to the unstable nature of the discretisation method[16,17]. The alternating direction implicit finite difference method in three dimensions has also been used, but can suffer from oscillation and stability issues depending on the algorithm method used[19-21]. Boundary element simulation can be mathematically complex for this work, and finite element simulation can suffer from poor accuracy[22]. The random-walk scenario is adaptable to a wide range of geometries and voltammetric situations by simple redefinition of boundaries and/or boundary conditions, with little change to simulation code compared to the methods discussed previously. The theory behind the random-walk simulations in this chapter is explained next.

### 6.2 Random Walk – The Theory

We first consider the transient current response at an electrode by stepping from a potential at which no current flows to a potential sufficient to drive a one electron reduction at a diffusion-limited rate:

$$A(aq) + e^- \rightleftarrows B(aq) \quad (1)$$

We assume the effects of migration and convection can be ignored. This system is simulated where particles are free to diffuse until they interact with a boundary, such as the electrode, the insulator, or the diffusion layer boundaries as discussed below. This system is shown in Figure 6.1.

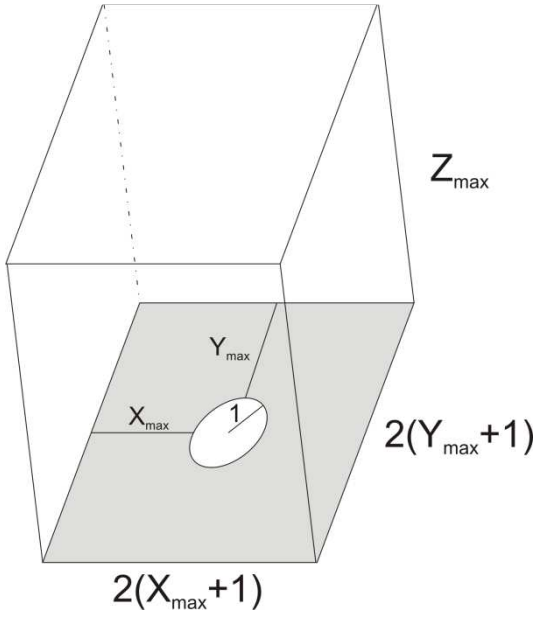



Figure 6.1 Representation of the open solution simulation space, with a dimensionless electrode of radius 1 flush with the insulator.

Preparation for a random-walk simulation takes the steps detailed in sub-chapters 6.2.1 to 6.2.7.

### 6.2.1 Formulation of Dimensionless Quantities

To save simulation time, the system is reduced into dimensionless quantities. This results in the ability to scale the result of one simulation to others and give meaningful comparison of trends. The dimensionless concentration concept in this thesis is different to that used in previous work: $C_E$ (standing for Computational Elements) defines the

number of particles per cubic dimensionless radius in the simulation. Other dimensionless conversions for these simulations are identical to that shown in previous work on finite difference algorithms [23-25], and briefly summarised here.

Dimensionless Length; $K = \frac{k}{r_e}, k = x, y, z;$ [6.1]

Dimensionless Time; $\tau = \frac{D_A t}{r_e^2};$ [6.2]

Dimensionless Concentration; $C_E = [\mathrm{A}] N_A r_e^3.$ [6.3]

In the equations, $r_e$ is the electrode radius, $N_A$ is the Avogadro constant ($6.0221 \times 10^{23}$ mol$^{-1}$), and $[\mathrm{A}]$ is the concentration of species A.

### 6.2.2 Definition of the Simulation Space

For potential step chronoamperometry, we consider a circular electrode flush with an insulator, as shown in Figure 6.1. The boundaries set away from the electrode are semi-infinite boundaries, positioned sufficiently far away from the electrode to exceed the diffusion layer. This has been shown to be a distance from the electrode of six times the square root of the maximum dimensionless time reached by the simulation, $6\sqrt{\tau_{max}}$, by Gavaghan[26], and has been used extensively in previous work [16,17,27]. Equations 6.4 and 6.5 show the calculation of the volume of the simulation space.

$K_{max} = 6\sqrt{\tau_{max}}, \; K = X, Y, Z$ [6.4]

$\text{Volume} = 2(X_{max} + 1) \times 2(Y_{max} + 1) \times Z_{max}$ [6.5]

### 6.2.3 Definition of the Boundary Conditions

For the reaction in equation 1, each particle in the solution is initially given the status of A. Upon contact with the electrode, the particle will be given the status of B if the potential of the electrode is sufficient to transfer an electron. In contrast, a particle with the status of B interacting with the electrode will lead to no change of state. When a particle hits an insulator surface, it 'bounces' on a trajectory defined by the contact vector. If a particle hits the semi-infinite boundary, a periodic boundary condition is imposed, i.e. if a particle exceeds the simulation domain on the left, it enters from the right on the same trajectory. This gives the simulation the effect of being in an array; however the simulation boundaries are set at a distance such that neighboring diffusion domains do not overlap. Should a particle with the status of B interact with the semi-infinite boundary under these conditions, then the simulation boundaries are not set far enough away from the electrode, and this provides an independent evaluation of the validity of the domain sizes. Simulation code for this chapter has been written such that these boundaries are monitored.

When a particle interacts with the semi-infinite boundary in the Z ordinate, one of two conditions can apply – the particle can be replaced at a random $(X, Y)$ co-ordinate across the ceiling to imitate zero net flux into the bulk solution; or the particle can 'bounce' off the surface, as at an insulating boundary. As a semi-infinite boundary at sufficient distance, both of these conditions have the same effect in simulation. However, if the boundary is no longer semi-infinite but rather an insulator, such as in a physically confined system, the 'bouncing' method was applied.

### 6.2.4 Population of the Simulation Space with Particles

The total number of particles in the system is the product of the volume of the simulation space, as defined in equation 6.4, and the dimensionless concentration, $C_E$, which is the number of computational elements per dimensionless radius cubed and is defined in equation 6.3.

Each initial coordinate of every particle is randomly generated from the distance between the boundaries in that coordinate axis. The random number generator used is of great importance – each computer-based random number generator will eventually repeat itself after a number of steps, and this number is known as the period[28]. Typical compiler random number generators have a low period (as low as $2^{15}$ = 32768), resulting in duplicated movement of particles, or repeated movement. The choice of random number generator is typically not given in previous work.

The convention for writing the generation of a uniform random number is such that for a variable $x$ to take a value between 0 and 1, $x \sim U(0,1]$, where 0 is included but 1 is not. Most uniform random number generators will output a value between 0 and 1, and hence to get a value from 0 to $2\pi$ requires a multiplication. Computer-based random number generators are described in Chapter 6.3.1.

### 6.2.5 Movement of the particles for a specified number of steps

The root mean square distance for the movement of a particle in $n$ dimensions is defined by equation 6.6 and is derived from the Einstein-Smoluchowski equation[29,30] relating distance and time, where $D$ represents the diffusion coefficient, and $t$ is the time[31].

Dividing this equation by $r_e$ gives the equation in dimensionless units for the purposes of simulation, and is shown for three dimensions in equation 6.7.

$$\sqrt{\langle r_m^2 \rangle} = \sqrt{2nDt} \quad [6.6]$$

$$\sqrt{\langle R_m^2 \rangle} = \sqrt{6\tau} \quad [6.7]$$

To determine the direction of movement of the particle, one initially tempting (but incorrect) technique is to generate two random spherical co-ordinate angles,$(\theta, \varphi)$ and then to convert to Cartesian positions. This technique, however, produces a weighted distribution at the poles. Instead, the surface area element, $d\Omega$, has to be examined.

In spherical polar coordinates, the solid angle of a sphere, $\Omega$, is defined as the area $A$ covered by that angle divided by the square of the radius, $r_e$, as shown in equation 6.8. The rate of change of this angle with respect to the area at constant radius gives the following set of equations:

$$\Omega = \frac{A}{r_e^2} \quad [6.8]$$

$$d\Omega = \frac{dA}{r_e^2} = \frac{(r.\sin\theta d\theta)(r.d\varphi)}{r_e^2} = \sin\theta d\theta d\varphi \quad [6.9]$$

$$d\Omega = -d(\cos\theta)d\varphi \quad [6.10]$$

Thus $\theta$ has a cosine weighting. This algorithm can then be used by following these steps:

Generate $\varphi \sim 2\pi \times U(0,1]$ [6.11]

Generate $\theta \sim \cos^{-1}(2 \times U(0,1] - 1)$ [6.12]

$$x = R_m \sin\varphi \sin\theta \quad [6.13]$$

$$y = R_m \cos\varphi \sin\theta \quad [6.14]$$

$$z = R_m \cos\theta \quad [6.15]$$

Computationally, $2\pi$ can be predefined as a constant, and thus does not require a multiplication every time it is used. This method requires the generation of two random numbers, seven multiplications, an addition/subtraction, five trigonometric functions and an inverse trigonometric function. Section 6.3.2 shows the computational cost of each of these functions on CPU time on our hardware. Other algorithms have previously been devised in fields other than electrochemistry for the purpose of spherical movement[32,33], and are described briefly in section 6.3.3 along with our own speed comparison to find which algorithm is best suited for simulation. In the case of our computational hardware, the trigonometric method is fastest for both the CPU and GPU, as it minimises the number of trigonometric functions used and requires no rejection of random numbers, and is used throughout this chapter.

The trigonometric method is based on the proof shown in Appendix B that the particles on a sphere will be uniformly distributed on one axis[33]. The particle is then distributed at a random angle on the $XY$ plane due to the constraint in $Z$. This method is given in equations 6.16-6.21:

Generate $z \sim 2 \times R_m \times U(0,1] - R_m$ [6.16]

Generate $\alpha \sim 2\pi \times U(0,1]$ [6.17]

Let $r = \sqrt{R_m^2 - z^2}$ [6.18]

$x = r\cos\alpha$ [6.19]

$y = r\sin\alpha$ [6.20]

The size of the time step is equally important. The time step should be sufficiently small such that particles have a chance to interact with the electroactive surface, and the distribution generated over successive time steps accurately models diffusion. Consider a one dimensional random-walk with a large time step, and an equivalent random-walk with a two time steps at half the size, and half again, such as in Figure 6.2. The small time step random-walk, with the same number of particles, ends up with a different distribution to the large time step walk, and as time steps get smaller, a pseudo Pascal's Triangle is formed in one dimension and the final distribution of the particles converge to a limit. This same concept applies in three dimensions, and is utilised in this thesis.

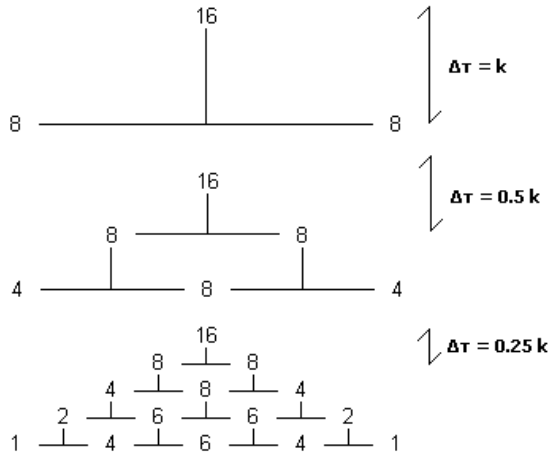



Figure 6.2. One dimensional random-walk starting with 16 particles and equal weighting to go left or right.

In terms of simulation, there are two approaches to consider; (i) simulate all the time steps of a particle, then move on to the next one, or (ii) simulate the first time step of every particle, then the second, until the end of the simulation. Approach (i) assumes that there is no localised interaction between particles to the simulation (such as homogeneous chemistry), and utilises much less computer memory as once each particle has finished its movement, the ordinates of the particle can be discarded. Approach (ii) can be adapted to accept chemical collision reactions, but as the state of each particle needs to be saved between each time step, requires greater computational memory and as a result may take longer to simulate. For potential step chronoamperometry, approach (i) is the most applicable and used in this work.

### 6.2.6 Analysis of the Results

For each successful electron transfer at an electroactive surface, $1.602 \times 10^{-19}$coulombs of charge are recorded. The total current of the time step, $i_\tau$, is thus the total charge transferred divided by the size of the time step, as shown in equation 6.21. The total charge is positive for an oxidation, and negative for a reduction.

$$|i_\tau| = \frac{\text{collisions} \times 1.602 \times 10^{-19}}{\Delta\left(r_e^2 \tau / D_A\right)} \qquad [6.21]$$

As $C_E$ or $\Delta\tau$ decrease, fewer interactions with the electroactive surface will occur each time step, giving a quantized result. To determine the trend in the quantized result, adjacent-point integration is employed, whereby each data point is averaged with a number of points next to it. The level of averaging is dependent on the quantization and determined by the person reviewing the results. If a simulation has $10^6$ data points at $\Delta\tau$

= $10^{-5}$, and the level of averaging is $10^4$, the new integrated $\Delta\tau$ is $10^{-1}$. This technique is used in examining ultra-low concentration chronoamperometry.

### 6.2.7 Adjacent Point Integration

As $\Delta\tau$ is low due to the modelling of complete spherical diffusion, $\Delta t$ is accordingly low, and usually beyond the normal readings taken of modern electrochemical equipment – e.g. at an electrode radius of 10 $\mu$m and diffusion coefficient of $5\times10^{-10}$ m$^2$ s$^{-1}$, $\Delta t$ is 20 $\mu$s for the simulation. At a variety of scan rates and concentrations, this leads to a very high signal-to-noise ratio, where the voltammetry is not obvious, and thus adjacent point integration $\Lambda_s$ is applied, as shown in equation 6.22.

$$i_n = \sum_{n-\frac{\Lambda_s}{2}}^{n+\frac{\Lambda_s}{2}} \frac{i_n}{\Lambda_s} \qquad [6.22]$$

The level of adjacent point integration is typically left to the discretion of the theorist when dealing with stochastic simulations, or the experimentalist due to their experiment situation. However, at high levels of adjacent point integration, extreme error can occur due to using too many points. Figure 6.3 represents this with the results obtained later in this chapter for cyclic voltammetry at a microdisk ($K^0$= $10^4$, $\sigma$ = $10^2$, $C_E$ = $10^5$), showing that as $\Lambda_s$ increases, the voltammetric wave shape becomes easier to decipher, until the point where it becomes wholly inaccurate (Figure 4f).

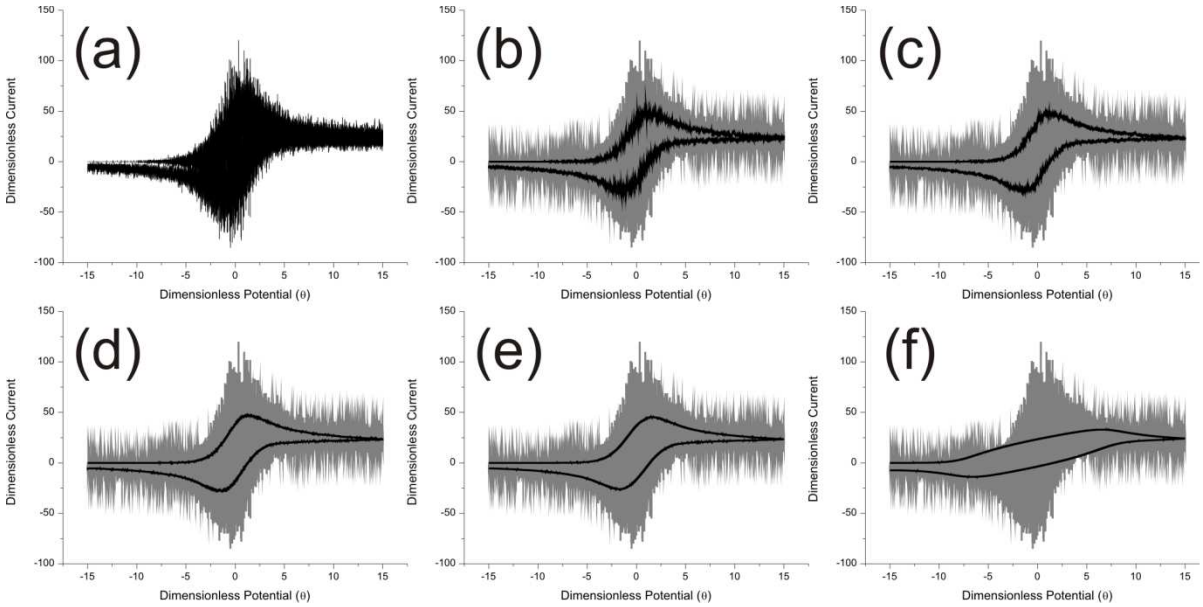



Figure 6.3. Effect of adjacent point integration for $K^0$= $10^4$, $\sigma$ = $10^2$, $C_E$ = $10^5$. (a) represents the original data, and at $\Lambda_s$ = (b) 50, (c) 100, (d) 500, (e) 1000, (f) 5000.

The most appropriate level of $\Lambda_s$ can be determined. For any level of $\Lambda_s$, the value of the points computed can be compared to the original voltammetric readings by the Pearson's product-moment correlation coefficient, as shown in equation 6.23. This provides a value between -1 and +1 for the correlation between the old values and the adjacent point integrated values.

$$r = \frac{\sum_{i=1}^{n}(X_i-\overline{X})(Y_i-\overline{Y})}{\sqrt{\sum_{i=1}^{n}(X_i-\overline{X})^2}\sqrt{\sum_{i=1}^{n}(Y_i-\overline{Y})^2}} \qquad [6.23]$$

where $X_i$ is the original dimensionless current value for a particular time step, $Y_i$ is the newly calculated adjacent point integrated value for the same time step, $\overline{X}$ is the average dimensionless current and $\overline{Y}$ is the average adjacent point integrated current for the whole reading. Statistically, this is comparing the sample covariance between the two ranges and dividing by the sample standard deviations.

