LBL-36293
UC-331

Lawrence Berkeley Laboratory

UNIVERSITY OF CALIFORNIA

ENERGY & ENVIRONMENT DIVISION

Thermal Modeling of the Lithium/Polymer Battery

C.R. Pals
(M.S. Thesis)

October 1994

ENERGY & ENVIRONMENT
DIVISION

REFERENCE COPY
Does Not
Circulate
Bldg. 50 Library.
LBL-36293
Copy 1

Prepared for the U.S. Department of Energy under Contract Number DE-AC03-76SF00098

DISCLAIMER

This document was prepared as an account of work sponsored by the United States Government. While this document is believed to contain correct information, neither the United States Government nor any agency thereof, nor the Regents of the University of California, nor any of their employees, makes any warranty, express or implied, or assumes any legal responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by its trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof, or the Regents of the University of California. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof or the Regents of the University of California.

LBL-36293
UC-331

Thermal Modeling of the Lithium/Polymer Battery

Carolyn R. Pals

M.S. Thesis

Department of Chemical Engineering,
University of California

and

Energy and Environment Division
Lawrence Berkeley Laboratory
Berkeley, California 94720

October 1994

This work was supported in part by the Assistant Secretary for Energy Efficiency and Renewable Energy, Office of Transportation Technologies, Electric and Hybrid Propulsion Division of the U.S. Department of Energy under Contract No. DE-AC03-76SF00098.

Abstract

Thermal Modeling of the Lithium/Polymer Battery

by

Carolyn Renee Pals

Master of Science in Chemical Engineering

University of California at Berkeley

Professor John S. Newman, Chair

Research in the area of advanced batteries for electric-vehicle applications has increased steadily since the 1990 zero-emission-vehicle mandate of the California Air Resources Board. Due to their design flexibility and potentially high energy and power densities, lithium/polymer batteries are an emerging technology for electric-vehicle applications. Thermal modeling of lithium/polymer batteries is particularly important because the transport properties of the system depend exponentially on temperature.

Two models have been presented for assessment of the thermal behavior of lithium/polymer batteries. The one-cell model predicts the cell potential, the concentration profiles, and the heat-generation rate during discharge. The cell-stack model predicts temperature profiles and heat transfer limitations of the battery.

Due to the variation of ionic conductivity and salt diffusion coefficient with temperature, the performance of the lithium/polymer battery is greatly affected by temperature. Because of this variation, it is important to optimize the cell operating temperature and design a thermal management system for the battery. Since the thermal conductivity of the polymer electrolyte is very low, heat is not easily conducted in the direction perpendicular to cell layers.

Temperature profiles in the cells are not as significant as expected because heat-generation rates in warmer areas of the cell stack are lower than heat-generation rates in cooler areas of the stack. This nonuniform heat-generation rate flattens the temperature profile. Temperature profiles as calculated by this model are not as steep as those calculated by previous models that assume a uniform heat-generation rate.

Table of Contents

Thermal Modeling of the Lithium/Polymer Battery

Chapter 1

1.1 Motivation 1
1.2 Lithium/Polymer Batteries 2
1.3 Thermal Modeling 3
1.4 Conclusion 5
References 6

Chapter 2

2.1 Introduction 8
2.2 The Cell 8
2.3 The Model of Doyle *et al.* 10
2.4 Energy Balance 16
2.5 Physical Properties for the $Li|PEO_{18}$-$LiCF_3SO_3|TiS_2$ System 19
List of Symbols 27
References 30

Chapter 3

3.1 Introduction 32
3.2 Isothermal Discharge Behavior 32
3.3 Adiabatic Discharge Behavior 38
3.4 Heat Transfer 48
3.5 Conclusion 57
List of Symbols 58
References 58

Chapter 4

4.1 Introduction 59
4.2 The Cell Stack 60
4.3 Energy Balance 60
4.4 Approximation for the Heat-Generation Rate 63

4.5 Validity of the Heat-Generation-Rate Approximation 64
4.6 Conclusion 75
List of Symbols 78
References 78

Chapter 5
5.1 Introduction 80
5.2 Temperature Profiles 80
5.3 Electric-Vehicle Battery 86
5.4 Conclusion 97

Acknowledgments

This work was supported in part by the Assistant Secretary for Energy Efficiency and Renewable Energy, Office of Transportation Technologies, Electric and Hybrid Propulsion Division of the U. S. Department of Energy under Contract No. DE-AC03-76SF00098.

Chapter 1: Introduction

1.1 Motivation

In 1990, as part of its Low-Emission Vehicle and Clean Fuels regulations, the California Air Resources Board adopted a zero-emission-vehicle (ZEV) mandate.[1] The ZEV mandate requires that by 1998, two percent of passenger cars and light trucks offered for sale in California by each major automobile manufacturer must be ZEVs. In 2001, the percentage increases to five percent , and in 2003, ten percent. The only ZEV technology advanced enough to meet the 1998 mandate is the battery-powered electric vehicle.

To streamline electric-vehicle battery research efforts so that the 1998 ZEV goal could be reached, the United States Advanced Battery Consortium (USABC) was formed in 1991. The USABC is comprised of the "big three" American automobile manufacturers: Chrysler, Ford, and General Motors, along with the United States Department of Energy and the Electric Power Research Institute, which is a research organization funded by utilities in the United States. The goals of the USABC are to expand battery technology in order to mass-produce electric vehicles potentially in this decade, and also to produce electric-vehicle batteries with performance competitive with today's internal combustion engines by early in the next decade.

Important performance factors characterizing electric-vehicle batteries include energy density and specific energy, along with power density and specific power. The energy density is the energy the battery can store per unit volume, and the specific energy is its energy storage per unit mass. The power density and specific power of the battery are important factors in vehicle acceleration. The power density is the maximum power per unit volume that the battery can deliver at a specified depth-of-discharge, and the specific power is the power per unit mass. For electric-vehicle applications, we

would like to minimize both the weight and the volume of the batteries, while maximizing energy and power.

Another important characteristic of electric-vehicle batteries is their cycle life. The cycle life of the battery is the number of times that the battery can be discharged and charged before a significant loss of performance occurs. For rechargeable batteries cycle life should be maximized.

The USABC has specified mid-term and long-term goals for electric-vehicle battery technology. Mid-term goals are to be met by the end of this decade, and the long-term goals are to be met by sometime in the next decade. These goals are given in Table 1-1.

Table 1-1. USABC goals for electric-vehicle battery technology.

	Mid-Term	Long-Term
Specific Energy (W·h/kg)*	80	200
Specific Power (W/kg)†	150	400
Cycle Life‡	600	1000
Battery Cost ($/kW·h)	Under 150	Under 100

* At the three-hour discharge rate.
† At 80% depth-of-discharge for 30 seconds.
‡ When discharged to 80% depth-of-discharge on each cycle.

1.2 Lithium/Polymer Batteries

In line with the goals of the USABC given above, lithium batteries have emerged as one of the most promising options for both mid-term and long-term electric-vehicle batteries. Lithium is the lightest and most electronegative of the alkali metals, and therefore lithium batteries are expected to have a high theoretical energy density.

Since reversible intercalation reactions are utilized in most lithium batteries, the batteries will also have high cycle lives.

Despite being an active research area for over a decade, only within the last few years have rechargeable batteries based on solid lithium[2-4] reached the consumer market. This has been partly attributed to concerns over the safety of the solid lithium electrode. These concerns are likely to become even more important for larger-scale electric-vehicle batteries.

It is advantageous to use polymer electrolytes in lithium batteries for several reasons. Although polymer electrolytes have a much lower ionic conductivity than liquid electrolytes, they allow the battery to have high energy density because they can be manufactured as thin films. Since the thin-film polymer acts as electrolyte, separator, and adhesive between layers of the thin-film battery, eliminating the need for other battery components, the energy and power densities are predicted to be high.[5]

1.3 Thermal Modeling

Heat is generated in batteries during discharge due to both reversible and irreversible phenomena.[6] Thermal management is particularly important in all-solid-state thin-film polymer-electrolyte batteries because heat transport out of the battery is made difficult by the low thermal conductivity of the polymer, and by the thin-film arrangement. Since the ionic conductivity and salt diffusion coefficient in the polymer vary exponentially with temperature, the performance of the battery is greatly affected by its temperature.

Computer simulations are useful in the design of batteries because of the possible savings in both time and materials. Modeling can also provide information that is difficult or impossible to obtain experimentally. Thermal modeling will help us

predict how the battery behavior will vary with temperature, as well as help us design a thermal-management system.

Doyle *et al.*[7] presented a one-dimensional isothermal model for the behavior of the lithium/polymer/insertion cell. Their model describes the performance of the cell for different current densities and gives detailed concentration profiles. Bernardi *et al.*[8] presented a general energy balance for battery systems. In this energy balance a uniform temperature is assumed throughout the cell, but the temperature is allowed to vary with time. The complete energy balance accounts for reactions in the cell, changes of heat capacity of the system, phase changes, mixing, electrical work, and heat transfer with the surroundings.

Two thermal models of the lithium/polymer battery are presented in this thesis. The first, the one-cell model, predicts the galvanostatic discharge behavior of the lithium/polymer cell for adiabatic and isothermal operation, as well as for limited cases of heat transfer. The second model, the cell-stack model, uses heat-generation rates calculated by the one-cell model to predict temperature profiles in cell stacks, and is useful in designing a thermal-management system.

The one-cell model is a combination of the works of Doyle *et al.*[7] and Bernardi *et al.*[8] presented above to study the effects of temperature rise on the galvanostatic discharge of the lithium/polymer cell. Using this model, battery performance at different temperatures can be assessed, and battery operating temperature can be optimized. This model is helpful in designing a thermal management system for the battery, as the heat released during operation is predicted. Temperature rise in a cell affects its performance because transport properties depend on temperature. In the one-cell model, the salt diffusion coefficient and the ionic conductivity vary with temperature. Other models have neglected the effect of temperature changes on the performance of the cell.

The cell-stack model uses heat-generation rates calculated by the one-cell model to predict one-dimensional temperature profiles of cell stacks in the direction perpendicular to cell layers. Since the heat-generation rate varies with time, temperature, and position in the cell stack, the cell-stack model will give us useful information about heat transport in cell stacks not attainable from the one-cell model or previous thermal models discussed below. Required heat-transfer coefficients, as well as maximum cell-stack thicknesses can be predicted by the cell-stack model. Results of both models are presented for a specific lithium/polymer cell configuration.

Chen and Evans[9,10] have presented two thermal models of lithium/polymer batteries. Their models focus on the two-dimensional heat transport inside the cell stack. Heat generation is calculated using experimental discharge curves. However, unlike under the present approach, in their work the electrochemistry of the cell is decoupled from the thermal model; thus, the effect of temperature changes on the performance of the cell can not be assessed.

Newman and Tiedemann[11] presented a three-dimensional analytical model for temperature rise at the center of a battery module assuming a heat-generation rate that is uniform throughout the module, but is allowed to vary with time. Temperature profiles were not presented. Their model assumes that the temperature at the edge of the cell stack remains constant throughout discharge. This assumption may not be valid for large temperature changes and high energy-generation rates. Like the model of Chen and Evans, this model also neglects the effect of temperature change on battery performance.

1.4 Conclusion

In response to regulations imposed by the California Air Resources Board, advanced electric-vehicle batteries are quickly being developed. Because of its high

energy density and design flexibility, one promising electric-vehicle battery is the lithium/polymer battery. Thermal considerations are especially important for the lithium/polymer system because of its thin-film configuration and the low thermal conductivity of the polymer.

Two thermal models of the lithium/polymer battery are presented in this thesis. The first model, the one-cell model, predicts how the battery will respond to temperature changes caused by discharge. The second model, the cell-stack model, predicts temperature profiles in batteries using heat-generation rates calculated from the one-cell model. The advantage of the cell-stack model over previous models is that the heat-generation rate is allowed to vary with time and position in the cell stack.

References

1. *Draft Technical Document for the Low-Emission Vehicle and Zero-Emission Vehicle Workshop on March 15, 1994; Zero Emission Vehicle Update*, California Air Resources Board.

2. Sony's Lithium Manganese Rechargeable Battery (AA Size), *JEC Press, Inc.*, p. 26 (February, 1988).

3. Moli Energy's New Product Data Sheet, *JEC Battery Newsletter*, No. 6, p. 15 (1988).

4. Sanyo's Lithium Rechargeable Battery, *ibid.*, No. 3, p. 15 (1989).

5. M. Gauthier, A. Bélanger, B. Kapfer, and M. Armand, "Solid Polymer Electrolyte Lithium Batteries," Chapter 9 in *Polymer Electrolyte Reviews*, edited by J. R. MacCallum and C. A. Vincent, Elsiever Applied Science, New York, NY (1987).

6. John S. Newman, *Electrochemical Systems*, Prentice Hall, Engelwood Cliffs, New Jersey (1991).

7. M. Doyle, T. F. Fuller, and J. Newman, "Modeling of the Galvanostatic Charge and Discharge of the Lithium/Polymer/Insertion Cell," *J. Electrochem. Soc.*, **140**, 1526 (1993).

8. D. Bernardi, E. Pawlikowski, and J. Newman, "A General Energy Balance for Battery Systems," *J. Electrochem. Soc.*, **132**, 5 (1985).

9. Y. Chen and J. Evans, "Heat Transfer Phenomena in Lithium/Polymer-Electrolyte Batteries for Electric Vehicle Application," *J. Electrochem. Soc.*, **140**, 1833 (1993).

10. Y. Chen and J. Evans, "Thermal Analysis of Lithium Polymer Electrolyte Batteries by a Two Dimensional Model--Thermal Behaviour and Design Optimization," *Elctrochimica Acta*, **39**, 517 (1994).

11. J. Newman and W. Tiedemann, "Temperature Rise in a Battery Module with Constant Heat Generation," Submitted to *J. Electrochem. Soc.*

Chapter 2: One-Cell Model Development

2.1 Introduction

This chapter presents the development of the one-cell model. The one-cell model is a modified version of the isothermal model given by Doyle *et al.*[1] with the addition of an energy balance in the form given by Bernardi *et al.*[2] First, the cell that is modeled is described, and its structure is explained. Next, a detailed summary of the model of Doyle *et al.*, along with the general energy balance given by Bernardi *et al.*, is presented, and applications to the thermal model are explained. Finally the variation of transport properties with temperature is given.

The one-cell model allows us to study the adiabatic and isothermal behavior of the system. It also allows us to study heat-transfer behavior in thin cells, those which can be assumed to have no temperature gradient. Results, along with an explanation of limiting cases, will be presented in Chapter 3.

2.2 The Cell

Figure 2-1 shows a schematic of the lithium/polymer cell sandwich that we are considering. The full cell consists of a negative electrode, a separator, a positive electrode, and a bipolar partition. The model is general and can be applied to many lithium/polymer battery systems. Results presented in this thesis are for the Li|PEO_{18}-$LiCF_3SO_3$|TiS_2 system. In this TiS_2 cell, the negative electrode is a lithium foil, and the separator consists of an inert polymer, polyethylene oxide (PEO), which acts as a solvent for the lithium salt, $LiCF_3SO_3$. The composite positive electrode is made up of the insertion electrode material, polymer electrolyte, and inert filler which provides additional electronic conductivity. The insertion electrode material in the composite

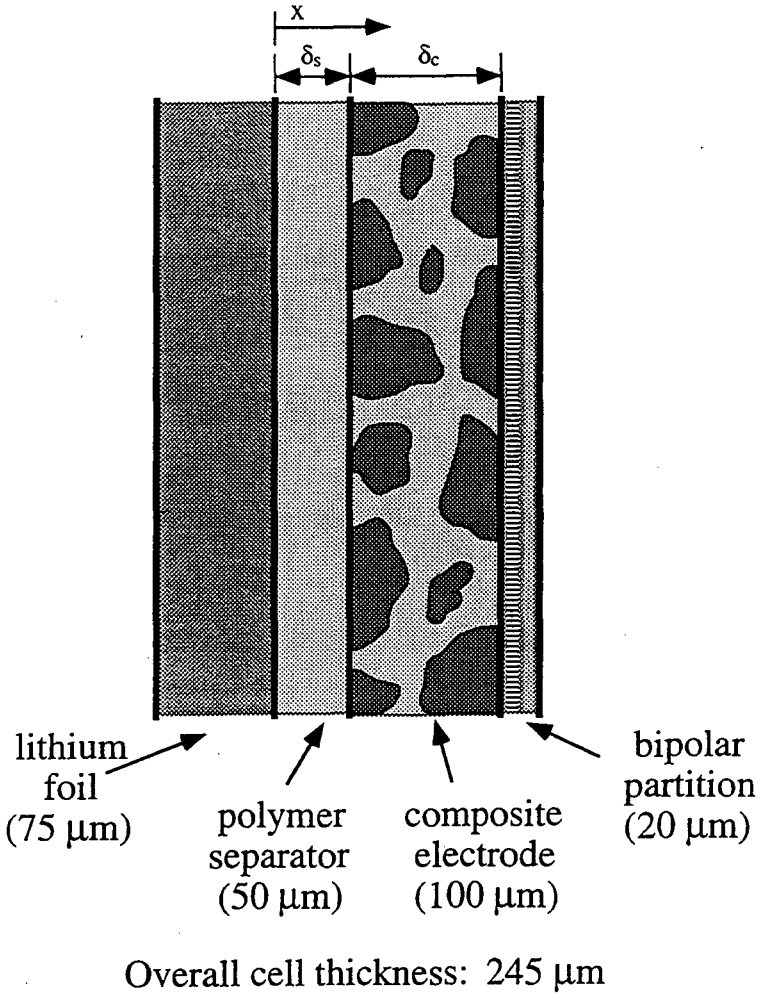

Figure 2-1. Schematic of the lithium negative electrode/ solid polymer separator/insertion positive electrode cell. Dimensions of cell layers are given on the figure.

positive electrode is TiS_2, and the inert filler is carbon black. The bipolar partition is made up of a layer of copper and a layer of aluminum. To maintain good contact between cell components, cells are placed under pressure.