The correlations for Figure 6.3 are shown in Figure 6.4. As $\Lambda_s$ increases, correlation decreases to a plateau, before extreme error becomes apparent and $\Lambda_s$ is no longer applicable. It is up to the theorist or experimentalist to decide, based on this correlation representation, which value of $\Lambda_s$ is most applicable.

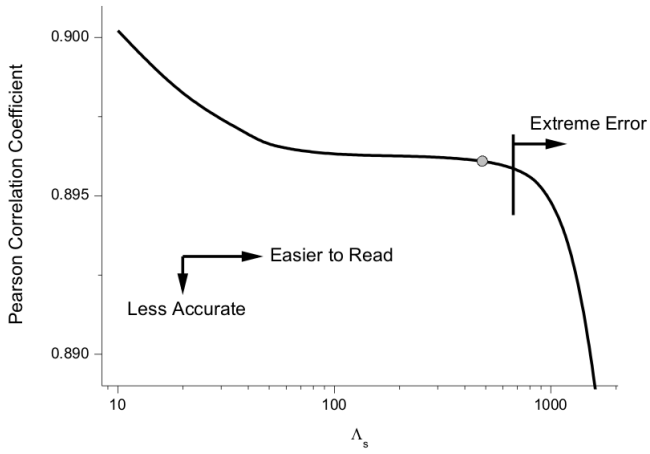



Figure 6.4. Pearson Correlation Coefficient for $\Lambda_s$ from figure 6.3.

## 6.3 The Random Walk Algorithm – Points to Consider

### 6.3.1 The Importance of a Random Number Generator

The concept of random number generators (RNGs) is a vast topic beyond the scope of this thesis, but it is important to note why the choice of a random number generator is crucial.

A true random number generator will measure a physical phenomenon that is expected to be random and compensates for possible biases. However a computational random number generator will generate a long sequence of apparently random results which are determined by a smaller value, known as a seed, and are thus called pseudorandom

number generators (PRNGs). These PRNGs will eventually repeat themselves, and the point at which this occurs is known as the period. The sequence of numbers can be recalled also if the same seed is used.

The choice of the PRNG should have a period greater than the number of times that the generator is called to produce a random number. For example, if a random-walk particle utilises a PRNG with a period of 100, but continues for 10000 steps, the motion of the first 100 steps will be repeated 100 times. While this is statistically plausible, it would happen to every particle in the random-walk simulation, and thus is not a true representation of a random-walk.

The periodicity of PRNGs pre-defined in PC compilers (such as the *rand()* function) is often very poor, and be as little as $2^{15}$ (32768). In previous work on electrochemical random walk theory, there is no mention of the PRNG used, and thus could bring into question any results obtained. In our work, for CPU simulations, we have utilised the freely available Ranq2 PRNG as found in *Numerical Recipes: The Art of Scientific Computing 3rd Ed.* by Press et al.[28], and the XORWOW PRNG for GPUs found in the CURAND library, courtesy of NVIDIA[34] which was initially proposed by Marsaglia[35].

The Ranq2 PRNG has a period of approximately $2^{126}$ and utilises XOR functions and bit-shifts to generate the required numbers. The XORWOW PRNG also utilises XOR functions and bit-shifts, and has a period of approximately $2^{190}$. Both are optimised for speed and have comfortably high periodicity for the random-walk scenarios simulated.

### 6.3.2 Comparison of Computational Operations

Optimisation of computational algorithms lies ultimately in achieving the same results using fewer calculations. However, it should be noted that different types of computational operations require different lengths of time to compute. This is compounded by the fact that computational hardware may have specific sections of the processor dedicated to certain operations. Division, for example, may seem trivial to humans, but requires different techniques within a computer, such as the Newton-Raphson method to calculate the reciprocal of the denominator[36], then multiplication with the numerator. Modern computer processors, such as the Penryn microarchitecture of Intel processors, have dedicated units or methods for trigonometric functions and division[37], which puts the time to perform a division on par with multiplication. This is in contrast to graphic processors, which have dedicated special function units for trigonometric functions, but no dedicated hardware for division.

Table 6.1 shows a list of 32-bit operations, the time taken to calculate $10^9$ iterations of the operation, and the speed relative to addition. These operations were performed using our hardware, on a single thread of a Xeon E5520 CPU at 2.26 GHz with Nehalem microarchitecture, under release mode of Visual C++ Express 2008 with all compiler optimisations enabled. Two of these operations may seem unfamiliar, but are utilised in random number generation for section 6.3.1 and are explained:

*Bit Shift* ($x \ll y$): A bit shift of an integer value takes the binary representation of the number x and shifts the value to the left (<<) or the right (>>) by y number of bits. In an arithmetic shift, bits are discarded and zeroes are generated, whereas in a carry shift, bits will cycle around. Standard rules in C/C++ dictate arithmetic shifts by default.

- Arithmetic Shift: `11100001 << 3 = 00001000`
- Carry Shift: `11100001 << 3 = 00001111`

*XOR* ($x$ ^ $= y$): The XOR logical operation is performed between each bit in both $x$ and $y$, and the result stored in $x$. Thus the XOR between 1001 and 0111 will result in 1110.

**Table 6.1 Comparison of speed of Mathematical Computer Operations**

| Operation | | Time for $10^9$ operations / sec | Speed relative to addition |
|---|---|---|---|
| XOR | x ^= y | 0.843 | 0.199 |
| Bit Shift | x << y | 0.875 | 0.207 |
| Addition / Subtraction | x += y | 4.235 | 1.000 |
| Multiplication | x *= y | 5.000 | 1.181 |
| Division | x /= y | 5.922 | 1.651 |
| Square Root | x = sqrt(y) | 12.141 | 2.867 |
| Sine / Cosine | x = cos(y) | 38.454 | 9.080 |
| Inverse Cosine | x = acos(y) | 50.173 | 11.847 |
| Tangent | x = tan(y) | 50.578 | 11.923 |

### 6.3.3 Comparison of Sphere Point Distribution Algorithms

#### 6.3.3.1 The Cosine Method

The cosine method is derived from the solid angle $\Omega$ and the area element $d\Omega$ of the circle. By determining $d\Omega$, two angles $\theta$ and $\varphi$ can be generated with the appropriate weighting and converted into Cartesian coordinates. Equations 6.11-6.15 show this method.

#### 6.3.3.2 The Normal-Deviate Method

As explained by Knuth[32], each $x$, $y$ and $z$ coordinate is randomly generated from a normal distribution with a mean of 0 and variance of 1. The result is normalised to the surface of the sphere to obtain a uniform distribution on the surface. The following equations describe this method, where λ is the normalisation factor:

$$Let\ u, v, w \sim N(0,1),\ \lambda = \frac{R_m}{\sqrt{u^2+y^2+z^2}} \quad [6.24]$$

$$x = u\lambda, y = v\lambda, z = w\lambda \quad [6.25]$$

This method requires generation of three normal random numbers, one division, one square root, two additions, and six multiplications. It should be noted that the generation of normally distributed random numbers requires an element of rejection, and thus generating normal random numbers takes significantly longer than generating uniform random numbers.

#### 6.3.3.3 The Hypercube Rejection Method

Similar to the normal-deviate method, but each ordinate is uniformly randomly distributed on $[-1, 1]$ (calculated as $2 \times U(0,1] - 1$). Compute the square root of the sum of squares, and if this value is greater than 1, the triplet is rejected. If this value is below 1, the vector is normalised and scaled to $R_m$.

Generate $u, v, w \sim 2 \times U(0, 1] - 1$ [6.26]

Let $s^2 = u^2 + v^2 + w^2$; if $s > 1$, reject, else $\lambda = R_m/s$ [6.27]

$$x = u\lambda, y = v\lambda, z = w\lambda \quad [6.28]$$

Before comparison, this method requires generation of three uniform random numbers, a square root, five additions and six multiplications. After comparison, another division and three multiplications are required. As this method requires comparison and rejection, the chance of a triplet being inside the sphere as required is governed by the areas of a cube and sphere, such that 6/3.141 ≈ 1.910 triplets are required to achieve one which passes comparison.

#### 6.3.3.4 The Trigonometric method

This method is based on the fact that particles will be uniformly distributed on the sphere in one axis. The proof of this is shown in Appendix B of this thesis. The uniform direction, in this case $Z$, is randomly generated on $[-R_m, R_m]$ to find the circle cross section at that point. The particle is then distributed at a random angle on the $XY$ plane due to the constraint in $Z$. The algorithm for the trigonometric method is shown in equations 6.16-6.21, and represented by figure 6.5.

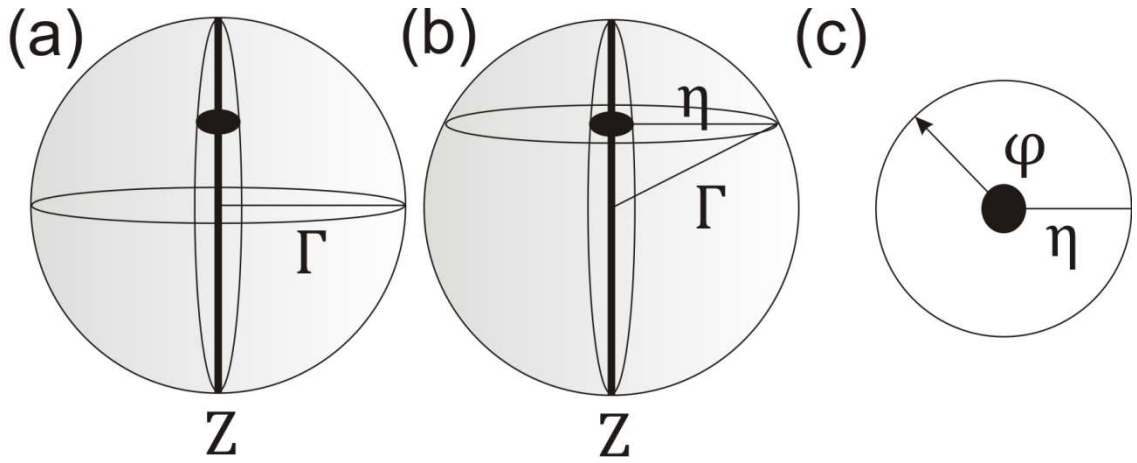



Figure 6.5. Using the trigonometric algorithm, (a) determining a Z ordinate, (b) taking the plane at that ordinate and calculating the radius $\eta$, and (c) generating a random angle $\varphi$ to find the exact $X$ and $Y$ ordinate movements.

This method requires two uniform random numbers, six multiplications, two additions, one square root and two trigonometric functions.

#### 6.3.3.5 The Two-Dimensional Rejection Method

By combining methods (iii) and (iv), a form of the trigonometric method can be devised without the use of trigonometric functions. Two random numbers are uniformly distributed along $[-1, 1]$ If the sum of squares of the two numbers is greater than 1, reject the doublet. If accepted, the ordinates are calculated based on the proof that points in a single axis are uniformly distributed:

Generate $u, v \sim 2 \times U(0, 1] - 1$ [6.29]

Let $s^2 = u^2 + v^2$; if $s > 1$, reject [6.30]

Let $a = 2\sqrt{1 - s}; k = a \times R_m$ [6.31]

$x = k \times u, y = k \times v, z = (2s - 1) \times R_m$ [6.32]

Use of this method over the trigonometric method described in 6.3.3.4 is dependent on whether the trigonometric functions of the system are slower than the combined speed of rejection and regeneration of the random numbers.

#### 6.3.3.6 The Bipyramidal Method

For completeness, the bipyramidal diffusion model [1] is also included. This method generates a random whole number between 0 and 5 inclusive, which dictates a movement in an axis, either positive or negative relative to the starting position. As a result, the one-dimensional representation of the random-walk, as shown in Figure 6.2, is

played out in each of the three dimensions, and any particles that would have ended up outside the bipyramid from other methods would now be inside the pyramid.

The method for bipyramidal diffusion is shown below:

Generate $u \sim \lfloor 6 \times U(0,1) \rfloor$ where $\lfloor x \rfloor$ is the floor function [6.33]

$$u \begin{cases} 0: x = x + R_m \\ 1: x = x - R_m \\ 2: y = y + R_m \\ 3: y = y - R_m \\ 4: z = z + R_m \\ 5: z = z - R_m \end{cases} \quad [6.34]$$

This method is technically the computationally least expensive in terms of mathematical functions, but due to equation 6.34, requires a lot of '$if$' type comparison statements and the speed of these will determine how fast the algorithm is.

#### 6.3.3.7 Speed Comparison

Each of the methods where written in C on the CPU and CUDA for the GPU. Optimisations were made where possible, and boundary conditions were removed as different simulations will have different levels of boundary interaction due to the random nature of the particle movement. Each simulation was allowed to run as many particles for 10000 time steps as it could for 10 seconds, and the speed calculated from the total number of particles moved in that time frame.

Simulations were performed on two types of CPU, representing a typical dual core lab machine, and high end computational workstation – both were tested utilising one and all available threads for calculation (where each particle is a separate thread). Two GPUs were also used, the first being used to demonstrate that even older CUDA-capable GPUs

are faster than multi-threaded high end workstations. GPU architecture is still in a state of flux in terms of features such as cache and ultimate throughput, and thus the most recent generation of NVIDIA architecture is also represented – the GTX 460 which utilises 336 cores with 48 per streaming multiprocessor, also with L1 and L2 cache.

Note that a simulation will not completely scale with the number of threads available, as certain parts of the simulation are serial and therefore cannot be parallelized. If a simulation is 20% serial and 80% parallel, splitting up the parallel section across four processors will drop the simulation time to 20 + 80/4 = 40%, or to under half the original simulation time. This is shown in Figure 6.6 for clarity.

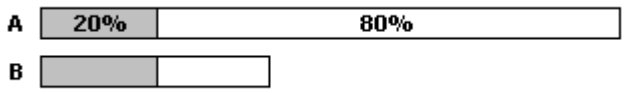



Figure 6.6 Simulation A has 20% a serial component, and an 80% parallel component. If the parallel component is split across four processors, such as Simulation B, the overall simulation time is now 20 + 80/4 = 40% of the original run time, not 25%.

Table 6.2 shows the results of the algorithms mentioned in this Appendix. There is a distinct trend whereby the cosine method is slow on older CPUs due to the inverse trigonometric function required, the normal-deviate method is slow due to the requirement of normally distributed random numbers, and the hypercube method loses speed due to rejection. Similarly, the 2D rejection method is slow due to the rejection, but not to the same degree. The trigonometric method is the fastest, as each loop will successfully generate numbers for motion; it uses uniform random number generators and reduces the number of trigonometric functions in the algorithm. Also of note is that the algorithms that involve rejection (normal-deviate, hypercube and 2D) are slower on

the GPU than algorithms that do not involve rejection. This is because the GPU works in batches of 32 threads (with multiple batches computed at once), and if one of these threads suffers from the rejection clause, the speed of the whole batch of 32 will be stunted. In the case of the GTX 460 being very slow in the hypercube and 2D rejection methods, this is due to the specific GPU architecture which deals in half-batches, so when a rejection occurs there is considerable delay in organising batch processing.

The bipyramidal method for the CPU is similar in speed to the trigonometric method due to the lack of calculations, but the magnitude of comparison statements in the method result in a slower GPU function than both the trigonometric and cosine algorithms.

**Table 6.2. Comparison of the Speed of Algorithms**

| Positions Recalculated Per Second / $10^6$ | | Cosine Method | Normal-Deviate Method | Hypercube Rejection | Trigonometric Method | 2D Rejection | Bipyramidal Method |
|---|---|---|---|---|---|---|---|
| **Athlon X2 5050e Brisbane 2.60Ghz** | CPU: 1 Thread | 6.54 | 3.07 | 4.51 | 16.34 | 10.97 | 15.75 |
| | CPU: 2 Threads | 12.74 | 6.17 | 8.89 | 32.38 | 21.73 | 30.94 |
| **Dual Xeon E5520 Nehalem 2.23 GHz** | CPU: 1 Thread | 8.24 | 3.30 | 4.44 | 25.87 | 11.46 | 25.91 |
| | CPU: 16 Thread | 96.39 | 41.69 | 67.14 | 230.63 | 157.54 | 230.84 |
| **NVIDIA Quadro 580** | GPU: 32 Cores | 704.71 | 491.16 | 318.15 | 1041.80 | 620.92 | 853.82 |
| **NVIDIA GTX 460 768MB** | GPU 336 Cores | 7291.70 | 4951.57 | 145.31 | 13602.94 | 249.78 | 6138.90 |

## 6.4 Probing Voltammetry With Small Numbers of Molecules:

### 6.4.1. Introduction

In this sub-section, random-walk simulations are used to predict voltammetry carried out on small numbers of molecules (and very low concentrations). In this introduction, we

first consider the background to these studies and then review previous work on random walk problems in electrochemistry.

Experimental research into single molecule electrochemistry has been pioneered prominently by Bard et al [38-42]. By utilising the tip of a scanning electrochemical microscope of nanometre size surrounded by a wax sheath, the positioning of the working electrode close to the counter electrode produces a small volume of low concentration solution, and allows the capture of small numbers of molecules inside the volume. Using a fixed potential to drive the electron transfer, the species in the solution can cycle between the two electrodes, as shown in Figure 6.6, which amplifies the signal by a factor of $10^7$ (assuming a typical diffusion coefficient of $5 \times 10^{-10}\ \mathrm{m}^2\ \mathrm{s}^{-1}$ and an inter-electrode distance of approximately 5 nm with a 5 nm radius nanode), resulting in voltammetry with a much larger signal to noise ratio than a single electron transfer event.

Some insight into such systems can be obtained using the continuous concentration method implicit in Fick's Laws, which is very widely applied and has been used to great effect on general voltammetric systems [16-18,22,43-48]. Fan et al.[42] used two-dimensional finite difference methods to simulate a similar system to that shown in Figure 6.7, with varying separation between the electrode and the substrate, until the wax sheath distorts the area of substrate exposed due to compression and the close proximity to the electrode.

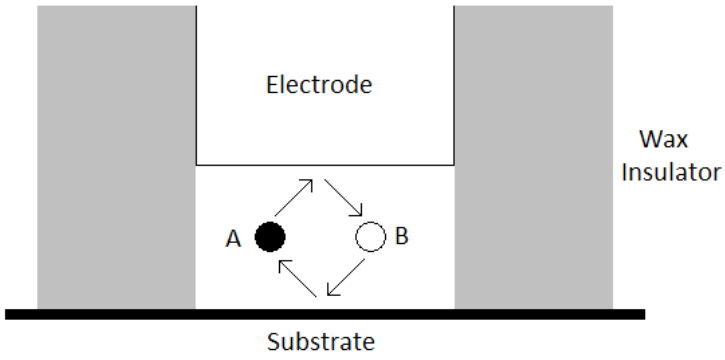



Figure 6.7 – Simplified representation of single molecule detection using scanning electrochemical microscopy. An electron is transferred at the electrode to transform Species A into species B, and vice versa at the substrate.

In this sub-section, we utilise the concept of random-walk simulations to probe voltammetric systems from the standpoint that under diffusion only conditions, a particle in solution will move an average distance per unit time, but in a randomly changing direction, until the particle interacts with a boundary. This boundary may be either a reflective boundary, such as the wax sheath 'walls' shown in Figure 6.7, or an electroactive boundary where a change of charge (i.e. electron transfer) may occur, depending on the potential of the electrode surface at that point in time. Potential-step chronoamperometry is examined, as well as generator-collector systems where two electrodes are brought close to each other, as in Figure 6.7, but without the wax-sheath reflective boundary. We next consider previous use of random-walk theory in electrochemical situations.

### 6.4.2. Theory

We first consider the transient current response at an electrode by stepping from a potential at which no current flows to a potential sufficient to drive a one electron reduction at a diffusion-limited rate:

$$A(aq) + e^- \rightleftarrows B(aq) \quad [6.35]$$

We assume the effects of migration and convection can be ignored. This system is simulated where particles are free to diffuse until they interact with a boundary condition, such as the electrode, the insulator, or the diffusion layer. This system is shown in Figure 6.1.

Secondly, we consider the case where particles are trapped inside a cylindrical space of finite volume, similar in cross-section to Figure 6.7. On the ends of the cylinder are electroactive surfaces; one is set at a potential sufficient to drive the reduction of A, and the other is set at a potential sufficient to drive the oxidation of B – together, this creates a generator-collector system, similar to that used by Fan and Bard [40].

Finally, we consider a system similar to the second case but with open, not reflective, boundaries in the $x$ and $y$ directions, which allows fresh material to diffuse from bulk to the electrode surfaces – the upper $z$ boundary still contains a collector electrode flush with an insulator surface. This is shown in Figure 6.8. Simulation is performed either with the collector electrode electroactive to drive the oxidation of B, or with it as an insulator, so no reaction occurs. This is similar to the redox cycling method by Goluch et al.[49] and Zevenbergen et al.[50], who have created dual channel electrode systems to take advantage of the different diffusion layer as compared to the cylindrical space case.

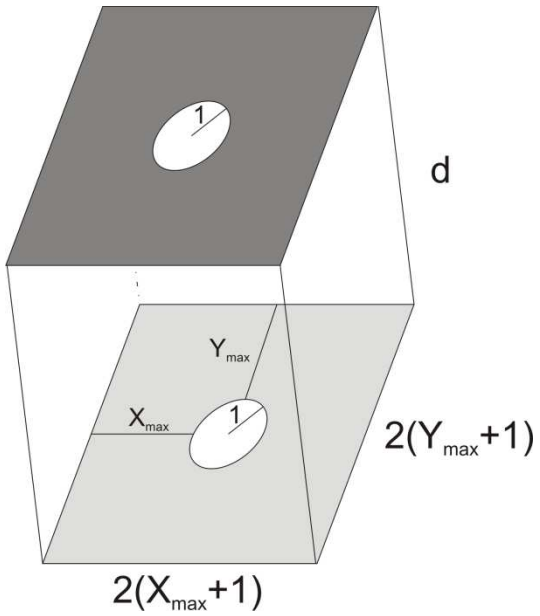



Figure 6.8. Collector-Generator experiments (see text).

### 6.4.3. Simulating Chronoamperometry

Potential step chronoamperometric random-walk simulations as outlined in Section 6.4.2 and by Figure 6.1 were prepared to test the convergence of $C_E$ and $\Delta\tau$ at a microdisk to the Shoup-Szabo equation[51], as shown in equation 6.36, when utilising the dimensionless transform in equation 6.2. This equation, derived from Fick's Laws of diffusion with the relevant boundary conditions is used in the statistical analysis of potential step chronoamperometry at microdisk electrodes, and also applies to the stochastic method given sufficiently converged simulation or appropriate adjacent-point integration as explained in chapter 6.2.7. The number of hits on the electrode surface at each time step, i.e. the dimensionless current, is given by equation 6.37.

$$f(\tau) = 0.7854 + 0.44315\tau^{-0.5} + 0.2146 \times \exp(-0.39115 \times \tau^{0.5}) \quad [6.36]$$

$$\begin{aligned} i_{\tau}^{ShSz} &= 4nFr_eD[C]\frac{\Delta t}{e} \times f(\tau) \\ &= 4nFr_eD\frac{C_E}{N_A r_e^3} \times \frac{\Delta\tau r_e^2}{eD} \times f(\tau) \\ &= 4n\frac{F}{e \times N_A} C_E \Delta\tau \times f(\tau) \\ &= 4nC_E\Delta\tau \times f(\tau) \end{aligned} \quad [6.37]$$

#### 6.4.3.1 Convergence Testing

Simulations with $C_E$ = $10^4$ were run to a $\tau_{max}$ of 2 at values of $\Delta\tau$ = $10^{-x}$, $x$ = 1, 2, 3, 4, 5, 6. The results of these simulations are shown in Figure 6.9, and show that at high values of $\Delta\tau$, the simulation underestimates the expected Shoup-Szabo value. This is consistent with the argument in chapter 6.2.6 regarding large time steps. As the time steps reduce in size, we see a trend towards the expected collision rate per time step, until the point where the expected collision rate per time step reaches single figures. Due to the quantization of electroactive events and the stochastic variation that comes from the random-walk, adjacent point integration, as outlined in chapter 6.2.7, is required to observe the trend over time. Each simulation is compared by the reduced chi-squared goodness-of-fit, $\chi^2_{\mathrm{red}}$, to the expected collision values, and these results are found in Table 6.3 (along with simulation times). The results suggest that a $\Delta\tau$ of $10^{-4}$ provides a balance between goodness of fit and simulation time.

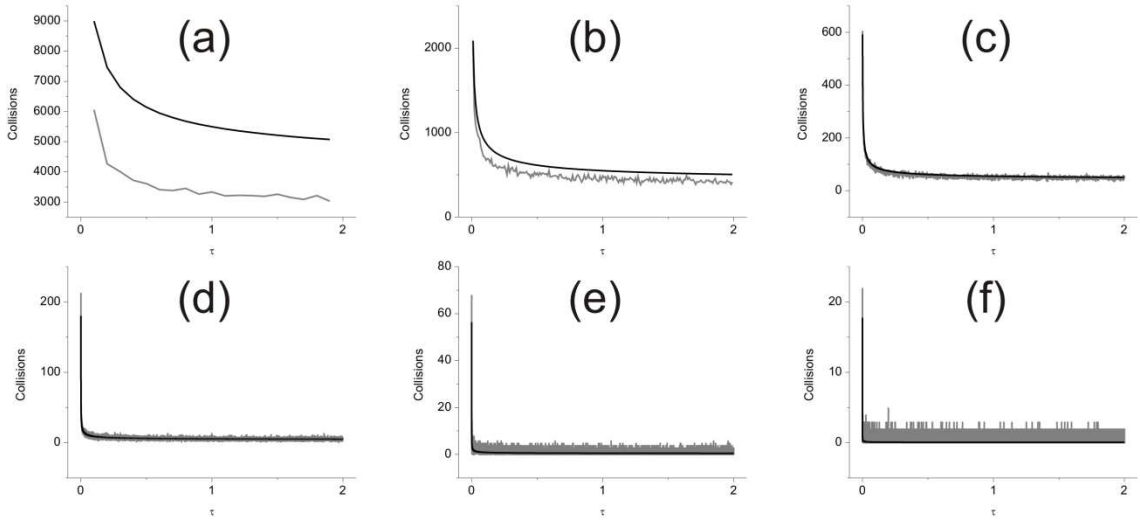



Figure 6.9. Comparison of the expected Shoup-Szabo[51] collision rate with random-walk simulations of $C_E$ = $10^4$, and $\Delta\tau$ = (a) $10^{-1}$, (b) $10^{-2}$, (c) $10^{-3}$, (d) $10^{-4}$, (e) $10^{-5}$, (f) $10^{-6}$. The grey lines represent the simulation, and the black lines show the expected Shoup-Szabo collision rate.

At the recommended $\Delta\tau$ value of $10^{-4}$, and to a $\tau_{max}$ of 2, a range of $C_E = 10^x$ were tested, where $x$ = 2, 3, 4, 5, 6. As $C_E$ increases, and more interactions with the electrode are expected, a reduction in the signal to noise ratio is observed.

**Table 6.3. Convergence Testing of $\Delta\tau$**

| $\Delta\tau$ | Reduced $\chi^2_{red}$ | Time / s |
|---|---|---|
| $10^{-1}$ | 32485.272 | 12.3 |
| $10^{-2}$ | 1014.493 | 13.3 |
| $10^{-3}$ | 59.747 | 22.7 |
| $10^{-4}$ | 6.002 | 128.6 |
| $10^{-5}$ | 0.614 | 1214.6 |
| $10^{-6}$ | 0.059 | 13437.8 |

At low $C_E$ (<$10^3$), the Shoup-Szabo steady-state predicts less than 1 electron transfer per time step, and so the quantization of the results gives some time steps with zero electron transfers, others with one interaction, and very few with more than one. At this $C_E$ level, some degree of adjacent-point integration is advised in order to accurately compare simulation and expected results. These results are shown in Figure 6.10.

Simulation times scale proportionally with the number of particles in simulation in the same simulation space and boundary conditions. At $C_E$ = $10^6$, $\Delta\tau$ = $10^{-4}$ and $\tau_{max}$ = 2, this corresponds to ~3.05×$10^9$ particles in the simulation space, each moving 20000 time steps, making ~6.1×$10^{13}$ particle movements in the simulation.

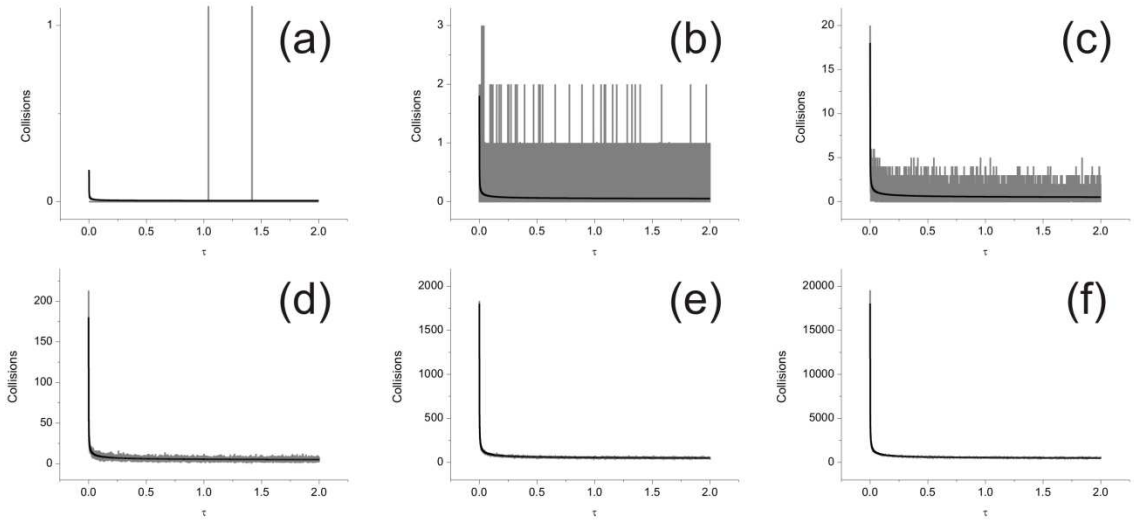



Figure 6.10. Comparison of the expected Shoup-Szabo collision rate with random-walk simulations of $\Delta\tau$ = $10^{-4}$, and $C_E$ = (a) $10^1$, (b) $10^2$, (c) $10^3$, (d) $10^4$, (e) $10^5$, (f) $10^6$. The grey lines represent the simulation, and the black lines show the expected Shoup-Szabo collision rate.

### 6.4.4 Computer Utilisation

Work in this section of the thesis was performed on a NVIDIA GTX 460 768MB GPU at 800 MHz, memory at 900 MHz, powered by an AMD Athlon X2 5050e CPU (2 threads, 2.6 GHz). CPU analyses were compared to a dual Xeon E5520 Workstation (16 threads, 2.23 GHz), using OpenMP for multithreading. Simulations were written in Microsoft Visual C++ 2008, using C/C++ for CPU and CUDA (version 3.2RC) for GPU. Random number generators were the Ranq2 (period $2^{126}$) for CPU[28], and XORWOW (period $2^{190}$) as part of the CURAND package for GPU[34,35].

### 6.4.5 Comparison with a Typical Experiment

We consider chronoamperometry for a 2 mM solution of a species with a diffusion coefficient of $5 \times 10^{-10}\ \mathrm{m^2\ s^{-1}}$ at a nanoelectrode with an 18 nm radius, corresponding to a typical experiment[41].

In our computational random-walk model, the concentration and radius relate to a simulation with a dimensionless concentration, $C_E$, of 7.024. A simulation was performed to $\tau_{max}$ value of 50 and $\Delta\tau$ = $10^{-4}$, in the absence of any finitely placed boundaries. A steady state current of ~7 pA is expected using the equation $i = 4nFDr_{\mathrm{e}}[\mathrm{C}]$. Due to the low value of $C_E$ in the simulation, the results were adjacent-point integrated at three levels: (a) 500, (b) 5000 and (c) 50000 points, to further demonstrate the effect of this integration. These simulations are shown in Figure 6.11 compared to the dimensionless Shoup-Szabo equation, along with (d), which gives the results in (c) compared to the expected steady state current.

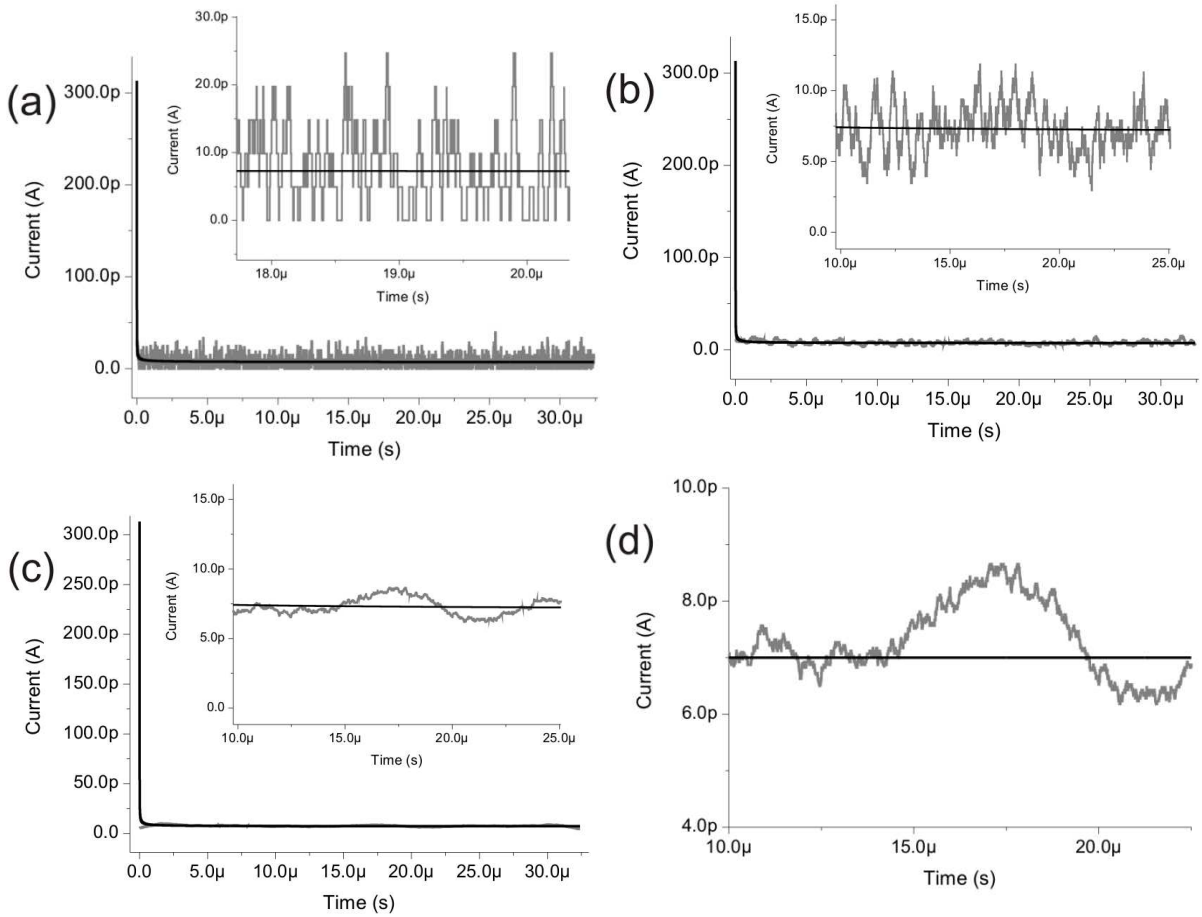



Figure 6.11. Simulation results compared to the Shoup-Szabo equation for (a) 500, (b) 5000, and (c) 50000 point integration. Graph (d) shows graph (c) compared to the steady state current expected for a concentration of 2 mM and a diffusion coefficient of $5 \times 10^{-10}$ $m^2$ $s^{-1}$.