2.3 The Model of Doyle *et al.*

The one-cell thermal model is a modified version of the isothermal, one-dimensional lithium/polymer/insertion cell model presented by Doyle *et al.*[1] As summarized in detail below, this model considers transport of Li^+ from the lithium negative electrode, through the polymer separator, and its insertion into the composite positive electrode. The model is general, and several lithium/polymer battery systems can be considered. Concentrated solution theory is used to describe transport in the polymer electrolyte. A binary electrolyte and single-phase polymer solvent are assumed, and physical properties are allowed to vary with concentration. The composite insertion electrodes are modeled using porous electrode theory, assuming Butler-Volmer kinetics for the insertion process. The model includes diffusion limitations inside the solid insertion particles. This earlier model is isothermal; it includes no energy balance.

In the model of Doyle *et al.*, concentrated solution theory is used to model transport in the polymer electrolyte. The gradient in electrochemical potential is the driving force for transport in concentrated solution theory.[3] The equation describing transport in the polymer phase is

$$c_i \nabla \mu_i = \sum_{j \neq i} K_{ij}(v_j - v_i). \tag{2-1}$$

The K_{ij} are frictional coefficients that account for interactions between species i and j, and can be related directly to the transport properties D, t_+^0, and κ.[4] For a polymer

electrolyte, we assume that the reference velocity is that of the polymer, which is taken to be zero. For a polymer solvent with a binary electrolyte, two equations of the form of Equation 2-1 are inverted to obtain flux equations for the ionic species.

$$\mathbf{N}_+ = -\nu_+ D\nabla c + \frac{\mathbf{i}t_+^0}{z_+F} \quad \text{and} \quad \mathbf{N}_- = -\nu_- D\nabla c + \frac{\mathbf{i}t_-^0}{z_-F}. \tag{2-2}$$

Doyle *et al.* give a material balance on the salt in the polymer separator;

$$\frac{\partial c}{\partial t} = \nabla \bullet \left[D(c)\left(1 - \frac{d\ln c_0}{d\ln c}\right)\nabla c\right] - \frac{\mathbf{i}_2 \bullet \nabla t_+^0(c)}{z_+\nu_+F}. \tag{2-3}$$

Next it is assumed that the partial molar volume of the solvent is constant. This assumption implies that the solution density is a linear function of salt concentration.[3] The potential variation in the separator is given by

$$\mathbf{i}_2 = -\kappa(c)\nabla\Phi_2 - \frac{\kappa(c)RT}{F}\left(1 + \frac{\partial \ln f_A}{\partial \ln c}\right)\left(\frac{s_+}{n\nu_+} + \frac{t_+^0(c)}{z_+\nu_+}\right)\nabla \ln c \tag{2-4}$$

where Φ_2 is measured with a lithium reference electrode. For the $Li|PEO_{18}LiCF_3SO_3|TiS_2$ system, the activity coefficient data, $f_A(c)$, are not known.

Following Sequeria *et al.*,[5] Doyle *et al.* assume that the reaction at the negative electrode is given by

$$Li + \Theta_p \leftrightarrow Li^+\text{—}\Theta_p + e^-$$

where Θ_p represents a reaction site in the polymer lattice. The total number of sites available is assumed to correspond to the solubility limit of the lithium salt in the

polymer solvent. The reaction is assumed to follow Butler-Volmer kinetics. The Butler-Volmer equation is given by

$$I = i_{o1}\left[\exp\left(\frac{\alpha_{a1}F\eta_{s1}}{RT}\right) - \exp\left(-\frac{\alpha_{c1}F\eta_{s1}}{RT}\right)\right] \tag{2-5}$$

The exchange current density for the TiS_2 system is given by Sequeria *et al.*[5] The form of the exchange current density is

$$i_{o1} = F(k_{a1})^{\alpha_{c1}}(k_{c1})^{\alpha_{a1}}(c_{max} - c)^{\alpha_{c1}}(c)^{\alpha_{a1}} \tag{2-6}$$

where k_{a1} and k_{c1} are reaction rate constants for the anodic and cathodic reactions, respectively. The local value of the surface overpotential, η_{s1}, is given by

$$\eta_{s1} = \Phi_1 - \Phi_2 - U_1 \tag{2-7}$$

where U_1, the theoretical open-circuit potential, is zero when using a Li reference electrode.

At the lithium foil, x=0, the potential of the solid lithium is set to zero. The other boundary conditions are that the lithium ion flux is equal to the current at the interface,

$$N_+ = \frac{I}{F} \text{ at } x = 0, \tag{2-8}$$

and that the flux and concentration of each species, as well as the potential in the solution phase, are continuous at $x = \delta_s$.

Doyle *et al.* use porous electrode theory to describe transport in the composite positive electrode. The three phases of the composite electrode in the TiS_2 system are

assumed to be superimposed continua, so that a material balance on the salt in the polymer phase gives

$$\varepsilon\frac{\partial c}{\partial t} = \nabla \bullet \left(\varepsilon D(c)\nabla c\right) - \frac{\mathbf{i}_2 \bullet \nabla t_+^0(c)}{z_+ \nu_+ F} + \frac{a j_n\left(1 - t_+^0\right)}{\nu_+} \tag{2-9}$$

where ε is the volume fraction of the polymer. In this expression j_n represents the pore wall flux, which is the flux of lithium ions from the solid phase entering the solution phase. The pore wall flux is given by

$$a j_n = \frac{-s_i}{nF} \nabla \bullet \mathbf{i}_2 \tag{2-10}$$

In the composite electrode, the ionic conductivity is assumed to follow the Bruggeman relationship[6]

$$\kappa_{eff} = \kappa\varepsilon^{1.5}, \tag{2-11}$$

and the diffusion coefficient in the polymer phase is given by its effective value

$$D_{eff} = D\varepsilon^{0.5}. \tag{2-12}$$

At the positive electrode/current collector boundary, the flux of each species is taken to be zero.

$$\mathbf{N}_i = 0 \quad \text{at} \quad x = \delta_s + \delta_+ \tag{2-13}$$

The insertion material in the composite electrode is assumed to be made up of spherical particles of radius R_S. Diffusion of lithium ions into the spherical particles is given by

$$\frac{\partial c_s}{\partial t} = D_s\left[\frac{\partial^2 c_s}{\partial r^2} + \frac{2}{r}\frac{\partial c_s}{\partial r}\right]. \tag{2-14}$$

D_S, the diffusion coefficient in the solid phase, is assumed constant. The first boundary condition for this expression is that there is no flux of ions across the center of the particles,

$$\frac{\partial c_s}{\partial r} = 0 \text{ at } r = 0 \tag{2-15}$$

The other boundary condition gives the relationship between the flux of lithium ions into the insertion particles and the rate of lithium ion diffusion to the surface of those particles

$$j_n = -D_s\frac{\partial c_s}{\partial r} \text{ at } r = R_s \tag{2-16}$$

This linear problem is solved using the method of superposition independently of the solution phase.

The reaction in the positive electrode is a non-stoichiometric intercalation reaction given by

$$y\,Li^+ + y\,e^- \leftrightarrow Li_yTiS_2$$

where $0 \leq y \leq 1$. In the model, this reaction is assumed to follow

$$Li^+\text{—}\Theta_p + \Theta_s + e^- \leftrightarrow Li^+\text{—}\Theta_s + \Theta_p,$$

leading to the following kinetic expression:

$$i = Fk_2(c_{max} - c)^{\alpha_c} c^{\alpha_a}\left[c_s \exp\left(\frac{\alpha_a F}{RT}(\eta - U')\right) - (c_T - c_s)\exp\left(-\frac{\alpha_c F}{RT}(\eta - U')\right)\right]. \quad (2\text{-}17)$$

In this expression, k_2 is an effective rate constant for the charge-transfer reaction. For the TiS_2 system, k_2 is not known, so a large value was assumed corresponding to a nearly reversible situation. For the TiS_2 system, U' is fit from experimental data as a linear function of solid concentration[7]

$$U' = 2.17 + \frac{RT}{F}(-0.000558c_s + 8.10) \quad (2\text{-}18)$$

where U' is in V and c_s is in mol/m^3. The overpotential in Equation 2-17 is defined as

$$\eta = \Phi_1 - \Phi_2. \quad (2\text{-}19)$$

The boundary condition for the potential at the separator is

$$\nabla\Phi_1 = 0 \text{ at } x = \delta_s \quad (2\text{-}20)$$

The current in the matrix is dictated by Ohm's law,

$$\mathbf{i}_1 = -\sigma\nabla\Phi_1. \quad (2\text{-}21)$$

The current in the two phases is conserved through a charge balance,

$$\nabla \bullet (\mathbf{i}_1 + \mathbf{i}_2) = 0, \quad (2\text{-}22)$$

which leads to the integrated form

$$I = \mathbf{i}_1 + \mathbf{i}_2. \quad (2\text{-}23)$$

These equations are linearized and solved simultaneously using the subroutine BAND.[3] There are two independent variables (x and t) and six dependent variables (c, Φ_2, c_s, i_2, j_n, and Φ_1). The Crank-Nicolson implicit method was used to evaluate the time derivatives.

2.4 Energy Balance

The model of Doyle *et al.* is modified for thermal calculations by adding an energy balance in the form of Bernardi *et al.*[2] In this energy balance, the temperature is assumed to be uniform throughout the cell and is allowed to vary with time. The general energy balance takes into account energy changes associated with chemical reactions, heat capacity changes, phase changes, mixing, electrical work, and heat transfer with the surroundings. The energy balance equation is given below:

$$
\begin{aligned}
q - IV = \; & \frac{1}{A_s}\sum_l \left[I_l T^2 \frac{d\frac{U_{l,avg}}{T}}{dT} \right] && \text{enthalpy-of-reaction} \\
& - \frac{1}{A_s}\sum_k \frac{d}{dt}\left[\int_{v_k} \sum_i c_{i,k} R T^2 \frac{\partial}{\partial T} \ln\left(\frac{f_{i,k}}{f_{i,k}^{avg}}\right) dv_j \right] && \text{enthalpy-of-mixing} \\
& - \frac{1}{A_s}\sum_{k,k\neq m}\sum_i \left[\left(\Delta\tilde{H}^o_{i_{k\to m}} - RT^2 \frac{\partial}{\partial T} \ln\frac{f_{i,k}}{f_{i,k}^{avg}} \right) \frac{dn_{i,k}}{dt} \right] && \text{phase-change} \\
& + \frac{1}{A_s}\frac{dT}{dt}\left[\sum_k\sum_i n^o_{i,k}\overline{C}^{avg}_{p_{i,k}} + \sum_l \frac{\int_0^t I_l dt}{n_l F}\Delta C_{p_l} + \sum_{k,k\neq m}\sum_i \left(\overline{C}^{avg}_{p_{i,k}} - \overline{C}^{avg}_{p_{i,m}}\right)\left(n_{i,k} - n^o_{i,k}\right) \right] \\
& && \text{heat-capacity} \qquad (2\text{-}24)
\end{aligned}
$$

where

$$\Delta C_{p_l} = \sum_i s_{i,l} \overline{C}^{avg}_{p_{i,m}}, \tag{2-25}$$

and

$$\Delta \widetilde{H}^o_{i_{k\to m}} = \widetilde{H}^o_{i,m} - \widetilde{H}^o_{i,k}. \tag{2-26}$$

In Equation 2-24, q represents heat transfer to the surroundings per unit separator area and can be expressed as

$$q = - h^*(T - T_a). \tag{2-27}$$

The per-separator-area heat-transfer coefficient, h*, is based on separator area and T_a is the ambient temperature.

When applying the general energy balance to the lithium/polymer electrolyte battery, several simplifications can be made. First, we have assumed that the lithium electrode reaction is the only important reaction. We have also assumed that the phase-change and enthalpy-of-mixing terms are negligible. Last, we make the assumption that the heat capacity of the battery is an average value that is invariant with temperature and state of charge.

We have assumed that only one reaction, the lithium electrode reaction, contributes substantially to the cell energy generation. This is true for the lithium/ polymer system during normal discharge. Side reactions are suspected of causing degradation of battery performance at high temperatures, but these reactions and effects have not been quantified, and are not included. Recently, there have been some attempts to measure the heat generated due to side reactions, and this may be eventually included in the calculations.[8]

The enthalpy-of-mixing term is neglected because activity coefficient data are not available for this system, causing the term to fall out of the equation. The enthalpy-of-mixing term is important when concentration profiles are formed or relaxed in the cell. Details of relaxation phenomena in the lithium/polymer system are given by Fuller *et al.*[9]

The phase-change term is also negligible when considering the lithium/polymer cell. Bernardi *et al.*[2] show an example of the general energy balance applied to the LiAl/FeS cell. For the LiAl/FeS cell, phase-change terms do not exist until KCl precipitation takes place in the liquid electrolyte. In the lithium/polymer system that we are considering, the salt concentration used is below the saturation concentration so that there is no formation of bulk amounts of the lithium salt in the electrolyte. The phase-change term will be important if the battery is heated to the melting temperature of lithium, 186°C, but we shall assume that the battery does not reach this temperature during normal operation.

For most practical battery applications, according to Bernardi *et al.*, the heat capacity of the battery does not change substantially during operation. Since we believe that this is true of the lithium/polymer system, the heat-capacity term has been replaced by an average value of the cell heat capacity.

When only one reaction is considered with the above simplifications, the energy balance becomes

$$h^*(T - T_a) = I\left(U - V - T\frac{dU}{dT}\right) - \frac{M\hat{C}_p}{A_s}\frac{dT}{dt}. \qquad (2\text{-}28)$$

Equation 2-28 is the energy balance equation used in our model. The energy balance in this form was used by Pollard and Newman[10] in a model of the LiAl/FeS cell. This equation is solved at each time step assuming $\frac{dT}{dt} = \frac{\Delta T}{\Delta t}$.

The cell heat-generation rate, Q, which is equal to the sum of q, the heat transferred out of the system, and the heat stored in the system, is calculated from

$$Q = I\left(U - V - T\frac{dU}{dT}\right) = h^*(T - T_a) + \frac{M\widehat{C}_p}{A_s}\frac{dT}{dt}. \qquad (2\text{-}29)$$

The term I (U - V) is the heat produced due to cell polarization, and the term -IT(dU/dT) is due to the reversible entropy change in the cell. For the simulations of the TiS_2 cell, the dU/dT term is set to zero because open-circuit potential data as a function of temperature are not currently available.

The computer program and input file for the one-cell model are presented in Appendix A.

2.5 Physical Properties for the $Li|PEO_{18}\text{-}LiCF_3SO_3|TiS_2$ System

The transport properties of the system will vary with temperature. Temperature variation of the salt diffusion coefficient and of the ionic conductivity is presented below. The thermal conductivity of the system is calculated assuming that it is temperature-invariant. Other important physical properties of the cell, assumed to be temperature-invariant, are given.

<u>Diffusion Coefficient.</u> The diffusion coefficient of an ionic species in solution is related to the temperature and solvent viscosity by the Stokes-Einstein relation,

$$\frac{D\mu}{T} = \text{constant}. \qquad (2\text{-}30)$$

In this system, the solvent is a polymer. Billmeyer[11] gives the WLF equation for the viscosity of a polymer as a function of temperature as

$$\log \mu = 3.4 \log \overline{Z}_w - \frac{17.44\,(T - T_g)}{51.6 + (T - T_g)} + k'. \qquad (2\text{-}31)$$

Equation 2-31 is generally valid for a temperature range from T_g to T_g + 100 K. Since the glass transition-temperature of PEO is taken to be -60°C,[12] this equation would be valid only for temperatures up to 40°C. Although the temperatures of operatation of the battery are out of this range, Equation 2-31 is used because we are not concerned with the absolute value of the polymer viscosity, we are concerned only with the ratio of polymer viscosities at different temperatures. Combining Equations 2-30 and 2-31, the diffusion coefficient is given as a function of temperature by

$$D = 2.2\times10^{-14}\, T \cdot \exp\left(\frac{40.16\,(T - T_g)}{51.6 + (T - T_g)} - 29.93\right). \qquad (2\text{-}32)$$

The diffusion coefficient is in m^2/s, and temperatures are in kelvin. Notice that we do not need to use $\overline{Z}_w$ or k' in this calculation. The reference value of the salt diffusion coefficient is taken to be 8×10^{-12} m^2/s at 90°C.[13] Figure 2-2 shows this function as an Arrhenius plot for the diffusion coefficient over the temperature range of interest for lithium/polymer battery applications.