### 6.4.6. Simulation of Voltammetry in Confined Spaces

Detection of low numbers of molecules has been shown[40] to be viable in a hollow cylinder (Figure 6.7), where the ends of the cylinder are electroactive (one being a generator electrode, and the other a collector electrode) and the walls are insulators. The diffusion in this system is thus limited to purely linear diffusion, rather than convergent diffusion as seen in open solution voltammetry. To ease a random-walk simulation, the electrodes are approximated as squares of equal area to nanoelectrodes of radius 7.5 nm. We consider particles of 2 mM concentration with a diffusion

coefficient of $5 \times 10^{-10}$ $\mathrm{m}^2$ $\mathrm{s}^{-1}$. This corresponds to a dimensionless concentration, $C_E$, of 0.508.

We consider inter-electrode gaps of (a) 10, (b) 20 and (c) 30 nm, which give dimensionless simulation heights, *d*, of 1.5, 3.0 and 4.5 respectively. The volume of the simulation space, $2 \times 2 \times \mathrm{d}$, multiplied by $C_E$, corresponds to a total number of molecules in the simulation space of (a) 3, (b) 6 and (c) 9. For such experiments, the key question is that given a small inter-electrode gap and volume, which will result in a larger current: the cycling of few species between the electrodes, or more particles being present due to the larger volume?

Random-walk simulations were performed using the parameters mentioned above in an enclosed volume, to a $\tau_{max}$ of 200 and a $\Delta t$ of $10^{-4}$. Results were given a $5 \times 10^5$ point integration, giving a resolution of $\Delta t$ = 5.625 μs. Due to the generator-collector nature of the closed system, particles were initially assigned a random status of A or B at the beginning of the simulation, giving the initial results as steady state.

As shown in Figure 6.12, as the inter-electrode spacing decreases, and thus the number of particles in simulation decreases, the current increases. This indicates that, as expected, in an ideal system, the effect on the current of cycling of particles in a small volume is greater than that of having larger numbers of particles in a larger volume at the same concentration.

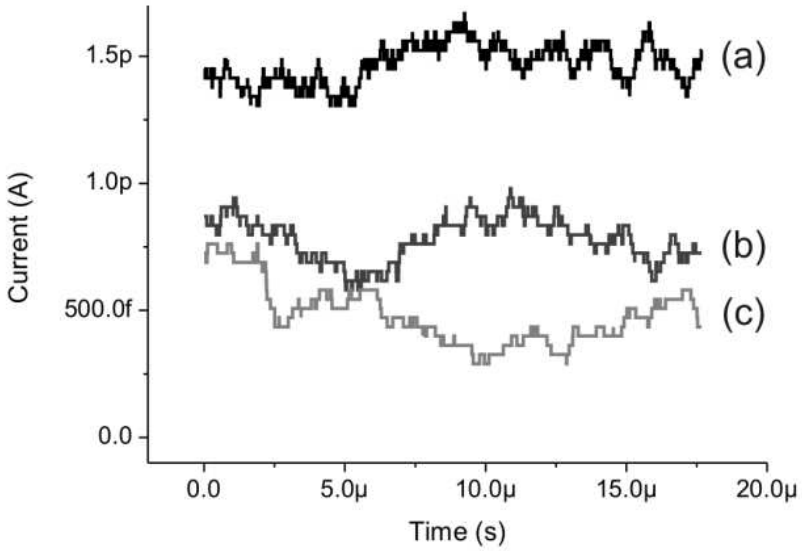



Figure 6.12 Low volume dual-electrode simulations of $C_E$ = 0.508 at an inter-electrode spacing of (a) 1.5, (b) 3.0 and (c) 4.5.

However, even with these results, the steady state chronoamperometric current is still less than that expected at an open-volume normal microdisk. This is due to the lack of convergent diffusion in the closed volume situation.

### 6.4.7 Simulation of a Collector-Generator System of Two Opposite Microdisk Electrodes

As in Figure 6.8, we consider a system where two microelectrodes flush with large insulators approach each other. This gives a simulation where the axial direction is limited, due to the second electrode and insulator, but in other directions, particles are free to diffuse material from bulk solution. This is similar to removing the insulator boundaries from the simulations in chapter 6.4.6, and setting them at the semi-infinite diffusion limit as in chapter 6.2.2. In this section, we will call the stationary electrode the generator electrode, where $A \rightarrow B$ occurs, and the electrode whose distance will vary from the generator electrode will be called the collector electrode, where $B \rightarrow A$ occurs.

We consider the collector electrode at various distances from the generator electrode, and in situations where the collector electrode is 'off', and no current flows. At large distances between the electrodes, the effect of the collector electrode and the upper insulator surface should become negligible.

Simulations were performed at a dimensionless distance, *d,* equal to (i) 0.10, (ii) 0.25, (iii) 0.50, (iv) 1.00, (v) 2.00, (vi) 3.00, (vii) 5.00, (viii) 10.00, and (ix) 20.00, and under the low concentrations of section 4, with a $\tau_{max}$of 200 and a $\Delta\tau$ of $10^{-4}$, and with the collector electrode both off and on.

With the collector electrode in the 'off' mode, and as the distance between the electrodes increases, an increase in current should be observed - the edge of the diffusion layer will take longer to hit the upper surface, and thus increases the volume of the diffusion layer compared to a shorter distance between the electrodes. With the collector electrode on, at ultra-low distances between the electrodes, when an individual particle approaches the electrode, it can alternate between the two electrodes, and then diffuse away. If more particles are between the electrodes at one point in time, the current will be higher, but overall, peaks of current should be observed. At larger inter-electrode gaps, the likelihood that a particle will diffuse from the generator to the collector and back again reduces, and thus feedback reduces, but since more particles are between the electrodes, a higher average current results as the diffusion layer extent increases.

Figure 6.13a shows the current with the collector electrode switched off, and shows that as *d* increases, the average currents increase up to the expected Shoup-Szabo values since the second surface has progressively less influence over the electrode – the average

currents and standard deviations are given in Table 6.4. Figure 6.13b shows the current with the collector electrode on – at low inter-electrode distances, large peaked currents are seen signifying particles rapidly diffusing between the electrodes in the small space. As the distance increases, the effect of the collector electrode becomes less significant, and the current drops below the expected Shoup-Szabo values, and when $d$ > 1.00, the current increases again, as with the results shown in Figure 6.13a.

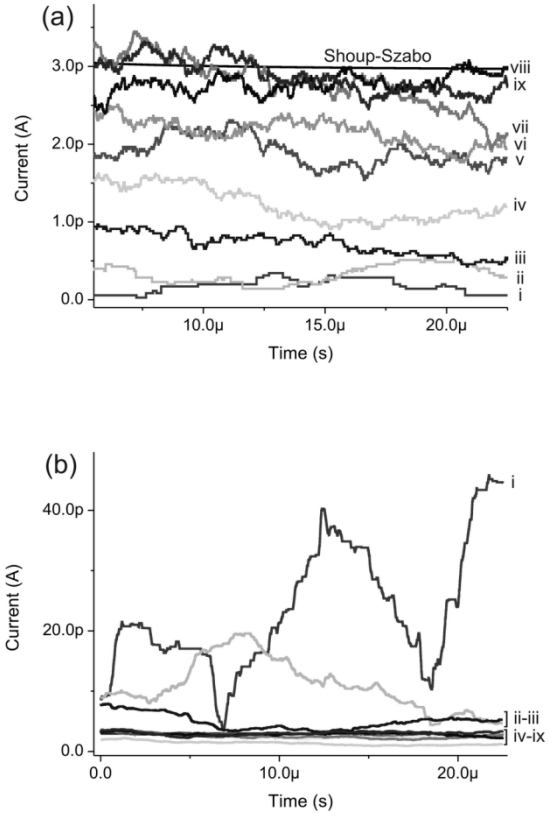



Figure 6.13 Simulations as the collector electrode approaches the generator, with the collector (a) on, (b) off, where d = (i) 0.10, (ii) 0.25, (iii) 0.50, (iv) 1.00, (v) 2.00, (vi) 3.00, (vii) 5.00, (viii) 10.00, and (ix) 20.00.

Comparing the results of chapters 6.4.6 and 6.4.7, it is inferred that having two electrodes close to each other in generator-collector mode on large insulating planes will give larger current than the system in Figure 6.7, even at low concentrations.

**Table 6.4. Average Currents from Figure 6.13, with Standard Deviations.**

| **d** | Figure 9a | | Figure 9b | |
|---|---|---|---|---|
| | $i_{average}$ **/ pA** | $\sigma$ **/ pA** | $i_{average}$ **/ pA** | $\sigma$ **/ pA** |
| 0.10 | 0.19 | 0.11 | 23.30 | 10.06 |
| 0.25 | 0.39 | 0.17 | 10.98 | 4.21 |
| 0.50 | 0.76 | 0.15 | 4.91 | 1.27 |
| 1.00 | 1.37 | 0.34 | 1.37 | 0.34 |
| 2.00 | 2.03 | 0.31 | 2.65 | 0.26 |
| 3.00 | 2.32 | 0.33 | 2.56 | 0.37 |
| 5.00 | 2.92 | 0.36 | 2.54 | 0.43 |
| 10.00 | 2.91 | 0.31 | 2.93 | 0.29 |
| 20.00 | 2.95 | 0.26 | 3.01 | 0.30 |
| $i_{SS}$ | 2.89 | - | 2.89 | - |

### 6.4.8 Conclusions

In this work, we have provided a framework for the simulation of random-walk voltammetry. From first principles, the simulation space, boundary conditions and movement algorithm have been investigated. Analysis of open volume potential-step chronoamperometry utilising the random-walk approach was used for convergence, and allowed the simulation of stochastic vs. statistical responses. A voltammetric trap, the hollow cylinder, was simulated using the random-walk method, providing insight into low-volume collector-generator methods. Finally, the case of two opposite planes each containing a microdisk electrode was compared against the hollow cylinder, and it was found that at small distances between planes, this method will produce periods of higher current than the cylinder geometry.

## 6.5 How Many Molecules Are Required To Measure A Cyclic Voltammogram?

### 6.5.1. Introduction

Cyclic Voltammetry is a widely used electrochemical technique for determining reductive and oxidative electron transfer mechanisms [52-54], measuring concentrations in the form of sensors for gases [55-57] and species dissolved in solution [55,58-60], observations of reactive intermediate species [61], determination of reaction rate constants and thermodynamic properties [62,63], measurement of formal potentials[64-69], and comparisons of surface and catalyst performance [70-75]. The voltammetric measurement is determined by the potential applied to the electroactive surface, the formal potential of the electron transfer of the species in question, the rate at which that electron transfer occurs, the solution composition, the rate of transport by diffusion to the electrode and the identity of the electroactive surface itself. Upon electron transfer, a change in current is observed and measured - a typical cyclic voltammogram will sweep from the potential from one value to another then back to the starting potential, encompassing the value of the formal potential, whilst recording the current as a function of the potential applied.

Each electron transfer to a molecule is a quantized event, transferring a small amount of charge (an integral multiple of $1.6\times10^{-19}$ C). At high concentrations and/or large time windows of observation (such that the time between readings on voltammetric equipment is relatively short compared to the timescale of the experiment), a voltammetric reading will appear smooth as the quantized events will average. In this work, we examine the idealised theoretical limits at which fluctuations from the random diffusional motion of species in solution will be reflected in the voltammetric readings.

This is of importance to the experimentalist when dealing with cyclic voltammetry at low concentrations of electroactive species assuming idealised equipment with low equipment/electrical noise; in practice specialist equipment is needed to probe quantised limits [76-78]. The work is of importance also to nanoelectroanalysis, specifically electrodes that are modified with nanoparticle arrays [79] since these electrodes are more sensitive than larger ones in respect of diffusional fluctuations.

This chapter has previously outlined [80] a general approach for the Monte Carlo simulation of electrochemical processes in terms of diffusion-only Brownian motion, whereby a particle/species moves randomly until it hits the electrode. In our previous work electron transfer was assumed to take place on every collision with the electrode and in this way diffusion controlled chronoamperometric transients were simulated. In this sub-section, we now assume the case where electron transfer takes place based on the potential at the electrode and the formal potential of the electron transfer species and the rate of electron transfer [81] – in this way, the voltammetric experiment can be simulated. Every particle in the solution moves at random and at a speed proportional to its diffusion coefficient.

In this sub-section, the voltammetric Random-Walk methodology as described in section 6.2 is used to simulate the stochastic motion of species in solution under sweeping voltammetric conditions, and is explained briefly in the next section.

### 6.5.2. Theoretical Model

We consider the simple one-electron redox couple, shown in equation 6.38.

$$A \overset{k^0}{\rightleftharpoons} B + e \qquad [6.38]$$

where $k^0$ is the standard electrochemical rate constant of the $A/B$ redox couple. In cyclic voltammetry, the potential, $E$, will sweep from a starting potential $E_{start}$ to $E_{max}$ at a constant scan rate $v$ over time $t$, as shown in equation 6.39, and then return to the start.

Forward Sweep: $E = E_{start} + vt; E < E_{max}$ [6.39a]

Reverse Sweep: $E = E_{max} - vt; E > E_{start}$ [6.39b]

The rate of transition of species A into species B is first determined by the potential at the surface of the electrode, the rate constant, and the kinetic model applied employed, and second by the surface concentrations of A and B which are controlled by diffusion. Both Butler-Volmer and Marcus-Hush kinetics are discussed to develop such schema with respect to random-walk type simulations. In the results presented later, fast electrode kinetics will be assumed corresponding to full electrochemical reversibility; both the Butler-Volmer and Marcus-Hush expressions are seen to give the same results in this fast kinetic limit. A comparison of the two types of electrode kinetics will be considered subsequently.

#### 6.5.2.1 Butler-Volmer Kinetics

As described in chapter 1.3.2.2, Butler-Volmer kinetics assume that at a high over-potential, conversion from A to B is absolute, and the rate constant at a given potential is exponentially proportional to both the potential applied and the transfer coefficient, $\alpha$, which describes the transition state in terms of its likeness to either A or B [82,83]. This is shown in equation 6.40 for a one electron process [84].

$$i = FAk^0 \left\{ [A]_0 exp\left[\frac{(1-\alpha)F}{RT}(E - E^0)\right] - [B]_0 exp\left[\frac{-\alpha F}{RT}(E - E^0)\right]\right\} \quad [6.40]$$

#### 6.5.2.2 Marcus-Hush Kinetics

Marcus-Hush kinetics take into account the Gibbs energy of activation and reorganisation energy such that the transition state between A + B (and the transfer coefficient, $\alpha$) can be altered by the potential applied depending on the formal rate constant of the electron transfer, leading to incomplete electron transfer even at high over-potential [81,84-88]. This is shown in equation 6.41a, where $\Delta G_i^{\ddagger}$ is the activation energy from the distortion of the inner coordination shell, $\Delta G_0^{\ddagger}$ is the rearrangement energy from solvent dipoles between the initial state and the transition state, and $\lambda$ is the 'reorganisation energy'. Equation 6.41b correlates the transfer coefficient of Butler-Volmer kinetics to the Gibbs energy and reorganisation energy terms, and thus becomes potential dependant.

$$\Delta G^{\ddagger} = \Delta G_i^{\ddagger} + \Delta G_0^{\ddagger} = \frac{\lambda}{4}\left(1 + \frac{\Delta G}{\lambda}\right)^2 \quad [6.41a]$$

$$\alpha = \frac{1}{2}\left(1 + \frac{\Delta G}{\lambda}\right) \quad [6.41b]$$

### 6.5.3 Dimensionless Units and Simulation Space

Using a system of dimensionless units, as shown in table 6.5 where $e$ is the electronic charge ($1.6\times10^{-19}$ C) and $E^0$ is the standard formal potential of the electroactive couple, simulation scaling factors are removed. The time of simulation is determined by the potential window of the scan divided by the scan rate:

$$\tau_{max} = 2\left(\frac{\theta_{end}-\theta_{start}}{\sigma}\right) \qquad [6.42]$$

**Table 6.5. Dimensionless Quantities**

| Unit | Dimensional | Dimensionless |
|---|---|---|
| Potential | E | $\theta = \frac{F}{RT}(E - E^0)$ |
| Time | t | $\tau = \frac{D_A t}{r_e^2}$ |
| Concentration | [c] | $C_E = [c]/N_A r_e^3$ |
| Scan Rate | $v$ | $\sigma = \frac{F}{RT}\frac{r_e^2}{D_A}v$ |
| Rate Constant | $k^0$ | $K^0 = \frac{k^0 r_e}{D_A}$ |
| Length | $x$ | $X = x/r_e$ |
| | $y$ | $Y = y/r_e$ |
| | $z$ | $Z = z/r_e$ |
| Current | $i$ | $j = \frac{i\Delta t}{e}$ |

The simulation space is defined by a flat microdisk electrode of dimensionless radius 1, flush against an insulator surface on the $Z = 0$ plane, with boundaries in the $X$, $Y$ and $Z$

dimensions defined by Gavaghan [26], at a distance $6\sqrt{\tau_{max}}$ beyond the electrode. This is represented in Figure 6.14.

The $X$ and $Y$ boundaries are simulated as periodic boundaries, whereby if a particle goes beyond the boundary it will reappear on the alternate side in the same state. This is similar to the modelling of an array, but as the boundaries are set sufficiently beyond the diffusion layer of the electrode, the effect on the voltammetry is negligible. When a particle interacts with the top Z boundary it is reinserted back into the simulation space at a random $XY$ position along the Z boundary to simulate the zero-flux boundary. When a particle interacts with the insulator, it is reflected back into solution - if the particle interacts with the electrode, the probability of electron transfer (and thus a change of state for the particle) is determined.