Since the functional form of the diffusion coefficient is almost a straight line on Figure 2-2, we can determine an Arrhenius rate expression. The Arrhenius expression for the diffusion coefficient as a function of temperature is:

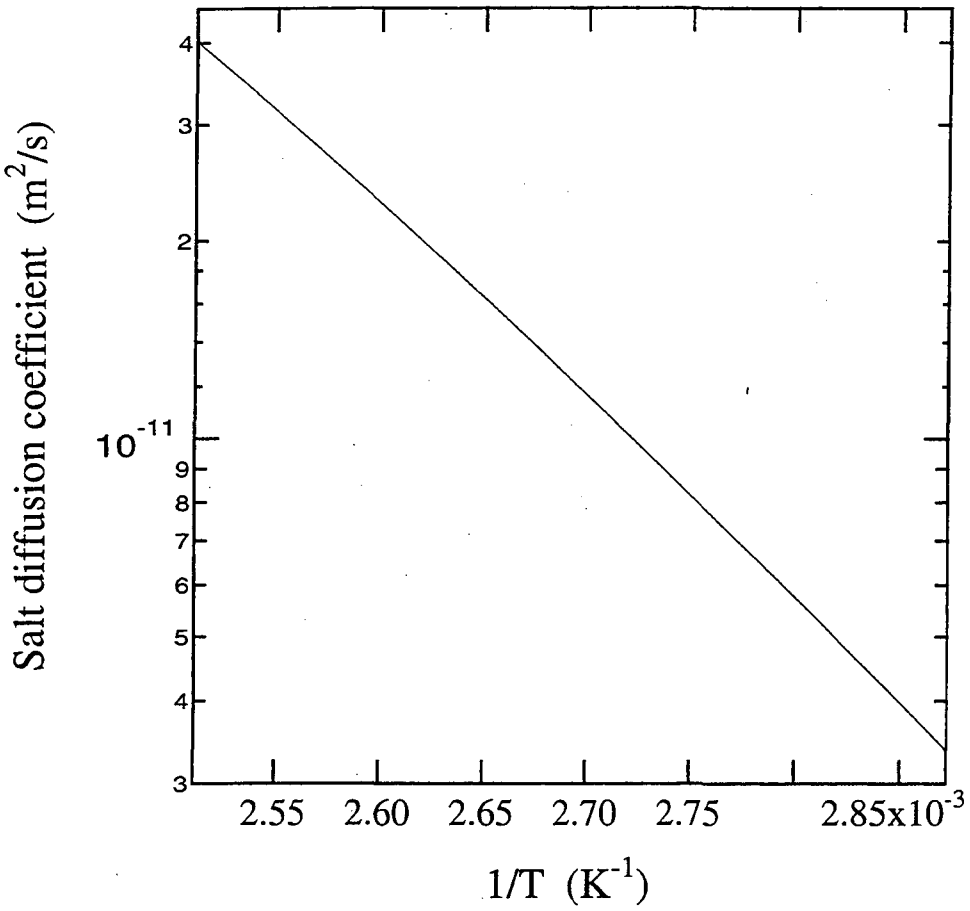

Figure 2-2. Approximation to the temperature dependence of the salt diffusion coefficient in polyethylene oxide. The experimental value at 90°C of 8×10^{-12} m^2/s was used as the reference point.

$$D = 1\times10^{-3}\exp\left(-\frac{6770}{T}\right) \tag{2-33}$$

Ionic Conductivity. Robitaille and Fauteaux[14] give the ionic conductivity of $LiCF_3SO_3$ in polyethylene oxide as a function of temperature and mass fraction of salt. The mass fraction was converted to concentration, and the data were fit to an analytic function of temperature. The concentration and temperature dependence of the ionic conductivity (κ, S/cm) is given by the expression

$$\kappa = c\exp\left[2.303(\,f(c) + g(c,T)\,)\right], \tag{2-34}$$

where

$$\begin{aligned} f(c) = &\left(\frac{1+\tanh(a_1 - c^{0.5})}{2}\right)\left(a_2 + a_3c^{0.5} + a_4c + a_5c^{1.5} + a_6c^2\right) + \\ &\left(\frac{1+\tanh(c^{0.5} - a_7)}{2}\right)\left(a_8 + a_9c^{0.5} + a_{10}c + a_{11}c^{1.5} + a_{12}c^2\right), \end{aligned} \tag{2-34a}$$

and

$$g(c,T) = \left(b_1 + b_2c^{0.5} + b_3c + b_4c^{1.5}\right)\left(\frac{1}{T} - \frac{1}{T_0}\right). \tag{2-34b}$$

The concentration, c, is in mol/m^3, and the temperatures are in kelvin. The reference temperature, T_0, is 358 K. The values for the constants are given in Table 2-1. Figure 2-3 shows the fit, where the symbols denote data points from reference 14 and the lines are calculated from Equation 2-34.

Table 2-1. Constants for conductivity fit.

a_1	32.1374
a_2	-6.60932
a_3	-0.0251516
a_4	0.0259545
a_5	-0.0019001
a_6	3.41711×10^{-5}
a_7	32.1262
a_8	-44.5889
a_9	2.59756
a_{10}	-0.0631626
a_{11}	6.40897×10^{-4}
a_{12}	-2.35565×10^{-6}

b_1	-1052.65
b_2	-182.053
b_3	3.74936
b_4	0.0242493

Thermal Conductivity The thermal conductivity of the cell was calculated by assuming parallel resistances for each cell layer. The equation giving this thermal conductivity for heat transport perpendicular to cell layers is

$$\frac{l_{cell}}{k_{cell}} = \sum_i \frac{l_i}{k_i} \tag{2-35}$$

where i denotes cell layers. The thermal conductivity of the composite positive electrode is taken to be a combination of series and parallel resistances. Meredith and Tobias[15] give the conductivity of a system of uniform spheres arranged in a cubic lattice:

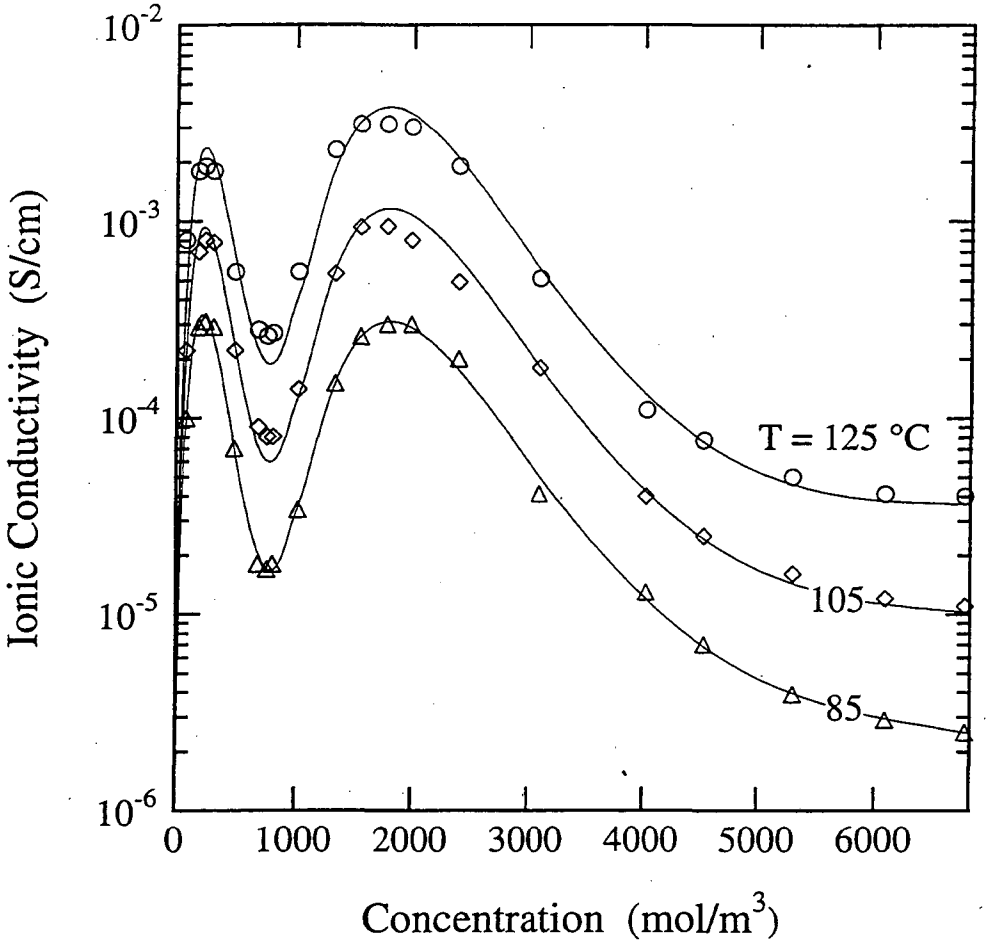

Figure 2-3. Fit of the ionic conductivity as a function of salt concentration. Symbols denote data points from reference 14. Lines represent the fit given by Equation 2-34.

$$K_+ = 1 - \frac{3\varepsilon}{\frac{(2+K_d)}{(1-K_d)} + \varepsilon - \frac{1.315(1-K_d)\varepsilon^{10/3}}{\frac{4}{3} + K_d + 0.409(1-K_d)\varepsilon^{7/3}}} \tag{2-36}$$

where ε is the porosity of the discrete phase,

$$K_+ = \frac{k_+}{k_c}, \tag{2-37}$$

and

$$K_d = \frac{k_d}{k_c}. \tag{2-38}$$

c refers to the continuous phase, d to the discrete phase, and + to the composite electrode. The continuous phase is assumed to be the polymer, and the discrete phase is assumed to consist of both the solid insertion electrode material and inert filler. The thermal conductivies of the solid and filler are assumed to be equal when using this equation. The thermal conductivity values used for the calculations are given in Table 2-2.

The heat capacity and mass of the cell are average values based on the system components. For the cell considered here, $\widehat{C}_p$ = 746.7 J/kg·K and M/A_s = 0.46 kg/m^2. Other physical properties and important simulation parameters are given in Table 2-3.

Table 2-2. Thermal Conductivity values used in calculations.

Component	k (W/m·K)	Reference
Lithium	81.8	16
Copper	384	16
Aluminum	229	16
TiS_2*	6.53	16
PEO	0.16	17
Composite Electrode	1.47	Equation 2-36
Full Cell	0.64	Equation 2-35

*Assumed to be that of TiO_2.

Table 2-3. Parameters used in the simulations.

System specific			Design adjustable	
Parameter	Value	Reference	Parameter	Value
D_s	10^{-13} m²/s	18	T_a	89 °C
σ	60 S/m	†	δ_s	50 μm
$i_{o,-}$	1.26 mA/cm²	5,*	δ_c	100 μm
c_t	2372 mol/m³	‡	ε	0.4
ν_+,ν_-	1	-	ε_f	0.1
α_a,α_c	0.5	†	c^o	1500 mol/m³
k_2	1×10^{-12} m⁴/mol·s	†	R_s	2 μm

* Value reported at initial conditions
† Assumed values.
‡ Calculated based on the density of the insertion material.

List of Symbols

a	specific interfacial area, m^2/m^3
A_s	separator area, cm^2
c	concentration of electrolyte, mol/m^3
c_{max}	maximum concentration of electrolyte, mol/m^3
c_i	concentration of species i, mol/m^3
c_s	concentration of lithium in solid phase, mol/m^3
c_T	maximum concentration of lithium in solid phase, mol/m^3
C_p	heat capacity at constant pressure, J/K
$\overline{C}_p$	partial molar constant-pressure heat capacity, J/mol·K
$\widehat{C}_p$	mass-averaged constant-pressure heat capacity of cell, J/kg·K
D, D_s	diffusion coefficient of electrolyte and of lithium in the solid matrix, cm^2/s
f	activity coefficient
f_A	activity coefficient of the salt
F	Faraday's constant, 96,487 C/eq
h	convective heat-transfer coefficient, $W/m^2 \cdot K$
h^*	per-cell heat-transfer coefficient, $W/m^2 \cdot K$
$\widetilde{H}$	molar enthalpy, kJ/mol
$\mathbf{i}$	current density, mA/cm^2
i_o	exchange-current density, mA/cm^2
I	superficial current density, A/m^2
I_l	partial current of electrode reaction l, A
j_n	pore-wall flux across interface, mol/m^2-s
k'	constant depending on polymer type
k_{cell}	thermal conductivity of cell, W/m·K

l_{cell}	thickness of the cell, m
M	mass, kg
n	number of electrons transferred in electrode reaction
$n_{i,k}$	moles of species i in phase k, mol
$\mathbf{N}_i$	molar flux in x-direction of species i, mol/m^2·s
q	heat transfer to surroundings, W/m^2
Q	heat-generation rate per unit separator area, W/m^2
r	radial distance within a particle of active material, m
R	universal gas constant, 8.314 J/mol-K
R_s	radius of solid electrode particles, m
s_i	stoichiometric coefficient of species i in electrode reaction
t	time, s
t_i^0	transference number of species i
T	temperature, K
T_g	glass-transition temperature of polymer
U	open-circuit potential, V
$\mathbf{v}_i$	velocity of species i, m/s
V	cell potential, V
x	distance from negative electrode
z_i	charge number of species i
$\overline{Z}_w$	weight-average chain length of polymer, m
δ_s	thickness of separator, m
δ_+	thickness of positive electrode, m
ε	porosity of electrode
η	surface overpotential, V
Θ_p	site concentration in polymer
Θ_s	site concentration in solid matrix

κ	ionic conductivity of the electrolyte, S/cm
μ	viscosity of polymer, Pa·s
μ_i	electrochemical potential, J/mol
ν_+,ν_-	number of cations and anions into which a mole of electrolyte dissociates
σ	electrical conductivity of solid matrix, S/cm
Φ	electrical potential, V

Subscripts

a	ambient
e	electrolyte
f	filler
i	refers to a species
j	refers to a species
k	refers to a phase
l	refers to a reaction
m	refers to a phase
s	solid phase or separator
1	solid matrix
2	solution phase
+	positive electrode
-	negative electrode

Superscripts

0	solvent or initial condition

References

1. M. Doyle, T. F. Fuller, and J. Newman, "Modeling of the Galvanostatic Charge and Discharge of the Lithium/Polymer/Insertion Cell," *J. Electrochem. Soc.*, **140**, 1526 (1993).

2. D. Bernardi, E. Pawlikowski, and J. Newman, "A General Energy Balance for Battery Systems," *J. Electrochem. Soc.*, **132**, 5 (1985).

3. John S. Newman, *Electrochemical Systems*, Prentice Hall, Engelwood Cliffs, New Jersey (1991).

4. T. F. Fuller and J. Newman, "Experimental Determination of the Transport Number of Water in Nafion 117 Membrane," *J. Electrochem. Soc.*, **139**, 1332 (1992).

5. C. A. C. Sequeria and A. Hooper, "The Study of Lithium Electrode Reversibility against PEO-$LiCF_3SO_3$ Polymeric Electrolytes," *Solid State Ionics*, **9&10**, 1131 (1983).

6. D. A. G. Bruggeman, *Ann. Physik.*, **24**, 636 (1935).

7. K. West, T. Jacobsen, and S. Atlung, "Modeling of Porous Insertion Electrodes with Liquid Electrolyte," *J. Electrochem. Soc.*, **129**, 1480 (1982).

8. M. Arakawa, Y. Nemoto, S. Tobishima, and M. Ichimura, "Thermal stability of lithium anodes in an amorphous V_2O_5/Li battery system," *J. Power Sources,* **43-44**, 517 (1993).

9. T. F. Fuller, M. Doyle, and J. Newman, "Relaxation Phenomena in Lithium-Ion-Insertion Cells," *J. Electrochem. Soc.*, **141**, 982 (1994).

10. R. Pollard and J. Newman, "Mathematical Modeling of the Lithium-Aluminum, Iron Sulfide Battery, Part 1," *J. Electrochem. Soc.*, **128**, 491 (1981).

11. F. W. Billmeyer, *Textbook of Polymer Science,* John Wiley & Sons, New York, NY, p. 307 (1984).

12. M. Armand, "Current State of PEO-Based Electrolyte," in *Polymer Electrolyte Reviews-1*, J. R. MacCallum and C. A. Vincent, eds., Elsevier Applied Science, London (1987).

13. A. Bouridah, F. Dalard, D. Deroo, and M. B. Armand, "Potentiometric measurements of ionic transport parameters in poly(ethylene oxide)-LiX electrolytes," *J. Applied Electrochem.*, **17**, 625 (1987).

14. C. D. Robitaille and D. Fauteux, "Phase Diagrams and Conductivity Characterization of Some PEO-LiX Electrolytes," *J. Electrochem Soc.*, **133**, 315 (1986).

15. R. E. Meredith and C. W. Tobias, "Conduction in Heterogeneous Systems," Chapter 2 in *Advances in Electrochemistry and Electrochemical Engineering,* Volume 2, edited by C. W. Tobias. Interscience Publishers, New York, NY (1962).

16. *Handbook of Chemistry and Physics*, 63rd Editon. CRC Press (1978-1979).

17. B. Kapfer, M. Gauthier, and A. Belanger, "Thin Film Polymer-Electrolyte Technology Electric-Vehicle Battery Design Concept: Thermal Management Aspect," In *Primary and Secondary Lithium Batteries*, edited by K. M. Abraham and M. Salomon. Proceedings 91-3, The Electrochemical Society, Pennington, NJ (1991).

18. D. Guyomard and J. M. Tarascon, "Li Metal-Free Rechargeable $LiMn_2O_4$/Carbon Cells: Their Understanding and Optimization," *J. Electrochem. Soc.*, **139**, 937-948 (1992).

Chapter 3: One-Cell Model Results

3.1 Introduction

This chapter presents results of the one-cell model. First, cell-potential, concentration-profile, and heat-generation-rate results are presented for isothermal runs. Next, cell-potential, concentration-profile, temperature, and heat-generation-rate results are presented for adiabatic operation. Finally, simulations are presented for cell stacks under heat-transfer conditions along with specifications for their applicability.

3.2 Isothermal Discharge Behavior

Isothermal modeling is important because it helps us understand operation of the battery at different temperatures. In the isothermal calculations it is assumed that all of the heat generated in the cell is transferred out of the system without considering the details of how it is done. We are interested in the behavior of the battery at different temperatures for several reasons. First, it is important to see how the battery performance changes with temperature to design a thermal-management system. If a battery shows a strong dependence on the temperature of operation, the thermal-management system will have to be designed to have precise temperature control. On the other hand, if the temperature does not affect the battery operation strongly, the temperature-control system will not need to be so precise. Second, we would like to define a temperature range over which the battery should operate. This temperature range should be wide enough to allow for the design of a simple temperature-control system but at the same time be narrow enough so that the performance of the battery is not adversely affected. Somewhere inside this range should be an optimum temperature of operation defined from the isothermal data. Last, we are interested in the heat-generation data as a function of temperature for use in the cell-stack model. All of the

isothermal simulations presented are for a discharge rate of 1.1 mA/cm^2, the nominal three-hour discharge rate used for electric-vehicle applications.

Cell Potential. The cell potential as a function of utilization of positive electrode material and time for several isothermal galvanostatic discharges at a current density of 1.1 mA/cm^2 is shown in Figure 3-1. The dashed line is the open-circuit potential of the cell. This figure shows how the cell performance is affected by the temperature of operation. At higher temperatures, the cell potential is higher for a given value of active material utilization. The cell also utilizes more of the active material at higher temperatures. At 80°C, the cell utilizes approximately 40% of the positive electrode material before reaching the cut-off voltage of 1.7 V while at 120°C the utilization reaches 98% before the cut-off. This is because the diffusion coefficient of the salt and the ionic conductivity increase with temperature and thereby decrease the cell polarization. The time-average cell potential, however, varies only about 5% from its value of 2.0 V at 80 °C to 2.1 V at 120 °C.

Concentration Profiles. As we have found that diffusion limitations reduce the utilization of active material, we examine the effects of temperature changes in the cell on the concentration profiles in the solution phase. Figure 3-2 gives the solution-phase concentration profile across the full cell at the end of several different isothermal discharges. All of the discharges were carried out at the nominal three-hour discharge rate, I = 1.1 mA/cm^2. The first important point to notice is that the concentration reaches zero only for discharges below 90°C. This indicates that, for discharges above 90°C, the cell is no longer diffusion-limited and is instead ohmically-limited. This transition into an ohmically-limited system occurs even though both the conductivity and the diffusion coefficient depend exponentially on temperature. The system is diffusion-

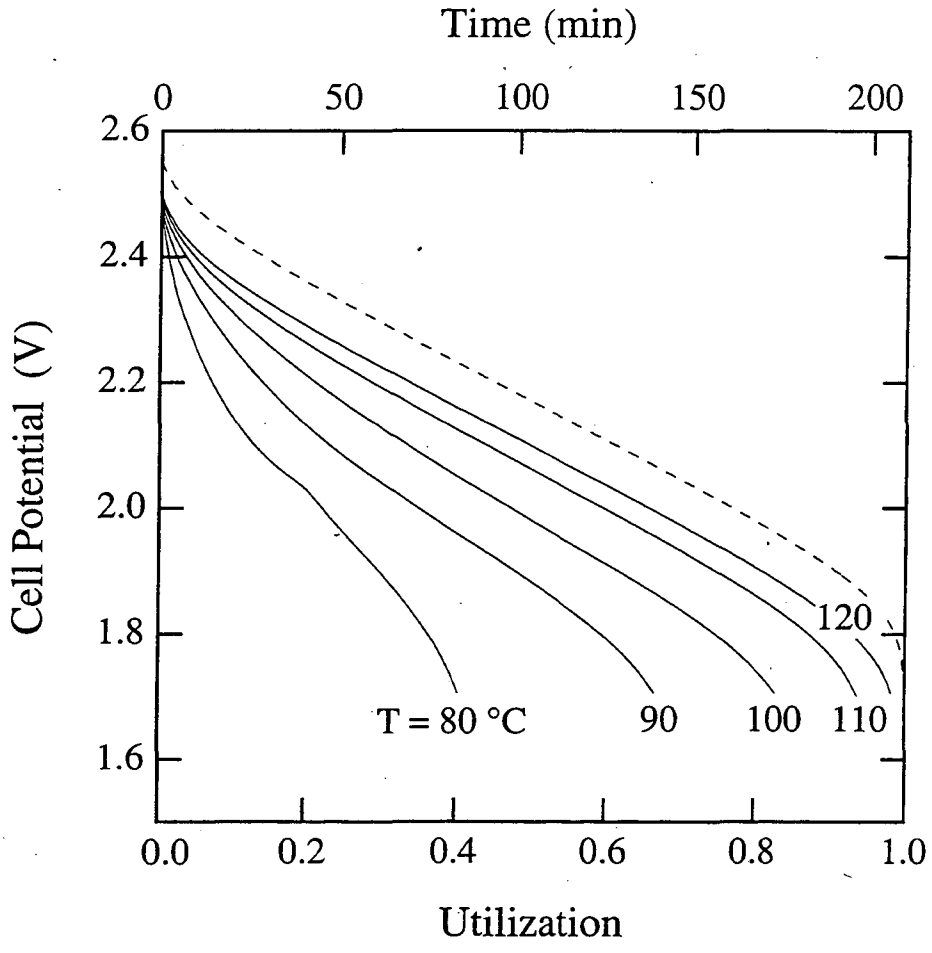

Figure 3-1. Cell potential as a function of utilization and time for isothermal discharge of the cell at I = 1.1 mA/cm^2 for several temperatures. The dashed line is the open-circuit potential of the cell.

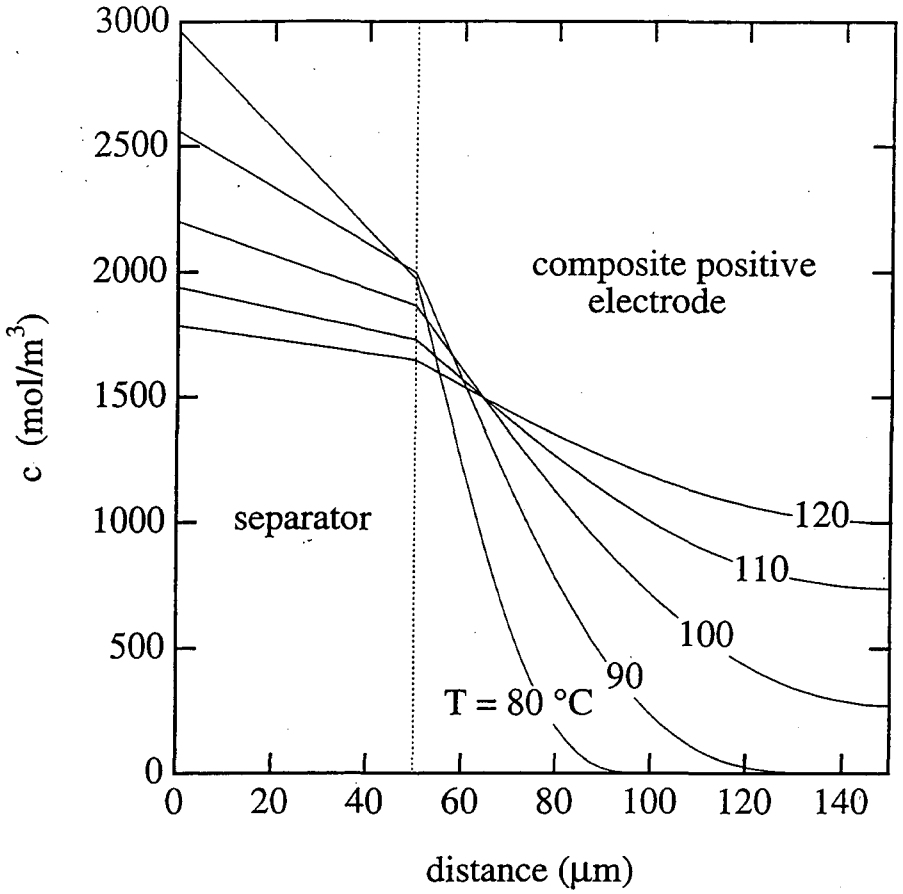

Figure 3-2. The solution-phase concentration profiles across the full cell at the end of discharge for isothermal galvanostatic discharge at I = 1.1 mA/cm^2 and various temperatures. Temperatures are indicated on the figure.

limited only if a limiting current is reached before the cell has reached 100% utilization of active material; this does not happen above a certain temperature.

Heat-Generation Rates. In Figure 3-3, the heat-generation rate as a function of time and utilization is given for the same simulations. This figure demonstrates that the heat-generation rate is much larger for lower temperatures than it is for higher temperatures. This result may also be seen by examining the cell-potential behavior, Figure 3-1. The heat-generation rate is equal to the product of the current density and the difference between the open circuit potential, the dashed line in Figure 3-1, and the cell potential, one of the solid lines in the figure. At lower temperatures, the conductivity is lower, and the larger ohmic drop leads to larger heat-generation rates. The heat-generation rate increases dramatically near the end of discharge for T = 80°C because of an increase in concentration overpotential.

Energy and Power Densities. For design purposes, it is useful to examine the temperature dependence of the energy and average power densities that the system can provide. The specific energy, E, is given by

$$E = \frac{A_s}{M}\int_0^t I\,V dt, \tag{3-1}$$

and the average specific power, P, is given by

$$P = \frac{A_s}{M\,t}\int_0^t I\,V dt, \tag{3-2}$$

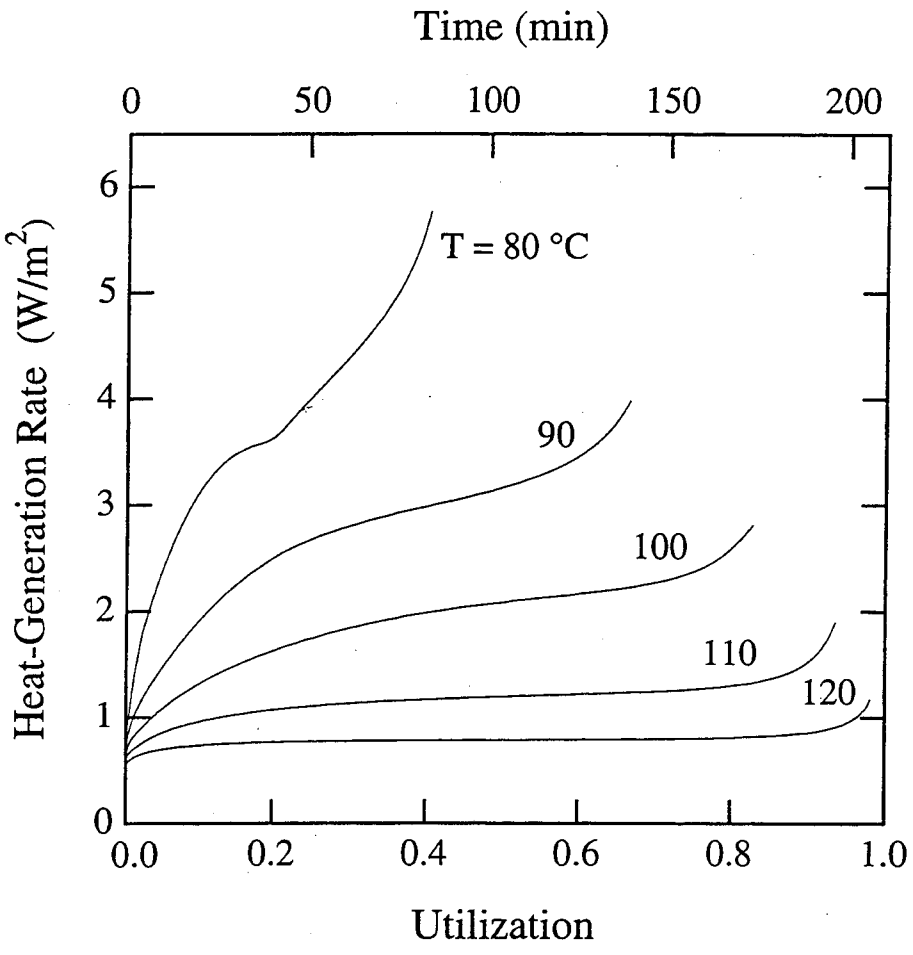

Figure 3-3. Heat generation as a function of utilization and time for the cell at $I = 1.1$ mA/cm^2. Temperature of operation is given in the figure.

where t is the discharge time, I is the discharge current density, and V is the cell potential. For galvanostatic discharge, the current can be taken out of the integral. The theoretical specific energy of the present system is 182.7 W·h/kg, based on the mass of all cell components shown in Figure 2-1. Figure 3-4 gives both the specific energy and average specific power as functions of isothermal cell temperature. In all of the simulations, the cell is discharged at I = 1.1 mA/cm^2, corresponding to a nominal three-hour discharge rate, to the 1.7 V cutoff potential. The figure shows that the average power is only weakly dependent on temperature, increasing by about 5% over the temperature range of 80 to 120°C. The specific energy, on the other hand, is strongly dependent on the temperature, increasing from 68.8 to 174.2 W·h/kg. At lower temperatures, from 80 to 100°C, the specific energy increases steadily as the cell attains higher utilization with increasing temperature. Then, at temperatures from 100 to 120°C, the energy density continues to rise, but at a slower pace, as the cell is reaching nearly 100% utilization. Above 120°C, the specific energy begins to level off to its theoretical value. This figure shows that the TiS_2 system can meet the USABC midterm goal for specific energy, 80 W·h/kg, but not its long-term goal of 200 W·h/kg.[1]

3.3 Adiabatic Discharge Behavior

During an adiabatic discharge, all of the thermal energy produced in the battery is used to heat the battery. The system is assumed to be of uniform temperature, and this temperature is allowed to vary with time. Adiabatic modeling is important because we are interested in the consequences of no heat removal, for example, if the temperature-control system fails. We can also analyze the thermal consequences of a short circuit in the battery.