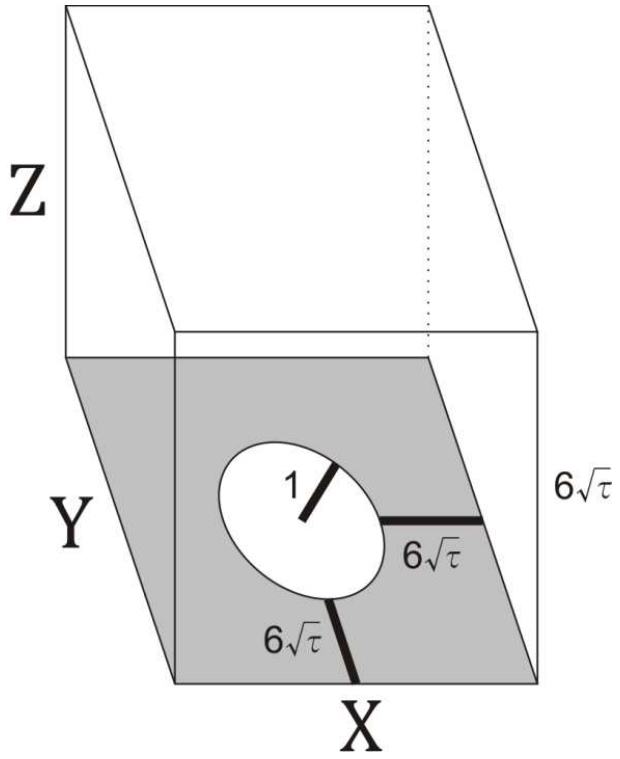



Figure 6.14. Representation of Simulation Space.

### 6.5.4 Implementation of the Electrode Kinetics

It has been shown previously for random-walk simulations [2] that kinetics at the electrode can be modelled in terms of a probability factor related to the potential, and at the Nernst limit of infinitely-fast kinetics (where $k^0 \to \infty$), as shown in equation 6.43.

$$P_{red} = \frac{1}{1+e^{\theta}} \qquad P_{ox} = \frac{1}{1+e^{-\theta}} \qquad [6.43]$$

where $P_{ox}$ is the probability, $0 \leq P_{ox} \leq 1$ , of successful oxidation from A to B and $P_{red}$ is the corresponding probability for successful reduction of B to A. To implement these probabilities, a random number between 0 and 1 is drawn at the time of interaction and compared to the value of $P_{ox}$ or $P_{red}$ depending on the current state of particle in the simulation (if it is currently A or B).

In terms of more complicated kinetic paradigms, such as those of Bulter-Volmer or Marcus-Hush, Feldberg [89] has shown that through the Oldman-Zoski approximation [90], the kinetics can be formalised in terms of probability for a disk electrode.

More specifically,

$$P_{red} = \frac{\kappa_{red}(Я_{red}+3\pi)}{Я_{red}(Я_{red}+3\pi+2)+12} \quad \text{and } P_{ox} = \frac{\kappa_{ox}(Я_{ox}+3\pi)}{Я_{ox}(Я_{ox}+3\pi+2)+12} \qquad [6.44]$$

where,

$$Я_{red} = \kappa_{red}\theta_{red} \quad \text{and } Я_{ox} = \kappa_{ox}\theta_{ox} \qquad [6.45]$$

$$\theta_{red} = 1 + \frac{D_A}{D_B}\exp(\theta) \quad \text{and } \theta_{ox} = 1 + \frac{D_B}{D_A}\exp(-\theta) \qquad [6.46]$$

$$\kappa_{red} = \gamma_{red}^{het}K_{red}^0 \quad \text{and } \kappa_{ox} = \gamma_{ox}^{het}K_{ox}^0 \qquad [6.47]$$

$\gamma^{het}$ represents the factors associated with either the Butler-Volmer representation of heterogeneous kinetics at the electrode surface, $\gamma^{BV}$, or the Marcus-Hush equivalents, $\gamma^{MH}$. The Bulter-Volmer expressions are defined by potential and the transfer coefficient between the reduced and oxidised species, $\alpha$:

$$\gamma_{red}^{BV} = \exp[-\alpha\theta] \qquad \text{and } \gamma_{ox}^{BV} = \exp[(1-\alpha)\theta] \qquad [6.48]$$

The Marcus-Hush formalism is defined in terms of the reorganisation energy, $\lambda$ (where $\lambda/4 = \Delta G^{\neq}$ when $\Delta G^{0} = 0$ where $\Delta G^{\neq}$ is the free energy of activation):

$$\gamma_{red}^{MH} = \exp\left[-\frac{\theta}{2}\right]\frac{\int_{-\infty}^{\infty}\frac{\exp\left[\frac{-(\varepsilon^*-\theta)^2}{4\lambda^*}\right]}{2\cosh\left[\frac{\varepsilon^*}{2}\right]}d\varepsilon}{\int_{-\infty}^{\infty}\frac{\exp\left[\frac{-(\varepsilon^*)^2}{4\lambda^*}\right]}{2\cosh\left[\frac{\varepsilon^*}{2}\right]}d\varepsilon} \qquad \text{and } \gamma_{ox}^{MH} = \exp\left[\frac{\theta}{2}\right]\frac{\int_{-\infty}^{\infty}\frac{\exp\left[\frac{-(\varepsilon^*-\theta)^2}{4\lambda^*}\right]}{2\cosh\left[\frac{\varepsilon^*}{2}\right]}d\varepsilon}{\int_{-\infty}^{\infty}\frac{\exp\left[\frac{-(\varepsilon^*)^2}{4\lambda^*}\right]}{2\cosh\left[\frac{\varepsilon^*}{2}\right]}d\varepsilon} \qquad [6.49]$$

$$\lambda^* = \frac{\lambda}{k_B T} \qquad \varepsilon^* = \frac{\varepsilon}{k_B T} \qquad [6.50]$$

where $\varepsilon$ is an integration variable [91]. Equation 6.44 has been modified from [89] to reduce computational complexity. While the uninviting integrals of equation 6.49 have been evaluated and analysed by Oldham [92] into a series of summations, the integrals are readily and quickly solvable [93,94] by modern computational hardware via trapezium rule - two million combinations of $\theta$ and $\lambda$ taking less than a ten seconds on our GPU.

### 6.5.5 Computer Utilisation

Due to the highly independent nature of the particles in this simulation, the relatively novel implementation of graphic processors has been utilised in this work to increase speed over by a minimum factor of 20 over a 16-thread computer workstation. This method has been shown to successfully aid in heavily parallel systems such as random-

walk [80] and explicit finite-difference [16,17,95]. Work in this section of the thesis was performed on a NVIDIA GTX 460 768MB GPU with 336 CUDA cores at 850 MHz, powered by an AMD Athlon X2 5050e CPU (2 threads, 2.6 GHz). Code was written in Microsoft Visual C++ 2008 Express with CUDA extensions. Speeds of 6.5 to 7.1 billion particle movements per second were observed. The XORWOW random number generator (period $2^{190}$) as part of the CURAND package for GPU[35] was used.

### 6.5.6 Computation of Voltammetry

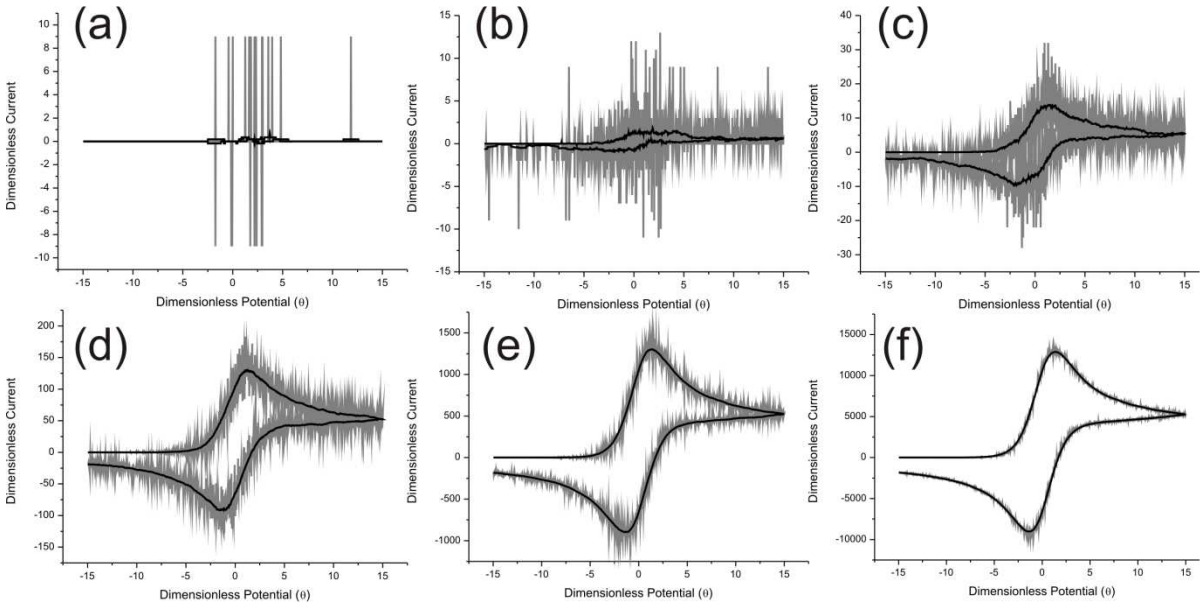



Figure 6.15. Effect of concentration for $K^0$= $10^4$, $\sigma$ = $10^3$, at $C_E$ = (a) $10^2$, (b) $10^3$, (c) $10^4$, (d) $10^5$, (e) $10^6$ and (f) $10^7$. Black lines are the adjacent-point integrated ($\Lambda_s$= 50) results.

For a range of concentrations, a range of scan rates were simulated to show typical voltammetry at these conditions. Ultimately, the concentration and the number of time steps determine the quantised fluctuations seen in the voltammetry – with a reasonable 'signal-to-noise level', the voltammetry can be adjacent point integrated as explained in [80] and in chapter 6.2.7. Figure 6.15 shows the voltammetry observed at $\sigma = 10^3$ for

dimensionless concentrations, $C_E$, between $10^2$ and $10^7$, where due to the increase in the concentration more interactions occur on the electrode surface per unit time.

Initially the quantization of the electron transfer events does not produce a discernable voltammetric response, even with adjacent point integration, $\Lambda_s$. At the point between $C_E$ = $10^3$ and $C_E$ = $10^4$, with adjacent point integration, a 'readable' voltammetric shape appears, and at $C_E$ = $10^7$, a non-smoothed result is produced which clearly shows the expected voltammetric wave shape.

#### 6.5.6.1 Comparison of Random Walk with Finite-Difference Simulations

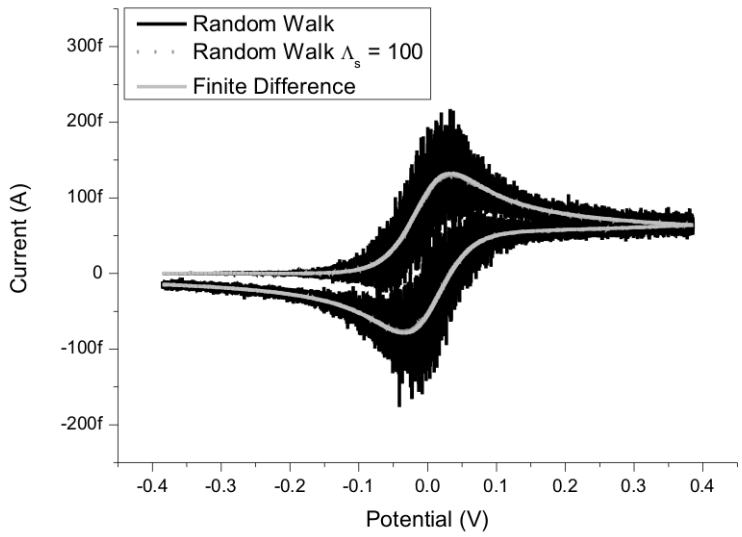



Figure 6.16. Comparison on Marcus-Hush finite difference simulation with Marcus-Hush random walk simulation. The adjacent-point integration is located on top of the finite difference result.

Simulations to compare the random-walk concept to cyclic voltammetry with implicit finite difference simulations [93,96] were performed. Using Marcus-Hush kinetics formulated in chapter 6.54, simulations of $\lambda^*$ = 40, $C_E$ = $10^5$, $K^0$= $10^4$ and $\sigma$ = $10^2$ were

calculated, with $\Lambda_S$ = 100. Results, as shown in Figure 6.16, correspond to to an electrode of 100 micron radius, diffusion coefficient of $5\times10^{-10}$ $m^2$ $s^{-1}$, a scan rate of 128.5 mV $s^{-1}$ and a concentration of 1.66 nM. The agreement of the peak current of the random-walk simulation is to within 4.7% of the finite difference. A factor 10 decrease in the time step increases the agreement to 1.3%, for a 10-fold increase in simulation time.

### 6.5.7 Results and Discussion

For a given system, it is expected that as a result of the increased number of collisions of A with the electrode, that as the scan rate decreases if the time step is constant between simulations, a smoother current should be observed. This will result in a increased accuracy in determining the voltammetric wave shape which would be otherwise lost in the noise. Also, it would be expected that voltammetry would be easier to decipher with a higher concentration present in solution. Based on these expected observations, Butler-Volmer simulations were performed to probe the limit at which the stochastic motion of a small number of molecules results in 'unreadable' voltammetry, even when adjacent-point integration is applied. Figure 6.17 shows simulations for $K^0$= $10^4$, $\alpha$ = 0.5, $\sigma$ = $10^2$ to $10^4$, $C_E$ = $10^3$ to $10^5$, $\Delta\tau$ = $10^{-4}$ and appropriate adjacent point integration.

The current-time behaviour seen at low concentrations ($C_E$ = $10^3$) and high scan rates ($\sigma$ = $10^3$) of Figure 6.17 gives somewhat undecipherable voltammetric readings with no discernable peak potential or peak current reading. However it is clear that as concentration increases, the voltammetric signal emerges from the noise. As the scan rate decreases, a similar effect is seen also, for example when $C_E$ = $10^4$ shifts from low to high scan rates.

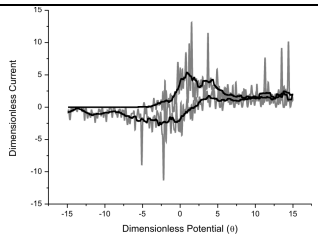
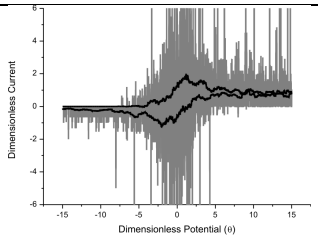
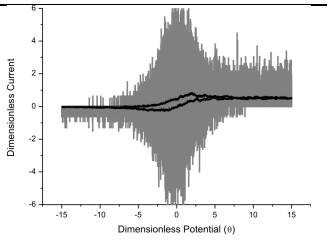
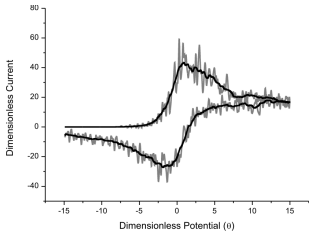
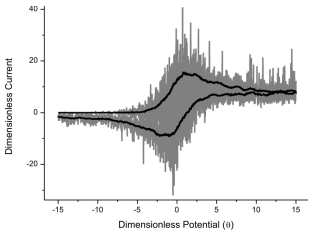
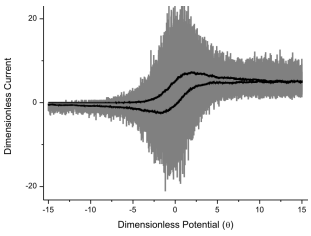
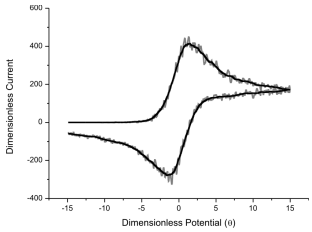
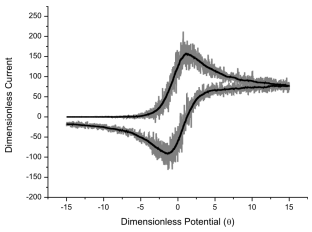
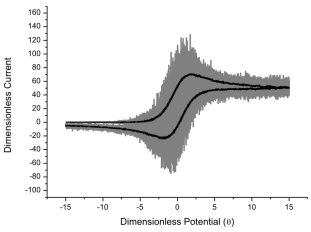
| $\Delta\tau = 10^{-4}$<br>$K^0 = 10^4$ | $\sigma = 10^3$<br>$\Lambda_s = 15$ | $\sigma = 10^2$<br>$\Lambda_s = 100$ | $\sigma = 10^1$<br>$\Lambda_s = 500$ |
|---|---|---|---|
| $C_E = 10^3$ |  |  |  |
| $C_E = 10^4$ |  |  |  |
| $C_E = 10^5$ |  |  |  |

Figure 6.17. Simulations to probe the limits of cyclic voltammetry for a fully reversible reaction.

In a real experimental context, these simulation values were compared against statistical finite difference simulations and transformed to relate to an electrode of radius 100 μm, and a diffusion coefficient of both species A and B of $5\times10^{-10}$ $m^2$ $s^{-1}$. These results are shown in Figure 6.18, where the concentration from 1.66 pM and 166 pM is probed, at accessible experimental scan rates between 1.285 V $s^{-1}$ and 12.85 mV $s^{-1}$. At these values of concentration and scan rate, the current response expected is in the atto-amp region ($\sim10^{-18}$ A), and is well below typical voltammetric equipment sensitivity.