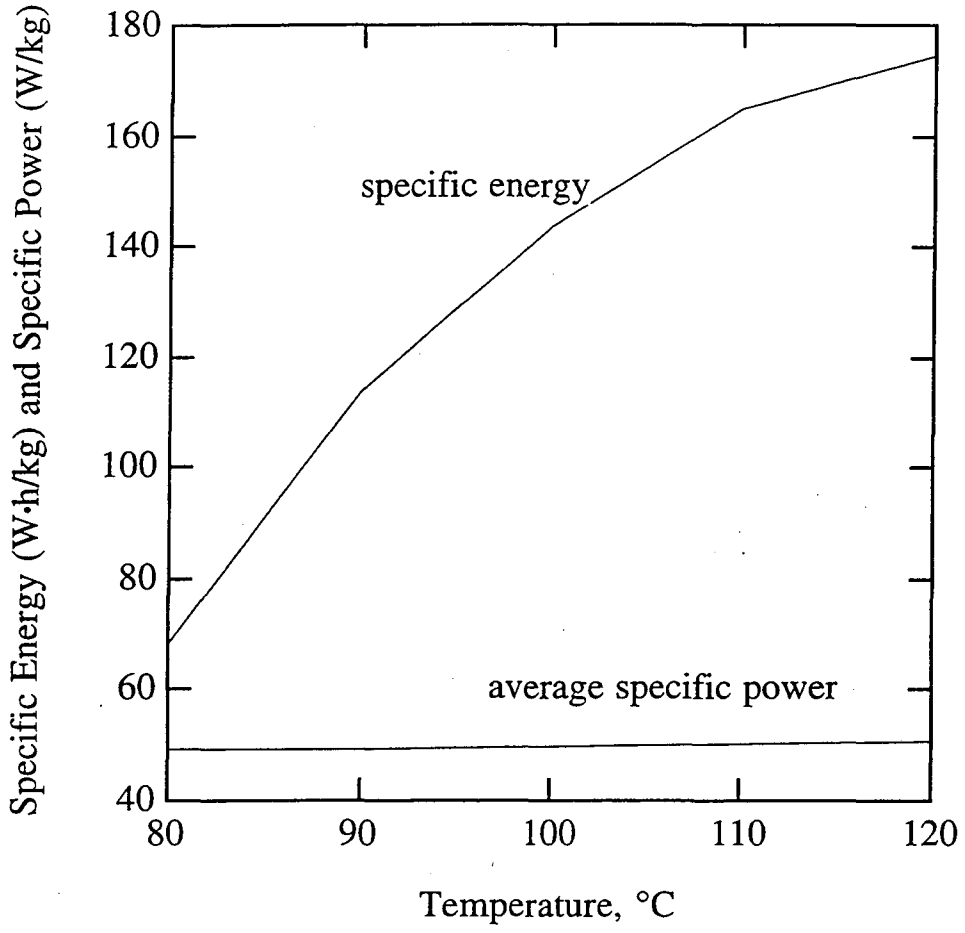

Figure 3-4. Specific energy and average specific power of the cell as functions of temperature for isothermal, galvanostatic discharge at I = 1.1 mA/cm^2.

Cell-Potential Behavior. Figure 3-5 gives the cell potential as a function of utilization of positive electrode material for an adiabatic galvanostatic discharge with the current density as a parameter. The dashed line is the open-circuit potential of the cell. Initially the discharge curve is similar to the isothermal case, but as heat is generated and the temperature rises, the overpotentials due to ohmic drop, and to a lesser extent, concentration and surface overpotentials, decrease. This improves the performance of the system so that even at I = 2.0 mA/cm^2, 100% utilization can be attained. At higher discharge rates (greater than 2.0 mA/cm^2), a local minimum in the cell potential can occur. In the most dramatic case, with I = 6.4 mA/cm^2, the cell potential almost reaches the cutoff potential before increasing again and consuming the remainder of the active material. The 6.4-mA/cm^2 curve corresponds to the highest discharge rate that will not reach the cutoff potential and shut down at the local potential minimum, and therefore will give the worst-case scenario for temperature rise under normal operation. At this discharge rate, heat produced during the discharge raises the conductivity and saves the cell from being cut off. This unusual effect is due to the large temperature increase in the cell under adiabatic conditions coupled with the strong temperature dependence of the ionic conductivity. The markers on the curve for I = 6.4 mA/cm^2 represent points for which salt concentration profiles are shown on Figures 3-6 and 3-7.

Concentration Profiles. In order to understand the effect of adiabatic temperature rise on the concentration profiles in the solution phase, we examine several concentration profiles across the cell taken at various times over the course of a discharge at I = 6.4 mA/cm^2. Figure 3-6 gives the profiles early in the discharge, represented as "x"s on Figure 3-5. Initially, the reaction proceeds primarily at the separator/electrode boundary, and the solution is most depleted of salt in this region. After one minute, the reaction starts to penetrate into the interior of the porous electrode,

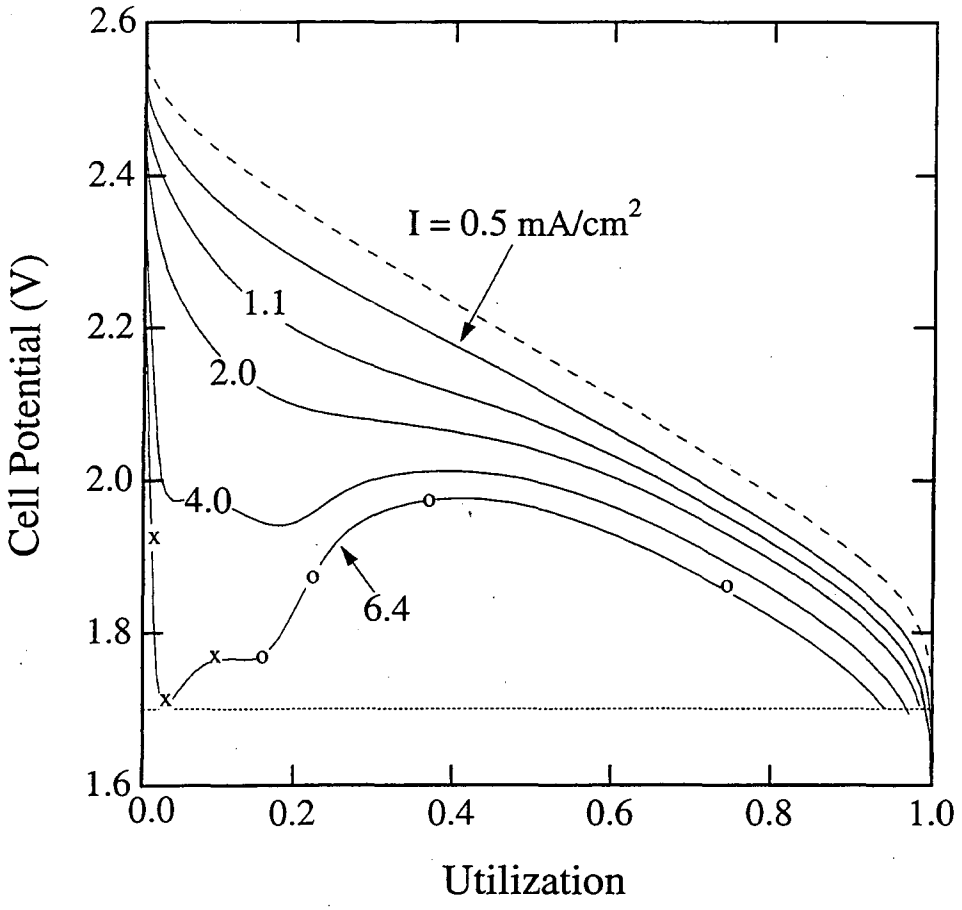

Figure 3-5. Cell potential *versus* utilization of active material at various current densities for adiabatic, galvanostatic discharges. The dashed line is the open-circuit potential of the cell. The markers on the I = 6.4 mA/cm^2 discharge refer to times for which concentration profiles are given in Figures 3-6 and 3-7. The initial cell temperature was 90 °C.

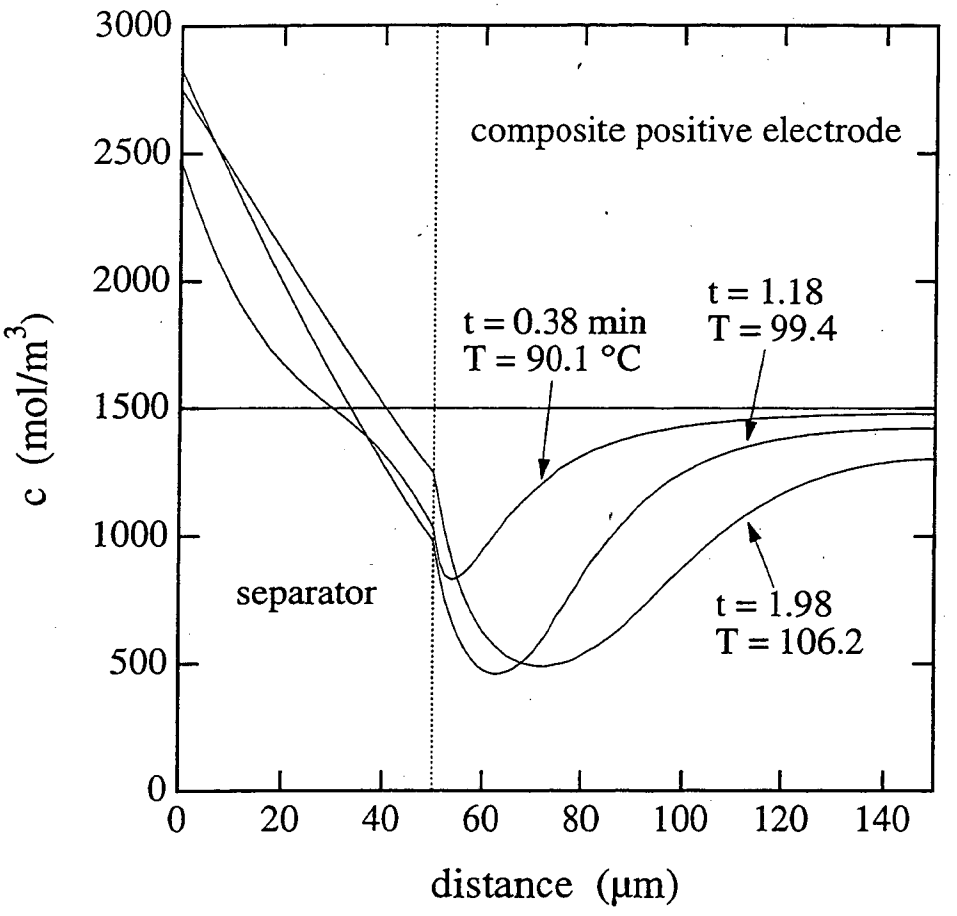

Figure 3-6. The solution-phase concentration profiles across the full cell during an adiabatic, galvanostatic discharge at I = 6.4 mA/cm^2 starting at T = 90 °C. The time since the beginning of discharge is given in minutes, and the cell temperature is given in °C.

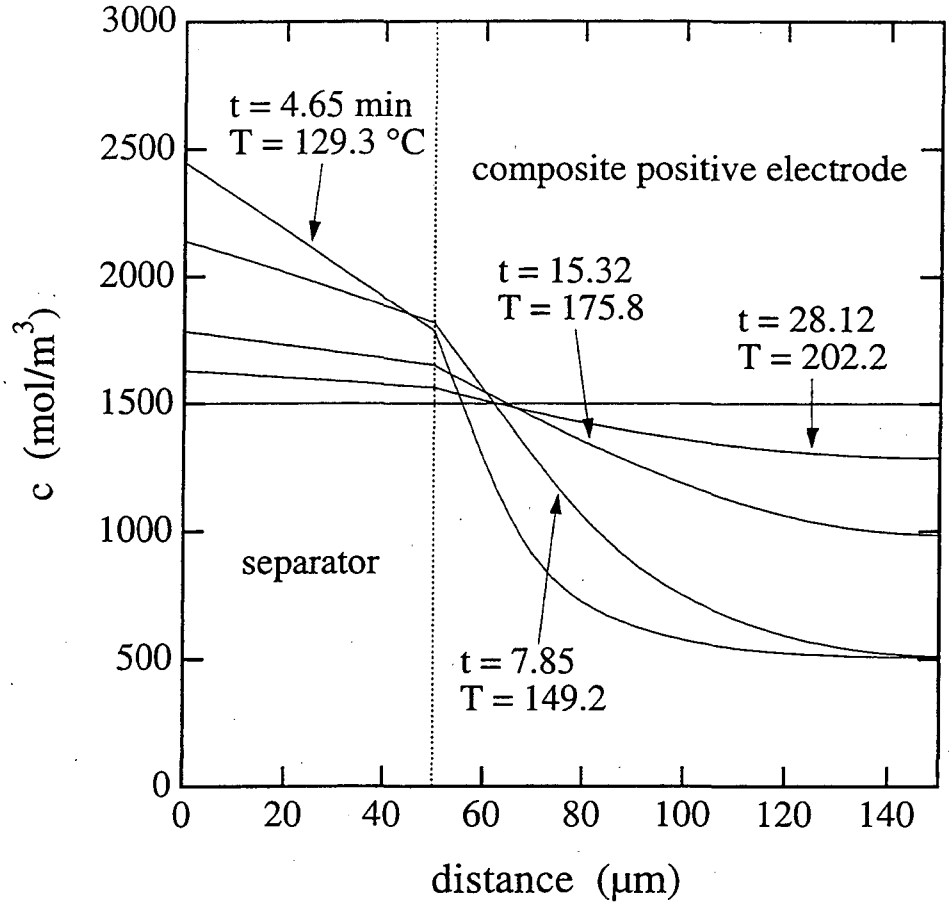

Figure 3-7. The solution-phase concentration profiles across the full cell during an adiabatic discharge at I = 6.4 mA/cm^2 starting at T = 90 °C. The time since the beginning of discharge is given in minutes, and the cell temperature is given in °C.

and the region of salt depletion increases in width. At this point, the cell potential has its minimum value; presumably this minimum reflects the strong temperature dependence of the ionic conductivity. Between one and two minutes into the discharge, heat generation is causing the cell's temperature to increase, which brings about an increase in the salt diffusion coefficient. This is reflected in the slope of the concentration profile in the separator region, which scales inversely with the diffusion coefficient.

In Figure 3-7 the concentration profiles later in the discharge, represented as "o"s on Figure 3-5, are given. Here we see that the concentration gradient in the cell slowly relaxes over the course of the discharge due to the increasing temperature. Near the end of discharge, at t = 28 minutes, there is only a minor concentration gradient in the cell.

Temperature Rise. The temperature in the cell *versus* utilization of active material is given in Figure 3-8 for the conditions shown in Figure 3-5. Higher temperatures are reached for higher current densities as would be expected. For a current density of 1.1 mA/cm^2, the temperature of the battery rises by 40°C. It is likely that there would be a maximum allowed temperature in the cell in order to avoid, for example, melting of the solid lithium or thermal decomposition of the polymer. This figure is then useful in predicting the maximum discharge rate to stay below this maximum temperature under any circumstances. For example, the melting point of lithium is 186°C, and from Figure 3-8 we find that this temperature is exceeded at current densities of greater than 4.0 mA/cm^2.

Heat Generation. The heat-generation rate of the cell as a function of utilization is given in Figures 3-9 and 3-10 for the same conditions shown in Figure 3-5. The heat

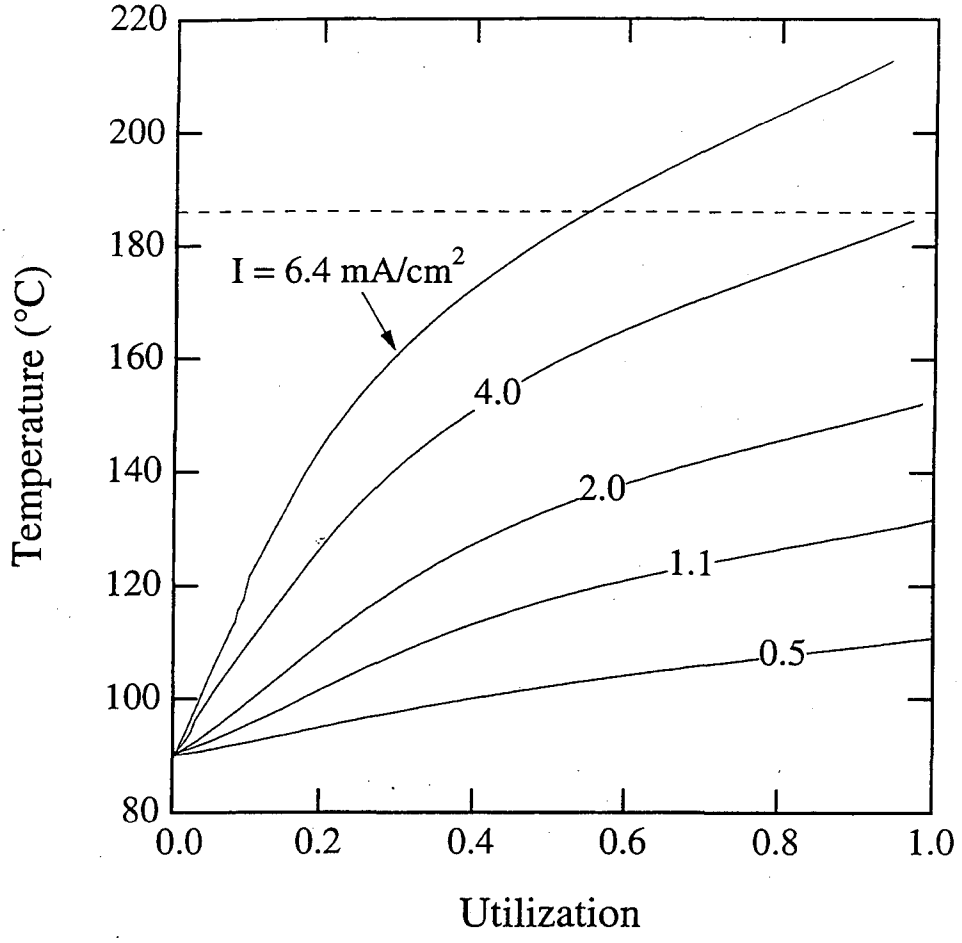

Figure 3-8. Temperature in the cell as a function of active-material utilization during several adiabatic, galvanostatic discharges. The discharge rates are given on the figure. The dashed line is the melting point of lithium.