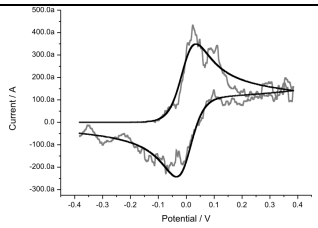
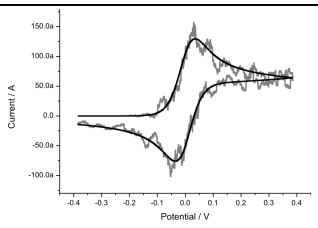
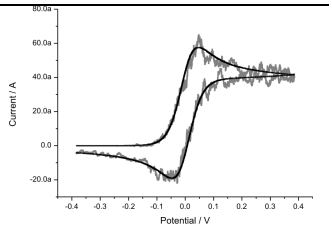
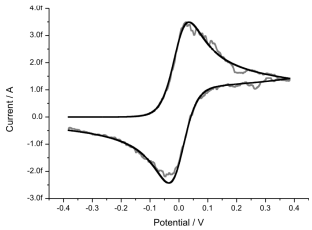
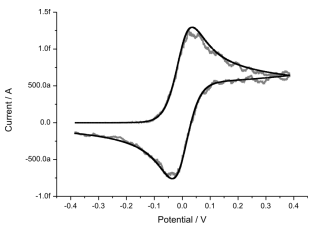
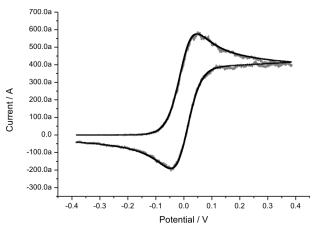
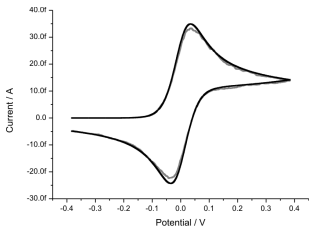
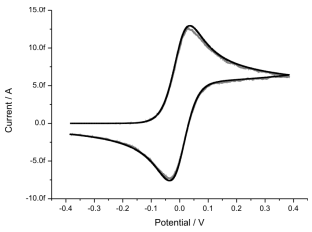
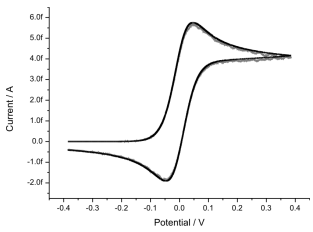
| $\Delta t$ = 2 ms<br>$r_e$ = 100 μm | $v$ = 1.285 V s$^{-1}$ | $v$ = 0.1285 V s$^{-1}$ | $v$ = 0.01285 V s$^{-1}$ |
|---|---|---|---|
| [c] = 1.66 pM |  |  |  |
| [c] = 16.6 pM |  |  |  |
| [c] = 166 pM |  |  |  |

Figure 6.18. Comparison of random walk simulations with finite difference simulation results with real-world dimensions applied.

From Figure 6.18, the comparison between the finite difference and the adjacent-point integrated random-walk simulations provides insight into the stochastic limit of fully reversible cyclic voltammetry. By comparing each peak potential, it is clear that for $v$ = 1.285 V s$^{-1}$, [c] = 1.66 pM is inaccurate – [c] = 16.6 pM has a distinct voltammetric wave shape, but stochastic motion could result in inaccurate peak potential being reported. In terms of the lowest concentration probed, all three scan rates probed perform poorly, however $v$ = 0.01285 V s$^{-1}$ could arguably be interpreted accurately.

Equation 6.23 can be reapplied between the random-walk and finite difference simulations in the context of Figure 6.18 to give Figure 6.19, which shows that the

concentration has a the bigger effect on voltammetric wave interpretation over scan rate. For a correlation within 0.5%, the two highest scan rates at 1.66 pM fall below this mark. For a correlation within 0.1%, the lowest concentration over all the scan rates probed and the two lowest scan rates for the 16.6 pM concentration do not fall within this mark.

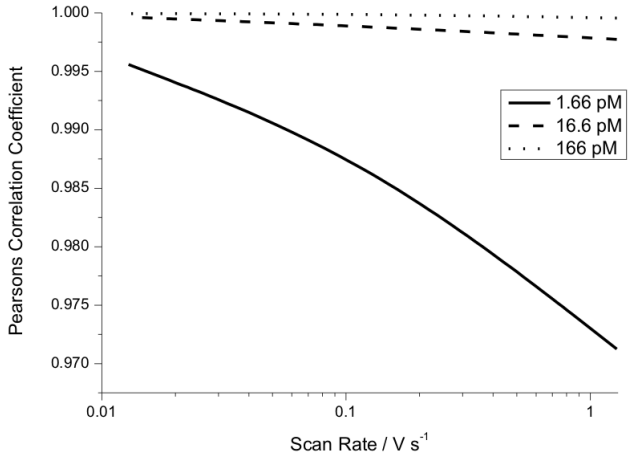



Figure 6.19. Correlation between random-walk and finite difference simulations from Figure 6.18.

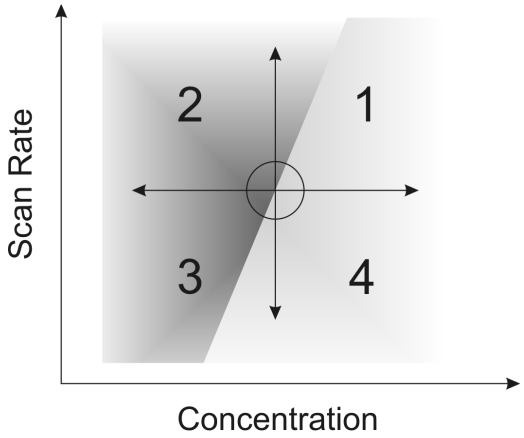



Figure 6.20. Quadrants around the stochastic focal point.

From this interpretation, we could consider, as shown in Figure 6.20, that the dimensionless quantities of $C_E$ = $10^3$ and $\sigma$ = 10 are a focal point – at this level, readable voltammetry is performed. In quadrant 1, thus at higher concentrations, a scan rate of

many decades higher is required to exceed the stochastic limit, but at lower concentrations or higher scan rates, as seen in quadrant 2, the stochastic limit will be surpassed and no readable voltammetry will be produced. Quadrant 3 represents that for each order of magnitude of lower concentration, a sufficiently lower scan rate of a several decades will be required to produce voltammetry, whereas in quadrant 4, all higher concentrations and lower scan rates will produce readable voltammetry.

Table 6.6 shows a list of dimensional values for this focal point for a range of electrode sizes, for a diffusion coefficient of $5\times10^{-10}$ $m^2$ $s^{-1}$.

**Table 6.6. Dimensional Values for $C_E$ = $10^3$ and $\sigma$ = 10 at D = $5\times10^{-10}$ $m^2$ $s^{-1}$.**

| Radius | Scan Rate | Concentration |
|---|---|---|
| 1mm | 128.5 μV $s^{-1}$ | 1.66 fM |
| 100μm | 12.85 mV $s^{-1}$ | 1.66 pM |
| 10μm | 1.285 V $s^{-1}$ | 1.66 nM |
| 1μm | 128.5 V $s^{-1}$ | 1.66 μM |
| 100nm | 12.85 kV $s^{-1}$ | 1.66 mM |
| 10nm | 1.285 MV $s^{-1}$ | 1.66 M |

At the focal point, for a dimensionless concentration of $C_E$ = $10^3$ and diffusion coefficient of of $5\times10^{-10}$ $m^2$ $s^{-1}$, an average of 19 particles (standard deviation of 4) will hit the electrode every 2 ms, of which at the peak potential an average of 0.8 or less per 2 ms is required for 'readable' voltammetry (relating to $\sigma$ = $10^1$). For a dimensionless concentration of $C_E$ = $10^4$, an average of 193 particles (standard deviation of 14) will hit

the electrode every 2 ms, of which at the peak potential an average of 40 or less per 2 ms is required for 'readable' voltammetry (relating to $\sigma = 10^3$).

### 6.5.8 Conclusions

We have shown through random-walk simulation for fully reversible cyclic voltammetry, the stochastic limit for discernable voltammetry can be determined in terms of concentration, scan rate, electrode radius and diffusion coefficient. The need for accurate adjacent-point integration for samples with high signal-to-noise ratios has been examined. Through direct comparison of finite difference and random walk simulations, a lower limit of concentrations coupled to scan rates for various sizes of microdisk electrodes has been established, and quantified for defining acceptable discernable voltammetry in experimental procedure.

## 6.6 Nanoparticle-Electrode Collision Processes: Investigating the Contact Time Required for the Underpotential Deposition (UPD) of Thallium on Silver Nanoparticles in Aqueous Solution

Special note should be given to The experiments in this chapter were conducted by Neil Rees and YiGe Zhou.

### 6.6.1 Introduction

The electrochemical study of particle impacts has been an active research area for less than twenty years [97-108], and there has been considerable interest recently into collisions of nanoparticles with electrodes. Initially, the focus was on reactions occurring on the surface of the nanoparticles (NPs) during the contact with the electrode [109-113], including the electrocatalytic hydrogen evolution reaction on Pt NPs [109]. Most recently,

however, the direct electrochemistry of NPs during collisions with electrodes has been studied, including the exhaustive oxidation of AgNPs as a means to determine their size [114], follow their aggregation kinetics [115], and measure the sticking probability of impacts with electrodes [116]. Of particular interest has been the observation that monolayer underpotential deposition (UPD) and multilayer bulk deposition are possible during NP-electrode collisions [117,118].

In the recent studies into thallium deposition onto AgNPs [117,118], AgNPs of radius 45nm were allowed to collide with a glassy carbon (GC) microelectrode (radius 11$\mu$m) in the presence of 10mM $TlNO_3$. Careful potentiostatting of the electrode enabled control of the number of layers of thallium that were deposited: ranging from an average coverage of ca. one layer at $E$ = -0.50 V (vs. Ag/AgCl) to ca. 19 layers when $E$ = -0.73 V.

However, earlier research into ultrasonically-driver microparticle electrode collisions, no evidence of faradaic charge transfer was found [11,12]. This contradiction has been suggested and might be explained in terms of both the particle size and the NP-electrode contact time: in the NP-electrode collisions, contact time is in the range of 1-10ms, whereas in the case of microparticle-electrode collisions driven under ultrasound, the duration of contact was only 1-10 $\mu$s.

In this work, we seek to develop an approximate model to confirm (or not) the above explanation of experimental observations, namely what is the required duration of the NP-electrode contact to be able to deposit a monolayer (or more) of thallium from solution onto the NP. Given the complexity of the impact process, the simplified model will aim to provide an 'order of magnitude' calculation for the required contact time,

which is a sufficient level of accuracy considering the three orders of magnitude difference in experimentally observed contact time.

Empirically, it therefore appears that the ability to deposit a metal onto a particle during contact with an electrode is dependent on solution metal ion concentration, size of particle and contact time. In this sub-chapter, we seek to address this question from a theoretical standpoint: based on a simple model where the surface area of the impacting particle is approximated to a microdisk electrode of equal area. Deposition of thallium onto the NP is assumed to be under mass transport control (i.e. there is sufficient overpotential to ensure that there are no kinetic effects) and that this occurs via diffusion only – whilst this neglects the additional mass transport due to stirring caused by the motion of the NP it does provide a set of minimum values for the key parameters for successful deposition.

### 6.6.2 Theory

In order to accurately model the impact of the spherical nanoparticle, it would be necessary to model of the deformation (i.e. flattening) of the NP, impact velocities, etc. that are not easily measurable in solution. We therefore adopt a simplified model of the contacting NP to produce results that lie with an order of magnitude of the actual, whilst being computationally tractable within a reasonable length of simulation time.

For monolayer deposition, the surface is subdivided into a number of sites on which material can deposit. If a site is deposited on, then that site becomes inactive to future deposition. In terms of thallium deposition on AgNPs, each deposition can be modelled

via stochastic deposition, using adapted Monte Carlo Random Walk simulations as described previously by Cutress et al. [119,120] and in this chapter.

The Random Walk methodology takes each particle in solution, and moves it in a random direction for a fixed distance every time-step [29,30]. If a particle hits a boundary or surface, the type of surface governs further interaction – at a deposition site, if the site is empty, the particle can deposit (if the potential is sufficient and kinetics allow), or reflect if the site is occupied. In this instance, the potential is set far enough to ensure deposition with every interaction with an open site.

Theoretically, we model the silver nanoparticle sphere as a microdisk of equivalent surface area, and focus on a quarter of the microdisk (to reduce simulation time), as shown in figure 6.21. The quadrant is subdivided into a deposition grid depending on the size of the deposited particle. The surface area of the depositing particle, for the simulation, is transformed into a square area, to give rise to the dimensionless quantity shown in equation 6.51 which divides the radius into *ES* number of sites available for deposition.

$$ES = \frac{length\ of\ deposition\ site}{radius\ of\ microdisk} \qquad [6.51]$$

The representation of a spherical nanoparticle as an inlaid (2-dimensional) microdisc of equivalent surface area is clearly a departure from the reality. For example, the transient behaviour observed at a sphere and disk is clearly different: even in unbounded space there is a factor of $\ln(2)$ in the steady-state currents. Further, the positioning of the impact site on the electrode will have an effect on the mass transport towards the contacting sphere, and the precise area of contact on the NP would not be expected to

be covered in deposited thallium. However, we estimate that the sum of these approximations, although not negligible, is within the order of magnitude estimate that we require in this work.

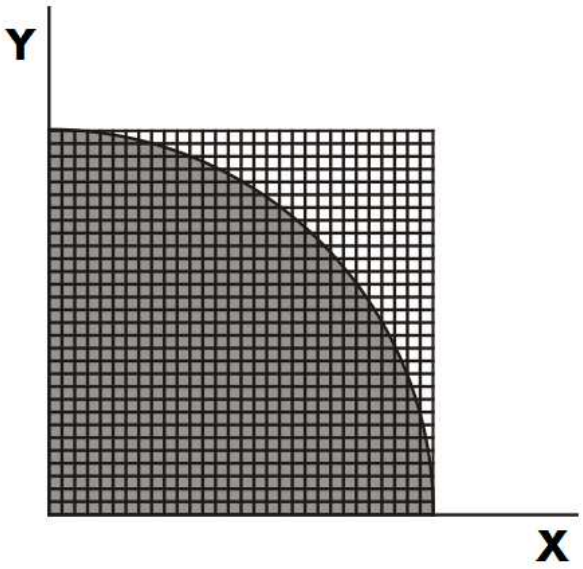



Figure 6.21 – Representation of computational deposition grid over a microdisk, where X and Y are the dimensionless lengths defined in table 6.7. Note, the sites outside the microdisk are ignored in the simulation.

These simulations are somewhat different to previous work on Monte Carlo Random Walk [119,120] due to the serial nature of the microdisk surface – previously, only the potential at the electrode surface was important to each particle in the simulation, and was defined explicitly as a function of the time. In these simulations, however, the electrode surface is a function of all the previous motions of all the previous time steps, depending on previous particles deposition. This means that at the end of the time step, each particle needs to know the state of the electrode surface in case that an interaction between the particle and the electrode happens on the next step.

The dimensionless quantities used in the simulation are analogous to those used in [120] and are given below in table 6.7, where $D_A$ is the diffusion coefficient of species A, $r_e$ is

the radius of the equivalent area disk electrode (i.e. double the NP radius), and $N_A$ is Avogadro's constant.

**Table 6.7. Dimensionless Quantities**

| Unit | Dimensional | Dimensionless |
|---|---|---|
| Time | t | $\tau = \frac{D_A t}{r_e^2}$ |
| Concentration | [c] | $C_E = [c]/N_A r_e^3$ |
| Length | $x$ | $X = x/r_e$ |
| | $y$ | $Y = y/r_e$ |
| | $z$ | $Z = z/r_e$ |
| Current | $i$ | $j = \frac{i\Delta t}{e}$ |

### 6.6.3 Computer Utilisation

Simulations were performed on an NVIDIA GTX 480 (448 CUDA cores, 1.536 GB memory, 737 MHz) using CUDA under Windows XP, similar to previous work[16,17,95,119,120]. Random numbers were generated with the CURAND package under the XORWOW algorithm (period of $2^{190}$), and simulation speeds were ~560 million particle movements per second.

The main limitation of the deposition method over previous non-deposition Monte Carlo random-walk simulation is that the position of every particle is needed in memory at all times, along with the random number seed (in this instance, 56 bytes of memory per particle), as well as the number of grid points (using $ES * ES *$ 4 bytes). The simulation memory usage is directly proportional to $ES^2 + (C_E \times \tau_{max}^3)$, which puts upper limits depending on the hardware available – thus the grid size of the nanoparticle being

simulated has a direct effect on the maximum possible concentration and dimensionless time calculable.

### 6.6.4 Results and Discussion

For the initial simulation, we adopt the experimental conditions described by Zhou et al. [117] for the UPD of thallium onto AgNPs of average radius 45 ± 3 nm. Using a spherical approximation for the AgNPs an equal-area microdisk would have a radius of 90nm. The ionic radius of thallium(I) is taken to be 164 pm, which gives the dimensions of the deposition sites to be 2.907 Å × 2.907 Å, leading to an ES value of 310 deposition sites per radius. For the purposes of the calculation, it is assumed that there is no change in radius of thallium between the ionic (solution phase) and atomic (deposited) species. Using the experimental concentration of 10 mM thallium(I) ions [117], the appropriate dimensionless concentration, CE, is 4390 (by table 6.7).

Figure 6.22 shows the results of a simulation based on CE = 4390 and ES = 310, in terms of percentage coverage against time. Using a diffusion coefficient of (2.4 ± 0.2) × 10-9 $m^2$ $s^{-1}$ for thallium(I) ions in 1.0 M $KNO_3$ [121] for (high supporting electrolyte concentrations are used to ensure full support [117]), it is found that 100% deposition occurs at a contact time of 78.6 μs, with upper and lower error bounds of 88.7 and 75.1 μs respectively. This result is in good agreement with the empirical observations noted above – that the occurrence of faradaic charge transfer processes during particle-electrode collisions, in this case, underpotential deposition, is strongly affected by the contact time.