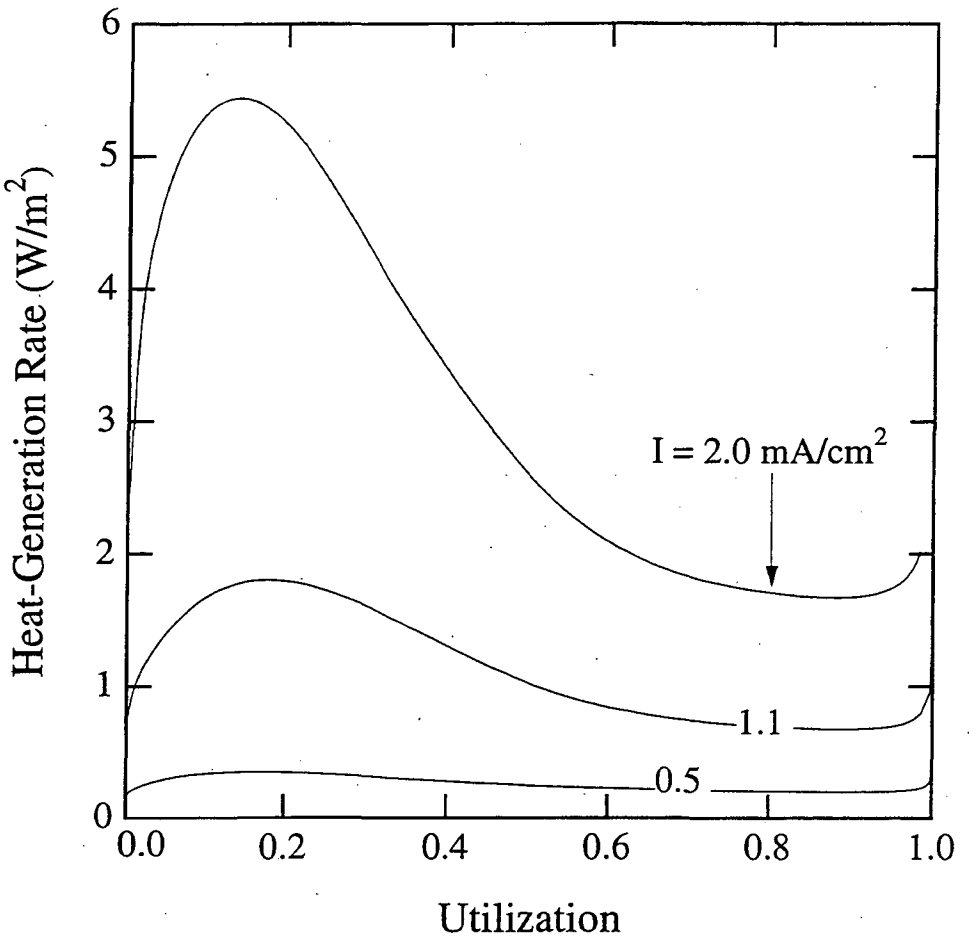

Figure 3-9. Heat-generation rate as a function of active-material utilization for adiabatic discharges at low rates. Current densities are given on the figure. The initial cell temperature was 90 °C.

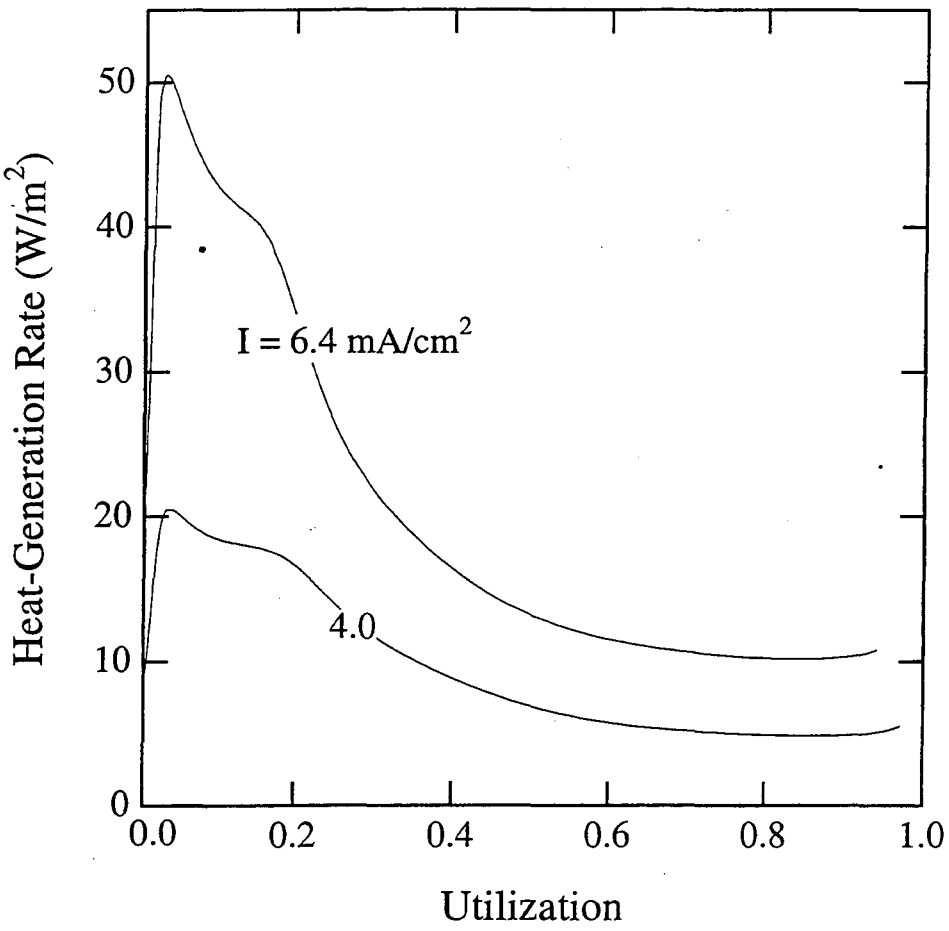

Figure 3-10. Heat-generation rate as a function of active-material utilization for adiabatic discharges at high rates. Current densities are given on the figure. The initial cell temperature was 90 °C.

generated increases early in the discharge and then decreases as overpotentials, primarily ohmic drop, are diminished. This causes the temperature in the cell to rise more quickly near the beginning of discharge, which was seen in Figure 3-8. Near the end of the discharge, the heat-generation rate increases again due to the sudden increase in overpotential as the active material is exhausted. As with the isothermal case, this result can also be seen in the cell-potential behavior, which was shown in Figure 3-5. It is interesting that the heat-generation rate for the 6.4-mA/cm^2 case is ten times greater than the heat-generation rate for the 2.0-mA/cm^2 case at the initial heat-generation peak.

3.4 Heat Transfer

In the one-cell model the temperature of each cell is assumed to be uniform and is allowed to vary with time. Thus far, the one-cell model has been used only to calculate results for adiabatic and isothermal conditions without regard to heat-transfer conditions. The one-cell model can also be used to analyze the behavior of a cell stack under heat-transfer conditions by defining an appropriate per-cell heat-transfer coefficient for each cell in the cell stack, or by applying results to a cell stack with a negligible temperature gradient. We consider one-dimensional heat transport in the direction perpendicular to cell layers. The assumption of one-dimensional heat transport is valid when transport in the direction parallel to cell layers can be neglected, such as when the cell stacks are very thin, or when the ends of the stack perpendicular to cell layers are insulated. Equation 2-28 is used to calculate the temperature as a function of time for a cell stack with appropriate values of h* for each cell.

In order to examine the results presented for the heat-transfer simulations, we need to know the order of magnitude of the convective heat-transfer coefficient, h, for heat-transfer conditions that might be applicable to the system. For electric-vehicle

applications, the most likely heat-transport medium will be air. Table 3-1 gives values of the convective heat-transfer coefficient for free- and forced-air convection.

Table 3-1. Order of magnitude of convective heat-transfer coefficients.[2]

Fluid	h, $W/m^2 \cdot K$
Air, free convection	6 to 30
Air, forced convection	30 to 300

The Per-Cell Heat-Transfer Coefficient. In order to use the one-cell model for cell-stack calculations, the per-cell heat-transfer coefficient must be related to the convective heat-transfer coefficient. The per-cell heat-transfer coefficient will be defined for each cell in the cell stack by comparing the one-cell energy balance at steady-state to the energy balance for a cell stack at steady-state. If we assume steady-state and combine Equations 2-28 and 2-29, we get the energy balance for a single cell:

$$Q = h^* (T - T_a) \tag{3-3}$$

where Q is the heat-generation rate per unit separator area, and h* is based on separator area.

For a cell stack at steady-state, the energy balance is:

$$k_{cell} \frac{d^2T}{dx^2} = \frac{Q}{l_{cell}}. \tag{3-4}$$

For a stack of thickness L, we define the center of the cell stack to be x = 0 and the outer face of the cell stack to be $x = \frac{L}{2}$. The boundary conditions are then:

$$-k_{cell}\frac{dT}{dx} = h(T-T_a) \quad \text{at} \quad x = \frac{L}{2}, \tag{3-5}$$

using the convective heat-transfer coefficient at the stack edges, and

$$\frac{dT}{dx} = 0 \quad \text{at} \quad x = 0, \tag{3-6}$$

because heat is not transported through the center of the stack. The solution of Equation 3-5 using the boundary conditions given above is:

$$T-T_a = \frac{Q}{2k_{cell}l_{cell}}\left[\left(\frac{L}{2}\right)^2 - x^2\right] + \frac{QL}{2hl_{cell}}. \tag{3-7}$$

To relate the per-cell heat-transfer coefficient to the convective heat-transfer coefficient, we combine Equations 3-3 and 3-7 and rearrange:

$$\frac{1}{h^*} = \frac{1}{2\,k_{cell}l_{cell}}\left[\left(\frac{L}{2}\right)^2 - x^2\right] + \frac{L}{2\,h\,l_{cell}} \tag{3-8}$$

where $0 \le x \le \frac{L}{2}$. At x=0, the center of the cell stack, Equation 3-8 becomes:

$$\frac{1}{h^*} = \frac{N_{cell}^2\,l_{cell}}{8\,k_{cell}} + \frac{N_{cell}}{2\,h}, \tag{3-9}$$

using the fact that $N_{cell} = \frac{L}{l_{cell}}$. At the outer face of the cell stack, $x = \frac{L}{2}$, h^* is given by:

$$h^* = \frac{2\,h}{N_{cell}}, \tag{3-10}$$

and for a cell half-way between the center and the outer face of the cell stack, $x = \frac{L}{4}$, h^* is given by:

$$\frac{1}{h^*} = \frac{3\, N_{cell}^2\, l_{cell}}{32\, k_{cell}} + \frac{N_{cell}}{2\, h}. \tag{3-11}$$

Now that the per-cell heat-transfer coefficient has been related to the convective heat-transfer coefficient in this way, the temperature and discharge behavior for each cell in the cell stack can be calculated from known values of the convective heat-transfer coefficient and from cell properties.

Negligible Temperature Gradients. The one-cell model can also be used to estimate the temperature and discharge behavior of the battery when the cell-stack temperature gradient can be neglected. Two criteria are given for assessing whether the temperature gradient is negligible. The first criterion is an estimation of the temperature variation in the cell stack based on a steady-state value of the heat-generation rate, and the second is the Biot-number analysis. The required accuracy of the calculation determines the acceptable limits for which the negligible-temperature-gradient approach may be applied.

The first criterion for determining whether the stack temperature gradient is negligible is an estimation of the temperature gradient at steady-state. If we solve Equation 3-8 for the temperature at the center and at the outer face of the cell stack, we can give an equation for the expected temperature variation in the cell stack at steady-state:

$$T_{x=0} - T_{x=\frac{L}{2}} = \frac{Q l_{cell} N_{cell}^2}{8 k_{cell}}. \tag{3-12}$$

Using an approximated value for the heat-generation rate, the temperature variation in the cell stack at steady-state can be estimated. If the temperature variation is negligible, according to the required accuracy of the calculations, the per-cell heat-transfer coefficient can be calculated from the convective heat-transfer coefficient by Equation 3-10.

The Biot-number analysis is another criterion that can be used to assess whether the temperature profile in the cell stack can be neglected. The Biot number, symbolized Bi, characterizes the importance of the internal thermal resistance of the solid as compared to the external thermal resistance of the fluid. For the slab geometry considered above,

$$Bi = \frac{hL}{k_{cell}}. \tag{3-13}$$

As the Biot number approaches zero, the temperature gradient in the solid can be neglected. A rule of thumb for using the Biot number is that if $Bi < 0.1$ the assumption of uniform temperature will introduce an error in the calculations of less than five percent.[2] Using the Biot-number analysis, the maximum number of cells allowed in a cell stack that can be assumed to be of uniform temperature, and therefore can be assumed to follow the calculations of the one-cell model, is given by

$$N_{cell} < \frac{0.1\ k}{h\ l_{cell}}. \tag{3-14}$$

For the TiS_2 cell presented here, the number of cells and the cell-stack thickness allowed by the Biot-number analysis for a negligible temperature gradient are given in Equations 3-15 and 3-16:

$$N_{cell} < \frac{261}{h} \tag{3-15}$$

$$L < \frac{0.064}{h} \tag{3-16}$$

where h is in $W/m^2 \cdot K$ and L is in m. The per-cell heat-transfer coefficient is calculated from the convective heat-transfer coefficient using Equation 3-10.

Results. Figure 3-11 gives the potential of the TiS_2 cell as a function of time and utilization for galvanostatic discharge at 1.1 mA/cm^2 for several values of the per-cell heat-transfer coefficient, h*, as well as for adiabatic conditions. Figure 3-12 and 3-13 give the temperature and per-cell heat-generation rate of the cell stack, respectively, for the same conditions.

Figures 3-11 and 3-12 show that as the per-cell heat-transfer coefficient increases, the cell potential and temperature decrease. This behavior is expected because as the heat-transfer coefficient increases, more heat is transferred out of the cell stack to the surroundings causing the cell temperature to decrease. This lower temperature causes the cell potential to decrease, as was seen in isothermal discharges. The per-cell heat-generation rate increases, as with the isothermal runs, as the temperature of the cell decreases. This behavior is due to the increase in overpotentials in the cell.

For example, a 300-cell stack that is exposed to a convective heat-transfer coefficient of 24 $W/m^2 \cdot K$ on its two outer faces will have a value of $h^* \doteq 0.095$ at its center (x=0) and $h^* = 0.16$ at its outer faces (x=L/2). The $h^* = 0.10$ and $h^* = 0.15$ curves on Figures 3-11 through 3-13 will approximate the behavior of these cells in the cell stack. From these figures, it is seen that the center cell of the stack will have a higher cell potential and will be much hotter than the cells on the edge of the cell stack.

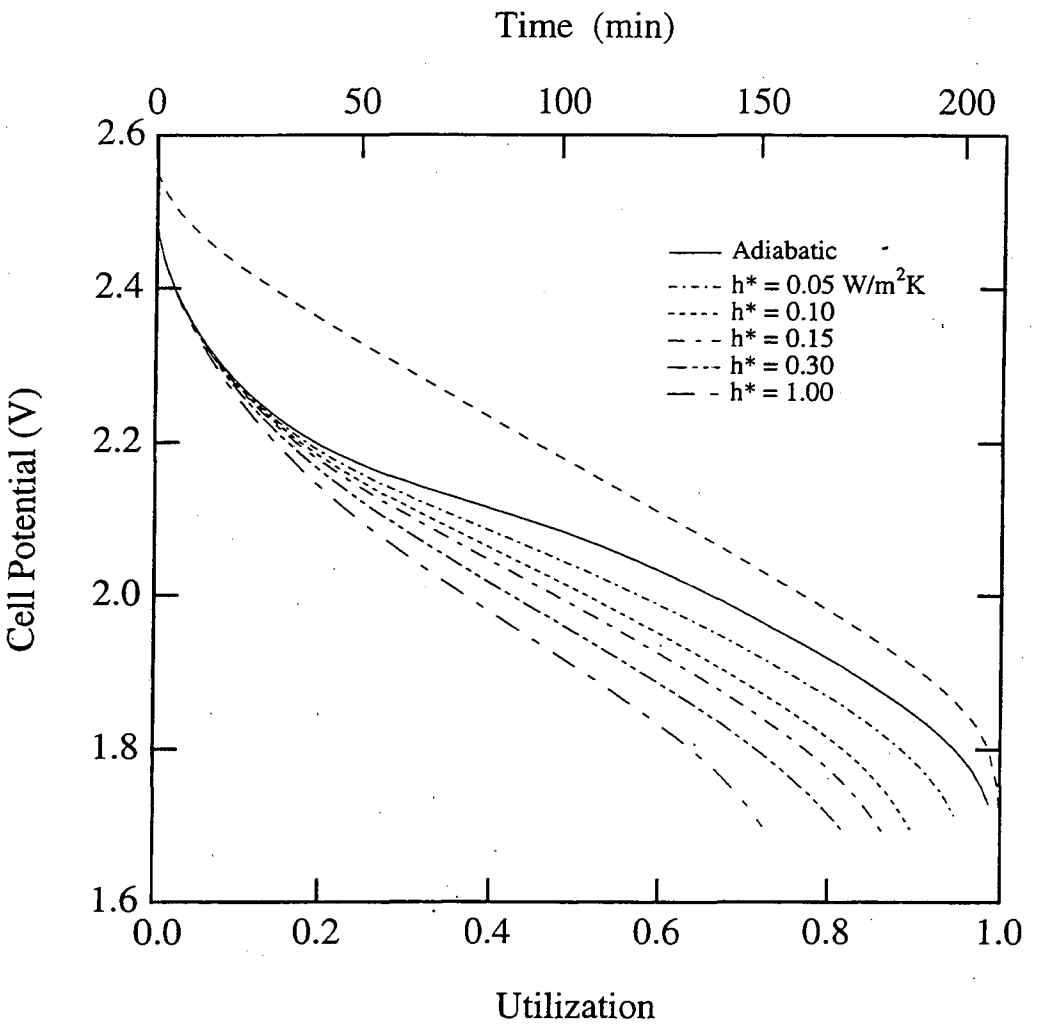

Figure 3-11. Cell potential as a function of time and utilization for galvanostatic discharges at I = 1.1 mA/cm^2. Values of the per-cell heat-transfer coefficient, h*, are given on the figure. The dashed line is the open-circuit potential of the cell.

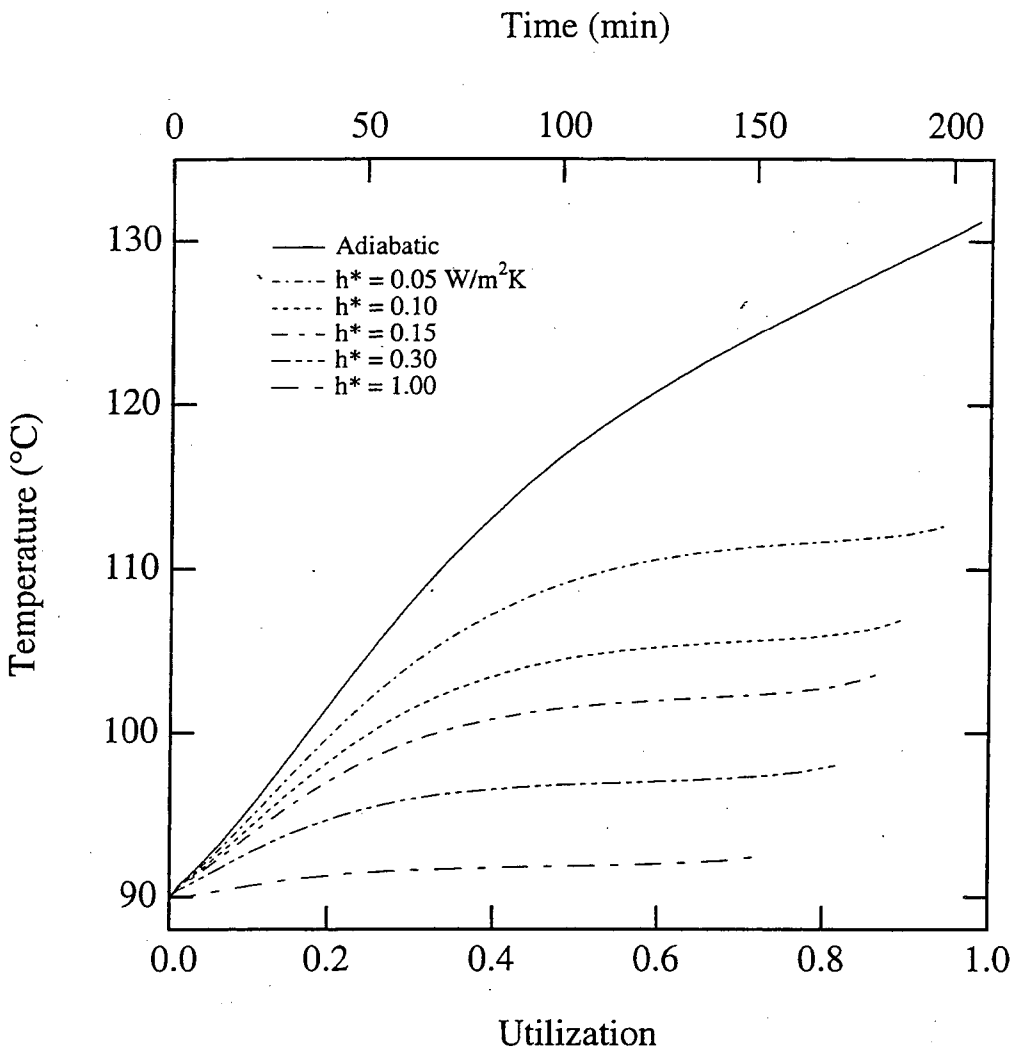

Figure 3-12. Temperature as a function of utilization and time for galvanostatic discharges at I = 1.1 mA/cm^2. Values of the per-cell heat-transfer coefficient, h*, are given on the figure.

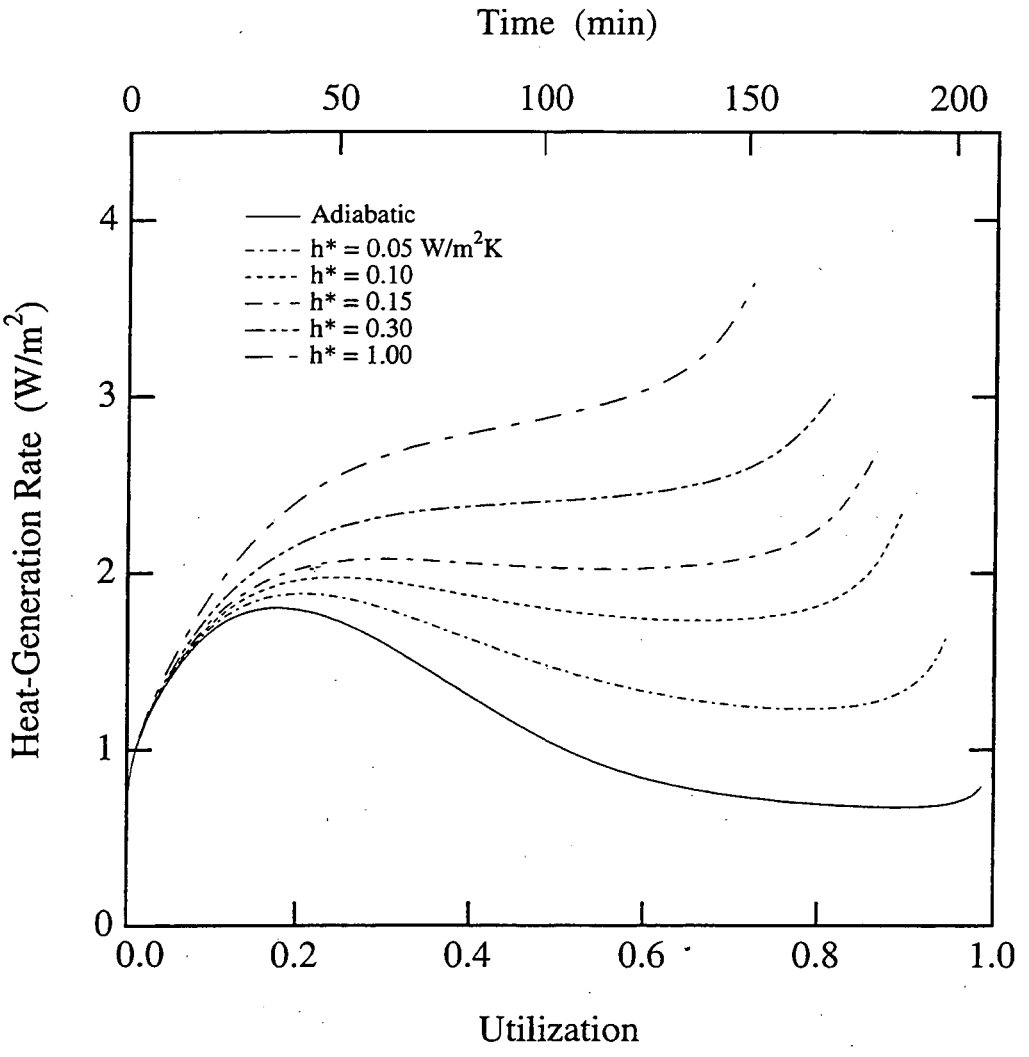

Figure 3-13. Heat-generation rate as a function of utilization and time for galvanostatic discharges at I = 1.1 mA/cm^2. Values of the per-cell heat-transfer coefficient are given on the figure.

The difference in temperature between the center and the edge of the stack at the end of discharge will be approximately 3 °C.

3.5 Conclusion

In this chapter, results of the one-cell model have been presented. Results for adiabatic and isothermal runs of the $Li|PEO_{18}$-$LiCF_3SO_3|TiS_2$ system were presented and explained. Finally, the per-cell heat-transfer coefficient was defined for each cell in the cell stack, and simulations for cell stacks under heat-transfer conditions were presented.

The simulations showed that the cell potential, along with active-material utilization at the cutoff potential increases with increasing temperature. Following from the cell-potential behavior, it was seen that the heat-generation rate of the cell decreases with increasing temperature. It was also shown that cells operating isothermally above 90 °C are ohmically limited. In adiabatic discharge, it was shown that a local minimum in cell potential is observed for currents above 2.0 mA/cm^2, and that above 4.0 mA/cm^2 the melting temperature of lithium is reached.

The per-cell heat-transfer coefficient was defined for each cell in the cell stack. Simulations were presented for discharge behavior for various values of the per-cell heat-transfer coefficient. The simulations presented for the per-cell heat-transfer coefficient can also be used if the temperature gradient in the cell stack is negligible according to the accuracy requirements of the calculations. An estimation for the steady-state temperature variation in the stack along with the Biot-number analysis were given to assess whether the temperature variation could be neglected.

List of Symbols

A_s	separator area, cm^2
A_{ht}	heat-transfer area, cm^2
E	specific energy W·h/kg
h	convective heat-transfer coefficient, $W/m^2 \cdot K$
h*	per-cell heat-transfer coefficient, $W/m^2 \cdot K$
I	current density, mA/cm^2
k_{cell}	thermal conductivity of cell, W/m·K
l_{cell}	thickness of the cell, m
L	thickness of the cell stack, m
M	mass, kg
N_{cell}	number of cells in cell stack
P	power density, W/kg
Q	heat-generation rate, W/m^2
t	time, s
T	temperature, K
T_a	ambient temperature, K
V	cell potential, V
x	distance from center of cell stack

References

1. *Draft Technical Document for the Low-Emission Vehicle and Zero-Emission Vehicle Workshop on March 15, 1994; Zero Emission Vehicle Update,* California Air Resources Board.

2. Frank Kreith and Mark S. Bohn, *Principles of Heat Transfer,* Harper and Row, New York, New York (1986).

Chapter 4. Cell-Stack Model Development

4.1 Introduction

Now that we have examined how cell performance varies with operating temperature, we are interested in modeling the temperature profile of a cell stack. Due to conductive-heat-transfer limitations, the temperature in the center of a cell stack will be higher than the temperature at its edges. Calculating temperature profiles in cell stacks is important because cell behavior depends on temperature as shown in Chapter 3, and we would like to predict how variation in individual cell performance due to temperature variation will affect the performance of the battery.

For example, we have seen that cells operating at a lower temperature have a lower cell potential, and reach their cut-off potential sooner, and at a lower value of active-material utilization. As shown in Chapter 3, cells on the outside of the stack operate at a lower temperature than cells on the inside of the stack; they also operate at a lower cell potential, and reach their cut-off potential sooner than the internal cells.

We are interested also in the design of the temperature-control system. The temperature-control system should keep the battery operating within some nominal temperature range, as well as try to minimize temperature gradients. Newman and Tiedemann[1] have given a three-dimensional model of temperature rise in a battery module with a constant heat-generation rate, but temperature profiles are not calculated. Chen and Evans[2,3] calculate temperature profiles using a uniform heat-generation rate calculated from experimental discharge curves, but fail to assess the effect of the temperature change on the performance of the battery or of the individual cells.

This chapter presents a one-dimensional model for temperature profiles in a cell stack. In the model, heat is assumed to be transferred only in the direction perpendicular to cell layers. This assumption is valid only for very thin cell stacks or

for cell stacks that are insulated on the edges perpendicular to cell layers. The model will estimate the variable energy-generation rate of each cell from data calculated in isothermal runs of the one-cell model, and this estimation will be assessed.

4.2 The Cell Stack

A cell stack is a configuration of several individual cells. For a bipolar configuration, the cells are connected electrically in series. The cell-stack model assumes that the temperature behavior of the full cell stack is symmetric, and therefore only half of the cell stack is modeled. Figure 4-1 is a schematic showing the temperature profile in one-half of a cell stack. The cell stack consists of N_{cell} cells, where $N_{cell} = 2n$. In the cell-stack model, as in the one-cell model, it is assumed that each cell is at a uniform temperature which is allowed to vary with time. Air at the outer face of the cell stack is represented by a heat-transfer coefficient, h, and the temperature of this external air is given by T_a. For the TiS_2 system, each cell is configured as shown in Figure 2-1.

4.3 Energy Balance

In order to model the heat-transfer behavior of a cell stack, we first write an energy balance around each of the cells in the stack. The energy balance is of the form:

$$\text{accumulation} = \text{input} - \text{output} + \text{generation}. \tag{4-1}$$

All of the terms are given on a per-unit-area basis, where the basis area is the separator area of a single cell. The accumulation term accounts for temperature rise in the cell;

$$\text{accumulation} = \frac{M\hat{C}_p}{A_s}\frac{dT_i}{dt}, \tag{4-2}$$

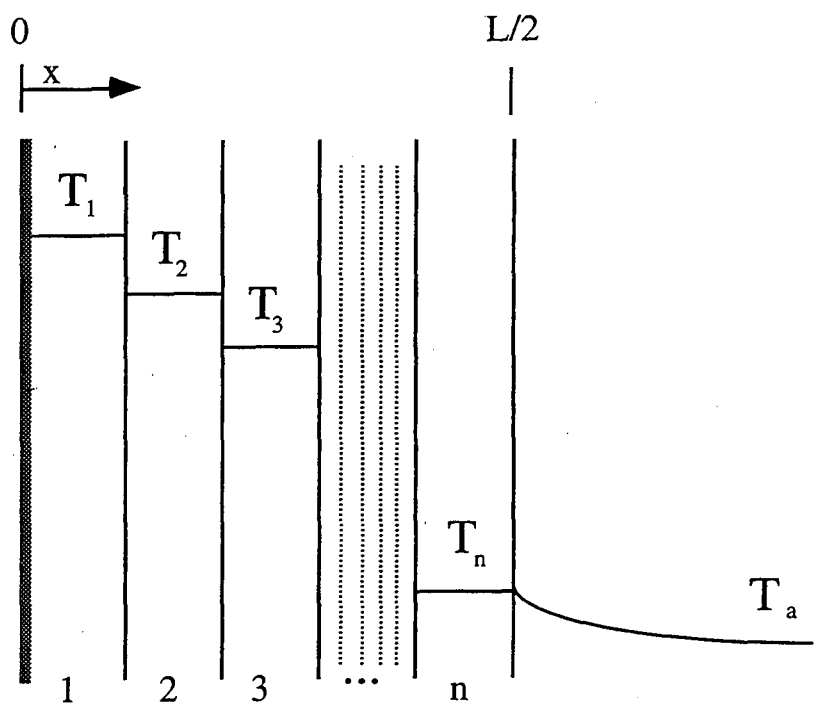

Figure 4-1. Schematic of one-half of a cell stack used for the cell-stack energy balance. Constant temperature is assumed for individual cells.

where M is the mass of one cell. In the simulations it is assumed that $\frac{dT_i}{dt} = \frac{\Delta T_i}{\Delta t}$. This assumption is valid for sufficiently small time steps.