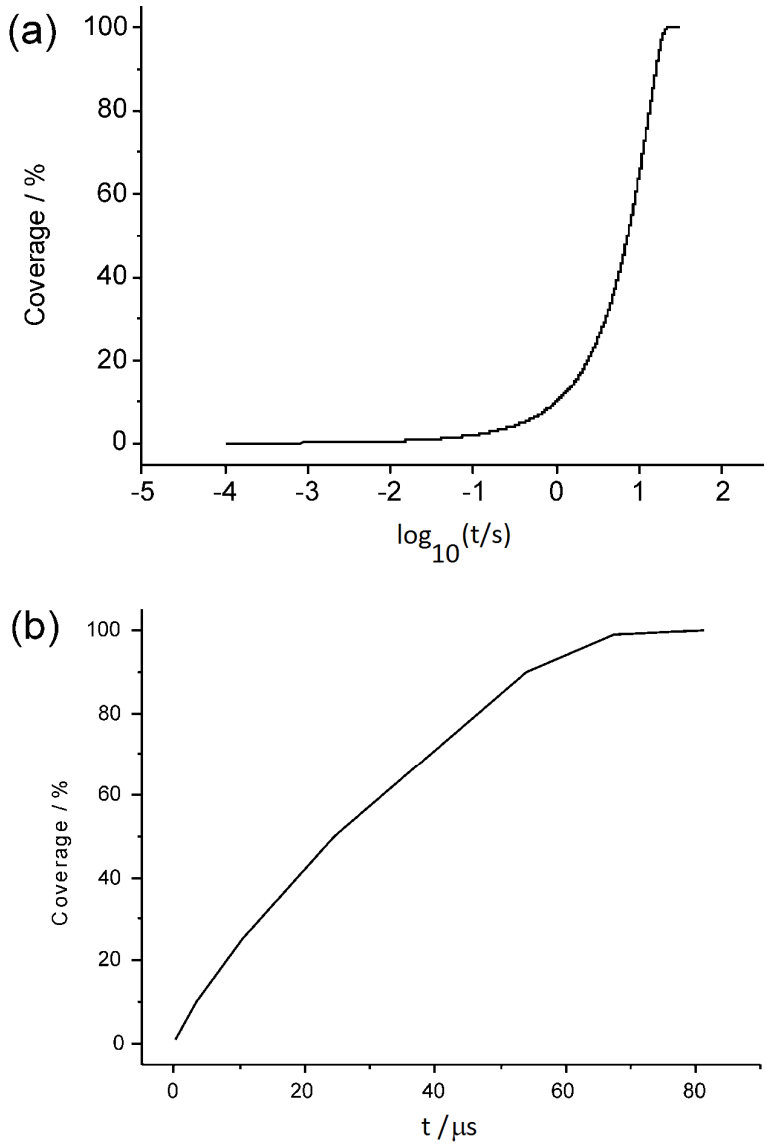



Figure 6.22 – Percentage coverage against dimensionless time for CE = 4390 and ES = 310, describing the experimental situation in [117], shown with (a) dimensionless time on a logarithmic scale, and (b) on a linear scale

The shapes of the graphs in figure 6.22 can be understood by considering the processes occurring during 'contact-deposition'. At short times, the simulation is under a Cottrellian regime of linear diffusion to the surface and essentially all of the deposition sites on the surface are available for particles to occupy. The deposition rate is therefore highest at this point, shown by the approximately linear section in figure 6.22b for early times. As

time continues, radial diffusion becomes more of a feature in terms of the number of thallium(I) ions having access to the surface; however there are fewer unoccupied deposition sites available. It is the decreasing number of free sites that becomes the dominating factor, and so the rate of deposition becomes increasingly slower, seen clearly in figure 6.22b as the gradient decreases until the surface has saturated. As an illustration, table 6.8 shows that deposition of 25% coverage occurs after 10.5 microseconds, which is ≈12.5% of the time needed for total deposition. However, the increase from 99% to 100% coverage requires 13.8 microseconds (or 16.9% of the time for full coverage).

**Table 6.8 – Percentage deposition of Thallium(I) ions on a 90 nm microdisk**

| Deposition / % | Time / μs |
|---|---|
| 1 | 0.107 |
| 10 | 3.399 |
| 25 | 10.516 |
| 50 | 24.445 |
| 90 | 53.913 |
| 99 | 67.547 |
| 100 | 81.324 |

It can therefore be seen that the possibility of successfully depositing a monolayer of thallium onto the AgNPs is dependent on the interplay of contact time, NP size, and solution concentration of thallium(I) ions. Hence, it would be helpful to investigate how the contact times required to achieve full monolayer deposition varies with both solution concentration and NP size. Simulations were carried out to study this, with for example, particles of 50 nm radius in 20 mM thallium(I) solutions, where simulation parameters of

$C_E$ = 10244 and *ES* = 344 are required. Due to the larger particle, there are more deposition sites in total that require to be filled. As a result, despite the larger value of dimensionless concentration, longer times are required for full deposition (and consequently, longer contact times for nanoparticles on the electrode surface). Tables 6.9-6.12 show the contact times required for full monolayer deposition for a variety of NP sizes and solution concentrations in terms of dimensionless time – for real world values in seconds, each value should be multiplied by $r_e^2/D_i$ (where $D_i$ is the diffusion coefficient of the depositing species) as per Table 6.7.

**Table 6.9 Dimensionless times for coverage at a 50 nm NP**

| | 1 mM | 5 mM | 10 mM | 20 mM |
|---|---|---|---|---|
| 1% | 0.8326 | 0.0816 | 0.0282 | 0.0074 |
| 10% | 13.0761 | 2.0895 | 0.8689 | 0.3375 |
| 25% | 35.5684 | 6.1670 | 2.7363 | 1.1829 |
| 50% | 75.7242 | 13.7080 | 6.3603 | 2.8710 |
| 90% | -- | 29.6697 | 14.1282 | 6.6453 |
| 99% | -- | -- | 17.5907 | 8.5165 |
| 100% | -- | -- | -- | -- |

**Table 6.10 Dimensionless times for coverage at a 20 nm NP**

| | 1 mM | 5 mM | 10 mM | 20 mM |
|---|---|---|---|---|
| 1% | 2.9080 | 0.2983 | 0.1061 | 0.0425 |
| 10% | 36.3028 | 6.0496 | 2.7841 | 1.2220 |
| 25% | 95.8526 | 17.1487 | 8.0769 | 3.6006 |
| 50% | 201.0538 | 36.9308 | 17.8247 | 8.2733 |
| 90% | 406.5494 | 79.0430 | 38.1605 | 18.1521 |
| 99% | 495.0224 | 96.1408 | 47.7311 | 22.6727 |
| 100% | 564.2000 | 107.7664 | 51.1772 | 25.7702 |

**Table 6.11 Dimensionless times for coverage at a 10 nm NP**

| | 1 mM | 5 mM | 10 mM | 20 mM |
|---|---|---|---|---|
| 1% | 4.7862 | 0.5469 | 0.3764 | 0.0939 |
| 10% | 79.1188 | 14.0905 | 7.1336 | 2.8059 |
| 25% | 187.4596 | 36.9692 | 17.8442 | 8.1119 |
| 50% | 391.6165 | 76.7961 | 38.2462 | 18.1691 |
| 90% | 795.3034 | 160.6070 | 79.5090 | 38.3163 |
| 99% | 960.7801 | 197.2455 | 96.7854 | 47.9889 |
| 100% | 1081.6888 | 238.7944 | 108.9662 | 55.0154 |

**Table 6.12 Dimensionless times for coverage at a 5 nm NP**

| | 1 mM | 5 mM | 10 mM | 20 mM |
|---|---|---|---|---|
| 1% | 10.3176 | 1.9564 | 0.7216 | 0.2175 |
| 10% | 139.2326 | 29.3294 | 12.6803 | 7.8970 |
| 25% | 371.8478 | 77.6535 | 35.3029 | 18.0922 |
| 50% | 771.9187 | 160.3380 | 77.5426 | 37.9909 |
| 90% | 1563.6218 | 305.4850 | 156.8094 | 75.8371 |
| 99% | 1882.6106 | 377.4778 | 189.2761 | 93.8214 |
| 100% | 1944.0189 | 388.2124 | 198.8721 | 98.3696 |

### 6.6.5 Conclusions

The contact time required for the deposition of a species onto a nanoparticle during a nanoparticle-electrode collision has been successfully simulated for coverages up to a monolayer using a simple physical model. The result that a contact time of ca. $10^{-3}$ s is required to deposit a monolayer of thallium onto a NP of radius 45nm is in agreement with the experimental observation that this process occurs successfully where the NPs

move under gravity and contact time is ca. 1-10ms [116], whereas similar faradaic processes could not be observed for microparticles under insonation where contact time was 1-10 $\mu$s [108].

**6.7 Conclusions on Random Walk Simulations**

This chapter has dealt with the concept of modelling voltammetric and electrochemical systems by the particles that those systems contain. By introducing the concept brought forward by Nagy [1], the underlying simulation has been analysed and improved to take into account full three dimensional diffusion, as well as standard electrochemical simulation theory. The algorithm for diffusion was also analysed, and compared to several sphere-point placement algorithms currently in use for speed and calculation optimisations. The fastest algorithm was subsequently coded onto a graphics card for high-speed parallel computation. Using the random-walk methodology, this chapter describes the three publications generated via this method for this thesis.

Initially, as shown in chapter 6.4, the low concentration scenarios of as probed by Fan et al [42] were studied, followed by the low concentration 'single-molecule' type systems involving redox cycling of reversible species as described [40] were simulated with success. These latter simulations showed that the close proximity of two electrodes resulted in a higher signal of individual molecules, rather than larger spaces containing more molecules at the same concentration. Finally, as a further exercise, a system of two close infinite planes was simulated, with two opposing microdisks in generator collector mode.

The second paper, as described in chapter 6.5, examined the concentration limit as which cyclic voltammetry can be performed, due to the statistical random motion of the particles in solution. The concepts of applying the kinetics of an electron transfer in the

context of probabilities, by both Butler-Volmer, and Marcus-Hush through the evaluation of the Marcus-Hush-Chidsey integral and optimisation for computational calculation.

Finally, the third publication for this thesis chapter, as shown in chapter 6.6, simulates work done regarding monolayer deposition of thallium on silver nanoparticles while they are in contact with an electrode surface. This situation provides a serial step in the random walk simulations such that each particle is knowledgeable of which sites on the electrode surface are available for deposition.

**Bibliography**


[1] G. Nagy, Y. Sugimoto, G. Denuault, J. Electroanal. Chem. 433 (1997) 167-173.
[2] J.E. Baur, P.N. Motsegood, J. Electroanal. Chem. 572 (2004) 29-40.
[3] S.A. Liang, Q.J. Guo, X.D. Dong, J. Electroanal. Chem. 633 (2009) 235-239.
[4] S. Licht, V. Cammarata, M.S. Wrighton, Science 243 (1989) 1176-1178.
[5] S. Licht, V. Cammarata, M.S. Wrighton, J. Phys. Chem. 94 (1990) 6133-6140.
[6] H. Yang, J.L. Peters, C. Allen, S.S. Chern, R.D. Coalson, A.C. Michael, Anal. Chem. 72 (2000) 2042-2049.
[7] T.M. Bartol, B.R. Land, E.E. Salpeter, M.M. Salpeter, Biophys. J. 59 (1991) 1290-1307.
[8] G. Nagy, G. Denuault, J. Electroanal. Chem. 433 (1997) 175-180.
[9] M.C. Gimenez, M.G. Del Popolo, E.P.M. Leiva, Langmuir 18 (2002) 9087-9094.
[10] M.C. Gimenez, M.G. Del Popolo, E.P.M. Leiva, Electrochim. Acta 45 (1999) 699-712.
[11] L. Oroszi, O. Hasemann, E. Wolff, A. Der, Bioelectrochemistry 60 (2003) 97-106.
[12] F. Grun, M. Jardat, P. Turq, C. Amatore, J. Chem. Phys. 120 (2004) 9648-9655.
[13] C. Amatore, O.V. Klymenko, A.I. Oleinick, I. Svir, Chemphyschem 10 (2009) 1593-1602.
[14] C. Amatore, A.I. Oleinick, O.V. Klymenko, I. Svir, Chemphyschem 10 (2009) 1586-1592.
[15] W. Krauth, in W. Krauth (Ed.), Statistical Mechanics: Algorithms and Computations. Oxford University Press, Oxford, 2007.
[16] I.J. Cutress, R.G. Compton, J. Electroanal. Chem. 645 (2010) 159-166.
[17] I.J. Cutress, R.G. Compton, J. Electroanal. Chem. 643 (2010) 102-109.
[18] D. Britz, in D. Britz (Ed.), Digital Simulation in Electrochemistry. Springer, New York, 2005, p. 187.
[19] J. Strutwolf, Electroanalysis 17 (2005) 1547-1554.
[20] J. Douglas, Numerische Mathematik 4 (1962) 41.
[21] J. Noye, in J. Noye (Ed.), Computational Techniques for Differential Equations. Elsevier, Amsterdam 1984, p. 95-354.
[22] I.J. Cutress, E.J.F. Dickinson, R.G. Compton, J. Electroanal. Chem. 638 (2010) 76-83.
[23] D. Menshykau, F.J. del Campo, F.X. Munoz, R.G. Compton, Sensors and Actuators B-Chemical 138 (2009) 362-367.
[24] S.R. Belding, R.G. Compton, J. Phys. Chem. C 114 (2010) 8309-8319.
[25] E.J.F. Dickinson, R.G. Compton, Chem. Phys. Lett. 497 (2010) 178-183.
[26] D.J. Gavaghan, J. Electroanal. Chem. 456 (1998) 1-12.
[27] I.J. Cutress, R.G. Compton, Electroanalysis 21 (2009) 2617-2625.

[28] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes: The Art of Scientific Computing. Cambridge University Press, 2007, p. 352.
[29] A. Einstein, Annalen der Physik 322 (1905) 549-560.
[30] M. von Smoluchowski, Annalen der Physik 326 (1906) 756-780.
[31] M.A. Islam, Phys. Scr. 70 (2004) 120-125.
[32] D.E. Knuth, in D.E. Knuth (Ed.), Seminumerical Algorithms. Addison-Wesley, 1981, p. 130-131.
[33] L. Sheng-Te, G. Marsaglia, http://www.math.niu.edu/~rusin/known-math/96/sph.rand, http://www.math.niu.edu/~rusin/known-math/96/sph.rand, 1996.
[34] http://www.nvidia.com/object/cuda_get.html, 2010.
[35] G. Marsaglia, Journal of Statistical Software 8 (2003) 1-6.
[36] A. Habegger, A. Stahel, J. Goette, M. Jacomet, Electronic Design, Test and Application, 2010. DELTA '10. Fifth IEEE International Symposium on 1 (2010) 183-187.
[37] Intel, http://www.intel.com/technology/architecture-silicon/intel64/45nm-core2_whitepaper.pdf, Introducing the 45nm Next-Generation Intel® Core™ Microarchitecture, 2008.
[38] A.J. Bard, F.R.F. Fan, J. Kwak, O. Lev, Anal. Chem. 61 (1989) 132-138.
[39] A.J. Bard, F.R.F. Fan, M.V. Mirkin, in A.J. Bard (Ed.), Electroanalytical Chemistry. Dekker, New York, 1994, p. 243-373.
[40] F.R.F. Fan, A.J. Bard, Science 267 (1995) 871-874.
[41] A.J. Bard, F.R.F. Fan, Acc. Chem. Res. 29 (1996) 572-578.
[42] F.R.F. Fan, J. Kwak, A.J. Bard, J. Am. Chem. Soc. 118 (1996) 9669-9675.
[43] C. Amatore, A.I. Oleinick, I. Svir, Chemphyschem 11 (2010) 149-158.
[44] R.S. Nicholson, I. Shain, Anal. Chem. 36 (1964) 706.
[45] R.S. Nicholson, I. Shain, Anal. Chem. 37 (1965) 178.
[46] K. Aoki, K. Honda, K. Tokuda, H. Matsuda, J. Electroanal. Chem. 182 (1985) 267-279.
[47] K. Aoki, H. Kaneko, J. Electroanal. Chem. 247 (1988) 17-27.
[48] T.J. Davies, R.G. Compton, J. Electroanal. Chem. 585 (2005) 63-82.
[49] E.D. Goluch, B. Wolfrum, P.S. Singh, M.A.G. Zevenbergen, S.G. Lemay, Anal. Bioanal. Chem. 394 (2009) 447-456.
[50] M.A.G. Zevenbergen, D. Krapf, M.R. Zuiddam, S.G. Lemay, Nano Lett. 7 (2007) 384-388.
[51] D. Shoup, A. Szabo, J. Electroanal. Chem. 140 (1982) 237-245.
[52] L.M. Goncalves, C. Batchelor-McAuley, A.A. Barros, R.G. Compton, J. Phys. Chem. C 114 (2010) 14213-14219.
[53] Q. Li, C. Batchelor-McAuley, R.G. Compton, J. Phys. Chem. B 114 (2010) 7423-7428.
[54] Q.A. Li, C. Batchelor-McAuley, R.G. Compton, J. Phys. Chem. B 114 (2010) 9713-9719.
[55] J.M. Orts, A. Fernandezvega, J.M. Feliu, A. Aldaz, J. Clavilier, J. Electroanal. Chem. 327 (1992) 261-278.
[56] G.K. Mor, M.A. Carvalho, O.K. Varghese, M.V. Pishko, C.A. Grimes, J. Mater. Res. 19 (2004) 628-634.
[57] A. Napier, J.P. Hart, Electroanalysis 8 (1996) 1006-1013.
[58] R.T. Kachoosangi, G.G. Wildgoose, R.G. Compton, Analyst 133 (2008) 888-895.
[59] B.C.M. Martindale, L. Aldous, N.V. Rees, R.G. Compton, Analyst 136 (2011) 128-133.
[60] J.S. Ye, Y. Wen, W. De Zhang, L.M. Gan, G.Q. Xu, F.S. Sheu, Electroanalysis 15 (2003) 1693-1698.
[61] W. Kaim, R. Reinhardt, M. Sieger, Inorg. Chem. 33 (1994) 4453-4459.
[62] S.A. Lerke, B.A. Parkinson, D.H. Evans, P.J. Fagan, J. Am. Chem. Soc. 114 (1992) 7807-7813.
[63] F. Marken, R.G. Compton, C.H. Goeting, J.S. Foord, S.D. Bull, S.G. Davies, Electroanalysis 10 (1998) 821-826.
[64] H.O. Finklea, D.D. Hanshew, J. Am. Chem. Soc. 114 (1992) 3173-3181.

[65] C.J. Tonzola, M.M. Alam, W. Kaminsky, S.A. Jenekhe, J. Am. Chem. Soc. 125 (2003) 13548-13558.
[66] A. Salimi, V. Alizadeh, R.G. Compton, Anal. Sci. 21 (2005) 1275-1280.
[67] R. Baron, N.M. Kershaw, T.J. Donohoe, R.G. Compton, J. Phys. Org. Chem. 22 (2009) 247-253.
[68] R. Baron, N.M. Kershaw, T.J. Donohoe, R.G. Compton, J. Phys. Org. Chem. 22 (2009) 1136-1141.
[69] R. Wibowo, L. Aldous, E.I. Rogers, S.E.W. Jones, R.G. Compton, J. Phys. Chem. C 114 (2010) 3618-3626.
[70] A. Pozio, M. De Francesco, A. Cemmi, F. Cardellini, L. Giorgi, J. Power Sources 105 (2002) 13-19.
[71] F. Selampinar, L. Toppare, U. Akbulut, T. Yalcin, S. Suzer, Synth. Met. 68 (1995) 109-116.
[72] Z.G. Qi, M.C. Lefebvre, P.G. Pickup, J. Electroanal. Chem. 459 (1998) 9-14.
[73] B.S. Sherigara, W. Kutner, F. D'Souza, Electroanalysis 15 (2003) 753-772.
[74] B.A. Brookes, T.J. Davies, A.C. Fisher, R.G. Evans, S.J. Wilkins, K. Yunus, J.D. Wadhawan, R.G. Compton, J. Phys. Chem. B 107 (2003) 1616-1627.
[75] T.J. Davies, R.R. Moore, C.E. Banks, R.G. Compton, J. Electroanal. Chem. 574 (2004) 123-152.
[76] C. Amatore, Y. Bouret, E. Maisonhaute, J.I. Goldsmith, H.D. Abruna, Chemphyschem 2 (2001) 130-+.
[77] C. Amatore, F. Grun, E. Maisonhaute, Angewandte Chemie-International Edition 42 (2003) 4944-4947.
[78] F.R.F. Fan, A.J. Bard, Science 277 (1997) 1791-1793.
[79] F.W. Campbell, R.G. Compton, Anal. Bioanal. Chem. 396 (2010) 241-259.
[80] I.J. Cutress, E.J.F. Dickinson, R.G. Compton, J. Electroanal. Chem. (2011) in press, doi: 10.1016/j.jelechem.2011.1002.1023.
[81] C.E. Banks, R.G. Compton: Understanding Voltammetry, World Scientific, Singapore, 2011.
[82] J.A.V. Butler, Transactions of the Faraday Society 28 (1932) 379-382.
[83] T. Erdey-Grúz, M. Volmer, Z. Phys. Chem. 150 (1930) 203-213.
[84] A.J. Bard, L.R. Faulkner: Electrochemical Methods: Fundamentals and Applications, Wiley, Phoenix, 2001.
[85] R.A. Marcus, J. Chem. Phys. 24 (1956) 966-978.
[86] R.A. Marcus, J. Phys. Chem. 67 (1963) 853.
[87] R.A. Marcus, Annu. Rev. Phys. Chem. 15 (1964) 155.
[88] N.S. Hush, Transactions of the Faraday Society 57 (1961) 557.
[89] S.W. Feldberg, Anal. Chem. 82 (2010) 5176-5183.
[90] K.B. Oldham, C.G. Zoski, J. Electroanal. Chem. 256 (1988) 11-19.
[91] C.E.D. Chidsey, Science 251 (1991) 919-922.
[92] K.B. Oldham, J.C. Myland, J. Electroanal. Chem. (2011) in press, doi: 10.1016/j.jelechem.2011.1001.1044.
[93] M.C. Henstridge, Personal Communication, 2011.
[94] K. Weber, S.E. Creager, Anal. Chem. 66 (1994) 3164-3172.
[95] I.J. Cutress, Y.J. Wang, J.G. Limon-Petersen, S.E.C. Dale, L. Rassaei, F. Marken, R.G. Compton, J. Electroanal. Chem. 655 (2011) 147-153.
[96] F.G. Chevallier, T.J. Davies, E.V. Klymenko, L. Jiang, T.G.J. Jones, R.G. Compton, J. Electroanal. Chem. 577 (2005) 211-221.
[97] M. Heyrovsky, J. Jirkovsky, B.R. Muller, Langmuir 11 (1995) 4293-4299.
[98] M. Heyrovsky, J. Jirkovsky, M. Struplovabartackova, Langmuir 11 (1995) 4300-4308.
[99] M. Heyrovsky, J. Jirkovsky, M. Struplovabartackova, Langmuir 11 (1995) 4309-4312.
[100] I.D. Charlton, A.P. Doherty, J. Phys. Chem. B 104 (2000) 8061-8067.

[101] I.D. Charlton, A.P. Doherty, Anal. Chem. 72 (2000) 687-695.
[102] D. Hellberg, F. Scholz, F. Schauer, W. Weitschies, Electrochem. Commun. 4 (2002) 305-309.
[103] F. Scholz, D. Hellberg, F. Harnisch, A. Hummel, U. Hasse, Electrochem. Commun. 6 (2004) 929-933.
[104] D. Hellberg, F. Scholz, F. Schubert, M. Lovric, D. Omanovic, V.A. Hernandez, R. Thede, J. Phys. Chem. B 109 (2005) 14715-14726.
[105] C.E. Banks, N.V. Rees, R.G. Compton, J. Electroanal. Chem. 535 (2002) 41-47.
[106] C.E. Banks, N.V. Rees, R.G. Compton, J. Phys. Chem. B 106 (2002) 5810-5813.
[107] N.V. Rees, C.E. Banks, R.G. Compton, J. Phys. Chem. B 108 (2004) 18391-18394.
[108] A.D. Clegg, N.V. Rees, C.E. Banks, R.G. Compton, Chemphyschem 7 (2006) 807-811.
[109] A.J. Bard, X.Y. Xiao, J. Am. Chem. Soc. 129 (2007) 9610-+.
[110] A.J. Bard, X.Y. Xiao, F.R.F. Fan, J.P. Zhou, J. Am. Chem. Soc. 130 (2008) 16669-16677.
[111] A.J. Bard, X.Y. Xiao, S.L. Pan, J.S. Jang, F.R.F. Fan, J. Phys. Chem. C 113 (2009) 14978-14982.
[112] A.J. Bard, S.J. Kwon, F.R.F. Fan, J. Am. Chem. Soc. 132 (2010) 13165-13167.
[113] A.J. Bard, S.J. Kwon, H.J. Zhou, F.R.F. Fan, V. Vorobyev, B. Zhang, PCCP 13 (2011) 5394-5402.
[114] R.G. Compton, Y.G. Zhou, N.V. Rees, Angewandte Chemie-International Edition 50 (2011) 4219-4221.
[115] N.V. Rees, Y.G. Zhou, R.G. Compton, Chemphyschem 12 (2011) 1645-1647.
[116] Y.G. Zhou, N.V. Rees, R.G. Compton, Chemphyschem (2011) submitted.
[117] Y.G. Zhou, N.V. Rees, R.G. Compton, Chemphyschem (2011) 10.1002/cphc.201100282.
[118] Y.G. Zhou, N.V. Rees, R.G. Compton, Chem. Phys. Lett. DOI 10.1016/j.cplett.2011.06.015 (2011).
[119] I.J. Cutress, E.J.F. Dickinson, R.G. Compton, J. Electroanal. Chem. 655 (2011) 1-8.
[120] I.J. Cutress, R.G. Compton, Chem. Phys. Lett. 508 (2011) 306-313.
[121] J.G. Limon-Petersen, I. Streeter, N.V. Rees, R.G. Compton, J. Phys. Chem. C 112 (2008) 17175-17182.

## Chapter 7 - Conclusions

This thesis has dealt with the concept of algorithm development in computational chemistry. Initially, this thesis discussed whether algorithm development was necessary, given the level of commercial engineering finite element solvers available for the electrochemist. Given the inherent problems found in these utilities, this thesis described the need for simulators to keep abreast of recent computational developments in terms of hardware and software, for two main simulation reasons: to increase speed of lengthy simulation, and to open new avenues of simulation theory.

Algorithm development in this thesis has been taken through both stages as defined at the beginning of chapter 3 – by either applying theory to a new area, or using novel methods to generate new algorithms. Chapter 3 itself deals specifically with applying theory relating to the stripping of a macroelectrode to the diffusional model of a microdisk, using common simulation techniques.

Chapter 4 introduces the concept of parallel computing, and how computational hardware has developed recently to take advantage of out-of-order calculations, by processing them in parallel to reduce simulation time. The novel area of graphics card simulation for highly parallel algorithms is also explained in a sufficient level of detail such that future simulators should be relevant of all the current features provided.

Chapter 5 discusses the adaptation of voltammetric finite difference algorithms in a purely parallel format for simulation by explicit solution. Through explicit solution, finite difference algorithms were applied to electrode geometries which necessitate a three-dimensional solution – elliptical electrodes; square, rectangular, and microband electrodes; and dual

microdisk electrodes in collector-generator mode. Previous solution methods to these problems by the explicit finite difference algorithm were excruciatingly slow and thus not applicable to be studied by this method, however the adaptation of the graphics card to the simulation affords a substantial speed up to achieve appropriate quantitative results.

Chapter 6 introduces a simulation concept which has only been applied recently to voltammetric analysis, whereby individual particles in the simulation are modelled and their trajectories over time are calculated. These 'Random Walk' simulations, for this thesis, were improved for pure three-dimensional diffusion, and adapted to graphics cards, allowing up to thirteen billion particles movements per second be calculated on current computational hardware, rather than the three million particle movements per second expected from previous methods. This method was adapted to various areas of low concentration confined voltammetry (chapter 6.4) and single molecule detection, ultra low concentration cyclic voltammetry (chapter 6.5), and underpotential deposition of thallium on mobile silver nanoparticles (chapter 6.6).

Overall, this thesis has presented and applied, with success, a series of algorithm development concepts in computational electrochemistry.

**Appendix A – Computational Digit Representation, and Reasons for Using Dimensionless Units**

An electrochemical voltammetric simulation is often reduced to 'dimensionless' units for a variety of reasons. By transforming the simulation space, time and parameters into dimensionless units, this allow for independence against the scaling parameters of the simulation. An input of dimensionless rate constant and scan rate can then relate to a family of results, rather than simulating one at a time.

Another reason for reducing to dimensionless units is one based in computational calculation. Most numbers stored in a simulation are either as whole numbers (integers) or as numbers with decimal values (floating point numbers). An integer is easily represented in a computer, as a series of 8 'bits' in a byte. Each bit represents $2^x$, where x is the position from the beginning, as shown in Figure A.1. The total value represented is determined by the addition of all the activated bits, such as figure A.2, which represents the number 81.

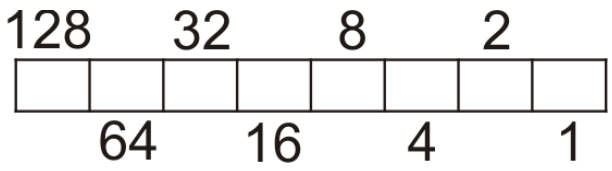



Figure A.1 – Representation of a one byte integer, up to 255.


Figure A.2 – One byte integer representing 81.

Integers are often 4, 8 or 16 bytes long, offering different maximum values presentable at the expense of computer memory. In the event that the maximum value is achieved and

subsequently more is added to the variable, the value is then misrepresented by a negative value in a process called 'wrapping around', and is inherent in all computational integer calculation.

Floating point numbers are represented similarly in bits, but the calculation of which bits are active is slightly different. The simplest floating point number is called a 'float' or a 'single', which is a 4 byte (32-bit) representation of a decimal number. These 32 bits are split as follows:

- Bit 1 : Sign (+/-)
- Bit 2-9: Exponent
- Bit 10-32: Significand bit

The calculation of the value this 4-byte single represents is given by:

$$value = (-1)^{sign}\left(1 + \sum_{i=1}^{23} bit_i 2^i\right) \times 2^{(exponent-127)} \qquad [A.1]$$

This leads to two different issues. One is that due to this representation, not every decimal value can be given accurately – for example, if the value 2.5478 is placed in a 4 byte float, the actual value stored is 2.5478001, as this is the nearest number presentable. This could potentially have issues when dealing with calculations where a combination of these errors could influence the overall result.

The other issue is the level of precision within the 4 byte float itself, which is limited to 7 significant figures. Take the circumstance of a variable which starts as a single value (e.g. pi) and is constantly added by the same value. There comes a point where due to the level of accuracy, only 3.141 is being added, then 3.14, then 3.1 and so on, thus underestimating the total value of the variable.

A way of dealing with both of these issues is switching to an 8 byte floating point variable, commonly referred to as a 'double'. A double uses the extra 4 bytes worth of bits to increase the exponent from 8 bits to 11, and the significant to 52 bits. This increases the maximum value presentable from $3.4028234 \times 10^{38}$ to $1.7976931348623157 \times 10^{308}$, and the precision from 7 significant figures to 16. These are both at the expense of requiring double the computer memory, however this is often not an issue for electrochemical simulation, thus the use of double floating point variables is ubiquitous in the literature.

Going back to the simulation and the discussion of dimensionless units, if real values of concentration, time and spacial ordinates were considered, computationally the manipulation of nano-molar to molar values combined with micro-seconds and sub-picometer length levels would lead to significant computational error in all the variables used. By converting to a dimensionless system, where lengths are normalised against the electrode radius, times are normalised with respect to the diffusion coefficient, and concentrations are normalised to the bulk values, not only does less computational error occur based on numerical misrepresentation, but also the algorithms are often easier to implement in computer code.

**Appendix B - Proof of Uniform Distribution in One Axis of a Sphere**

The area of a surface of revolution is given by equation B.1.

$$A = 2\pi \int_a^b f(x)\sqrt{1 + [f'(x)]^2}\, dx \qquad \text{[B.1]}$$

Consider a sphere of radius $R$. At any height $z$ along the $z$-axis, the radius of the circle intersected at that point, is given by:

$$f(z) = \sqrt{R^2 - z^2} \qquad \text{[B.2]}$$

$$f'(z) = \frac{-z}{\sqrt{R^2 - z^2}} \qquad \text{[B.3]}$$

$$1 + [f'(z)]^2 = 1 + \frac{z^2}{R^2 - z^2} = \frac{R^2 - z^2}{R^2 - z^2} + \frac{z^2}{R^2 - z^2} = \frac{R^2}{R^2 - z^2} \qquad \text{[B.4]}$$

Thus the area between points $b$ and $a$ on the $z$-axis of the sphere is:

$$A = 2\pi \int_a^b f(z)\sqrt{1 + [f'(z)]^2}\, dz$$

$$= 2\pi \int_a^b \left(\sqrt{R^2 - z^2}\right)\left(\frac{R^2}{R^2 - z^2}\right) dz \qquad \text{[B.5]}$$

$$= 2\pi R \int_a^b dz = 2\pi R(b - a)$$

Given that $(b - a)$ is the height of the vertical zone and is reduced to a constant, the density of points in the axis is uniform.

**Appendix C – Publications During D.Phil. Research**

[1] **IJ Cutress**, NV Rees, YG Zhou, RG Compton, *Nanoparticle–electrode collision processes: Investigating the contact time required for the diffusion-controlled monolayer underpotential deposition on impacting nanoparticles*, Chemical Physics Letters, *514*, (**2011**), 58-61

[2] **IJ Cutress**, RG Compton, *How many molecules are required to measure a cyclic voltammogram?*, Chemical Physics Letters, *508*, (**2011**), 306-313.

[3] **IJ Cutress**, Y Wang, JG Limon-Petersen, SEC Dale, L Rassaei, F Marken, RG Compton, *Dual-microdisk electrodes in transient generator-collector mode: Experiment and theory*, Journal of Electroanalytical Chemistry, *655*, (**2011**), 147-153.

[4] **IJ Cutress**, EJF Dickinson, RG Compton, *Electrochemical random-walk theory Probing voltammetry with small numbers of molecules: Stochastic versus statistical (Fickian) diffusion*, Journal of Electroanalytical Chemistry, *655*, (**2011**), 1-8.

[5] **IJ Cutress**, RG Compton, *Theory of square, rectangular, and microband electrodes through explicit GPU simulation*, Journal of Electroanalytical Chemistry, *645*, (**2010**), 159-166.

[6] **IJ Cutress**, RG Compton, *Using graphics processors to facilitate explicit digital electrochemical simulation: Theory of elliptical disc electrodes*, Journal of Electroanalytical Chemistry, *643*, (**2010**), 102-109.

[7] **IJ Cutress**, EJF Dickinson, RG Compton, *Analysis of commercial general engineering finite element software in electrochemical simulations*, Journal of Electroanalytical Chemistry, *638*, (**2010**), 76-83.

[8] **IJ Cutress**, RG Compton, *Stripping Voltammetry at Microdisk Electrode Arrays: Theory*, Electroanalysis, *21*, (**2009**), 2617-2625.

[9] **IJ Cutress**, F Marken, RG Compton, *Microwave-Assisted Electroanalysis: A Review*, Electroanalysis, *21*, (**2009**), 113-123.