The generation term accounts for thermal energy generated in the cell due to chemical reactions, heat-capacity changes, phase changes, mixing, and electrical work as explained in Section 2.4. The energy generation in cell i, Q_i, may be calculated from Equation 2-29 for each time step. Since using Equation 2-29 to calculate thermal energy generation would require running the one-cell program for each cell, requiring a large amount of computer time, the heat-generation-rate calculation will be greatly simplified as explained in Section 4.4.

The input term follows Fourier's law;

$$\text{input} = - k_{cell} \frac{dT}{dx}\Big|_x. \tag{4-3}$$

Since we have assumed that each cell is at a uniform temperature that varies with time, $\frac{dT}{dx}\Big|_x$ is set to equal $\frac{T_i - T_{i-1}}{l_{cell}}$. The output term for cells 1 through n-1 is also given by Fourier's law,

$$\text{output} = - k_{cell} \frac{dT}{dx}\Big|_{x+\Delta x}, \tag{4-4}$$

where, similarly to the input term, $\frac{dT}{dx}\Big|_{x+\Delta x}$ is given by $\frac{T_{i+1} - T_i}{l_{cell}}$. Cell n is assumed to be in thermal contact with a convective-heat-transfer medium characterized by a convective-heat-transfer coefficient, h. The output term for cell n is given by

$$\text{output} = h\,(T_n - T_a). \tag{4-5}$$

Making the previous substitutions into Equation 4-1 for each cell we get n equations and n unknowns. The unknowns are the cell temperatures, T_i, and the equations are given below.

For cell 1, the center of the cell stack, we assume symmetry, and therefore, $\frac{dT_1}{dx} = 0$. This makes the input term for cell 1 fall out of Equation 4-1, giving

$$\frac{M\hat{C}_p}{A_s \Delta t}\left(T_1^{new} - T_1^{old}\right) = \frac{k_{cell}}{l_{cell}}\left(T_1^{new} - T_2^{new}\right) + Q_1. \tag{4-6}$$

The energy balance gives

$$\frac{M\hat{C}_p}{A_s \Delta t}\left(T_i^{new} - T_i^{old}\right) = \frac{k_{cell}}{l_{cell}}\left(T_{i+1}^{new} - 2T_i^{new} + T_{i-1}^{new}\right) + Q_i, \tag{4-7}$$

for cells 2 through n-1, and

$$\frac{M\hat{C}_p}{A_s \Delta t}\left(T_n^{new} - T_n^{old}\right) = \frac{k_{cell}}{l_{cell}}\left(T_{n-1}^{new} - T_n^{new}\right) - h\left(T_n - T_a\right) + Q_n \tag{4-8}$$

for cell n. Equations 4-6 through 4-8 are cast into an $n \times n$ matrix and solved using the subroutine MATINV.[4] Cell dimensions and physical properties used for simulations based on the TiS_2 system are given in Chapter 2.

4.4 Approximation for the Heat-Generation Rate

The heat-generation rate, Q_i, is a function of time and position in the cell stack as dictated by electrochemical phenomena. Calculating the heat-generation rate of each cell in the stack using the detailed one-cell model given in Chapter 2, along with solving the $n \times n$ matrix given by Equations 4-6 through 4-8, would require unnecessary

computational complexity. We should like to introduce an approximation for the heat-generation rate that will reduce computational difficulty while introducing little error.

Figure 3-3 gives the heat-generation rate of the TiS_2 system as a function of time, active-material utilization, and temperature for isothermal discharge at the nominal three-hour rate. We will make the assumption that the heat-generation rate as a function of time and temperature as shown in Figure 3-3 for isothermal discharge at the three-hour rate gives an accurate approximation to the heat-generation rate as a function of time and position for nonisothermal discharge, and can be used in Equations 4-6 through 4-8. The cell-stack model uses the temperature of cell i at each time step to estimate its heat-generation rate from one-cell model calculations for isothermal discharge. The cell-position dependence of the heat-generation rate is thus given by the cell temperature; $Q_i(t) = Q(t, T_i)$.

Heat-generation rates as functions of time for isothermal discharge were calculated at many temperatures using the one-cell model and were written to an input file. The cell-stack model reads the time, temperature, and heat-generation rate from the input file, and then uses linear interpolation to estimate the heat-generation rate for each cell at a given time and temperature from the times and temperatures read from the input file. These interpolated heat-generation rates are used in the cell-stack model to calculate cell temperatures at the next time step. The accuracy of this assumption will be assessed in the next section. The computer program for the cell-stack model, along with input file examples, is given in Appendix B.

4.5 Validity of the Heat-Generation-Rate Approximation

It is necessary to assess the accuracy of the cell-heat-generation-rate assumption made in Section 4.4. Temperature and heat-generation rates as functions of time for adiabatic discharges at different discharge rates, as well as temperature and heat-

generation rates as functions of time for different heat-transfer coefficients are presented. Results calculated from the one-cell model and the from cell-stack model are compared. Calculations from the cell-stack model are for stacks in which the temperature gradient can be neglected as allowed by the Biot-number analysis.

Figure 4-2 shows the heat-generation rate for adiabatic discharge as calculated by the cell-stack model and the one-cell model as functions of time and discharge rate. The figure shows that the heat-generation rate calculated by the cell-stack model is most accurate for low discharge rates. For the higher discharge rates, the interpolated values of the heat-generation rate underestimate the values calculated by the one-cell model for most of the discharge, and then slightly overestimate the one-cell values at the end of discharge.

The discrepancy between heat-generation rates as calculated by the one-cell model and the cell-stack model can be explained by examining discharge behavior as calculated by the one-cell model for adiabatic and isothermal discharge. Figures 4-3 and 4-4 show the one-cell-model calculations of the cell potential and heat-generation rate, respectively, as functions of time and active-material utilization for adiabatic and isothermal discharges at the three-hour rate. The initial temperature of the adiabatically-discharged cell was 90 °C. Initially, the adiabatic discharge follows the 90 °C isothermal curve, but then as its temperature rises, its cell potential and heat-generation rate turn toward the 100 °C isothermal curves. The adiabatically-discharged cell reaches 100 °C at point a. This point lies between the isothermal calculations for 90 °C and 100 °C discharge, and therefore, at this point, the isothermal data will overestimate the cell potential and underestimate the heat-generation rate of the adiabatic discharge. This difference leads to the discrepancy between heat-generation rates calculated from the one-cell and cell-stack models.

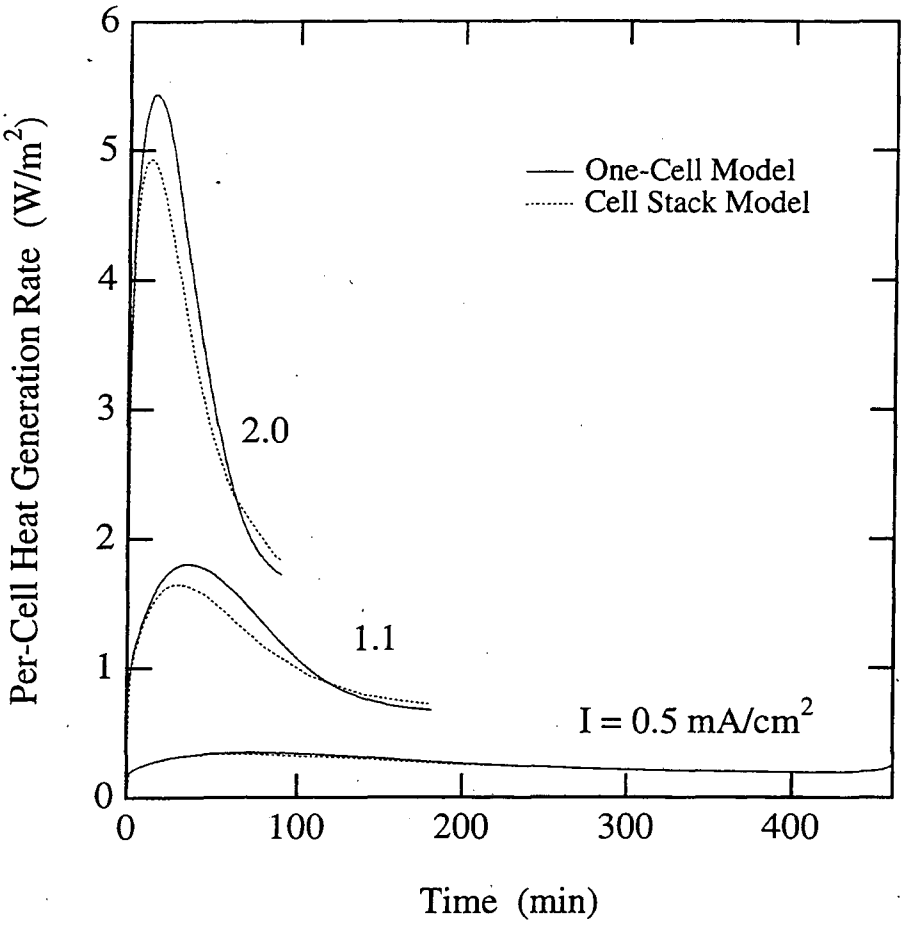

Figure 4-2. Comparison of heat-generation rates as calculated from the one-cell and cell-stack models for adiabatic, galvanostatic discharge at several discharge rates. Discharge rates are given in the figure.

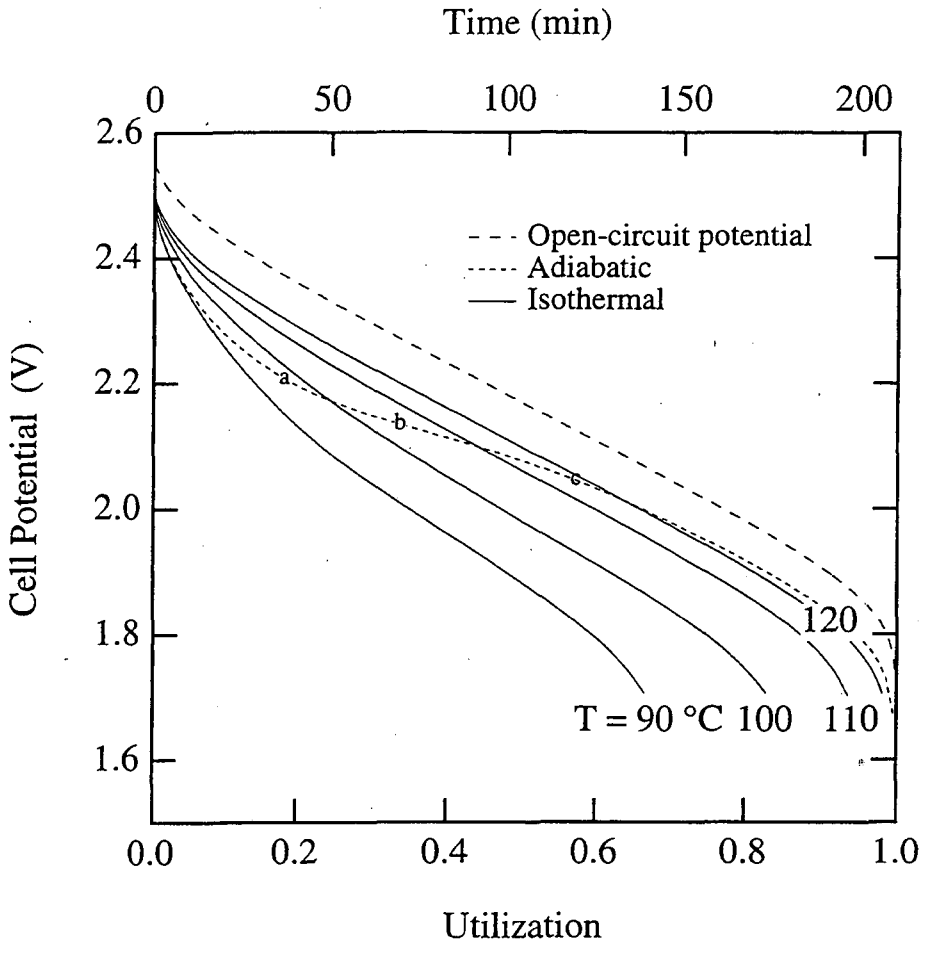

Figure 4-3. Cell potential as a function of utilization and time for adiabatic and isothermal discharge of the cell at I = 1.1 mA/cm^2.

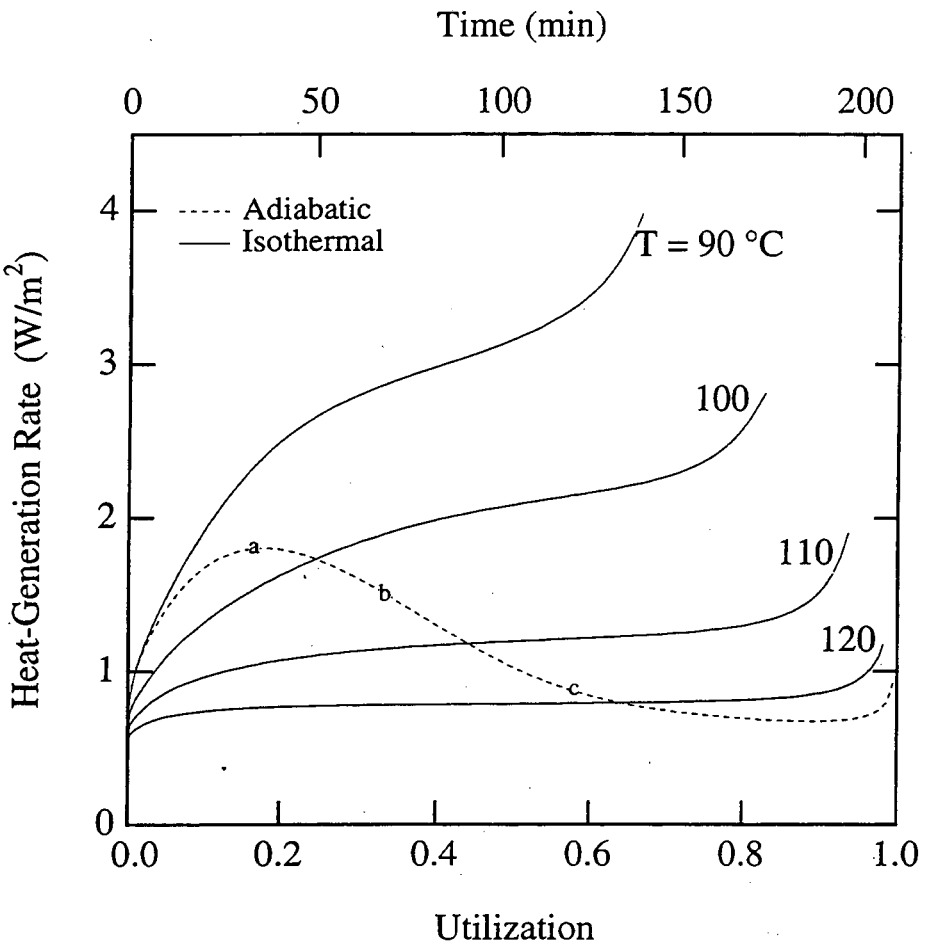

Figure 4-4. Heat-generation rates as functions of utilization and time for adiabatic and isothermal discharges of the cell at I = 1.1 mA/cm^2.

As the adiabatic discharge proceeds, the cell temperature continues to increase. The cell reaches 110 °C at point b, which lies between the isothermal discharge curves for 100 °C and 110 °C, and, as before, the heat-generation rate is underestimated by the isothermal data. When the adiabatic discharge reaches 120 °C, at point c, the 120 °C-isothermal and the adiabatic curves are closer together, and consequently the isothermal data more closely estimates the adiabatic data. As shown in Figure 4-2, later in the discharge, the isothermal data will overestimate the adiabatic heat-generation rate.

In order to understand the difference in the cell potential and heat-generation rates between adiabatic and isothermal discharges at the same times and temperatures, we will compare discharge characteristics of an adiabatic discharge with discharge characteristics of isothermal discharges at corresponding times and temperatures. For adiabatic discharge at the three-hour rate starting at 90 °C, the cell reaches 103.0 °C at 31.5 minutes, corresponding to an active-material utilization of 0.15860, and it reaches 107.5 °C at 61.5 minutes and an active-material utilization of 0.30012. Data at these times and temperatures of the adiabatic run will be compared to data calculated from isothermal runs at corresponding times, temperatures, and values of active-material utilization.

Figure 4-5 shows the solution-phase concentration profiles across the full cell for adiabatic discharge at 1.1 mA/cm^2 at the conditions described above. This figure shows that the concentration profiles are more uniform for the runs in which the temperature was higher initially; the isothermal runs. This uniformity leads to a higher cell potential for the isothermal runs at the same times and temperatures.

Figure 4-6 gives the solid-phase concentration profile across the composite positive electrode for the same runs of the one-cell model shown in Figure 4-5. The distance is measured from the negative electrode, but the separator region is not shown

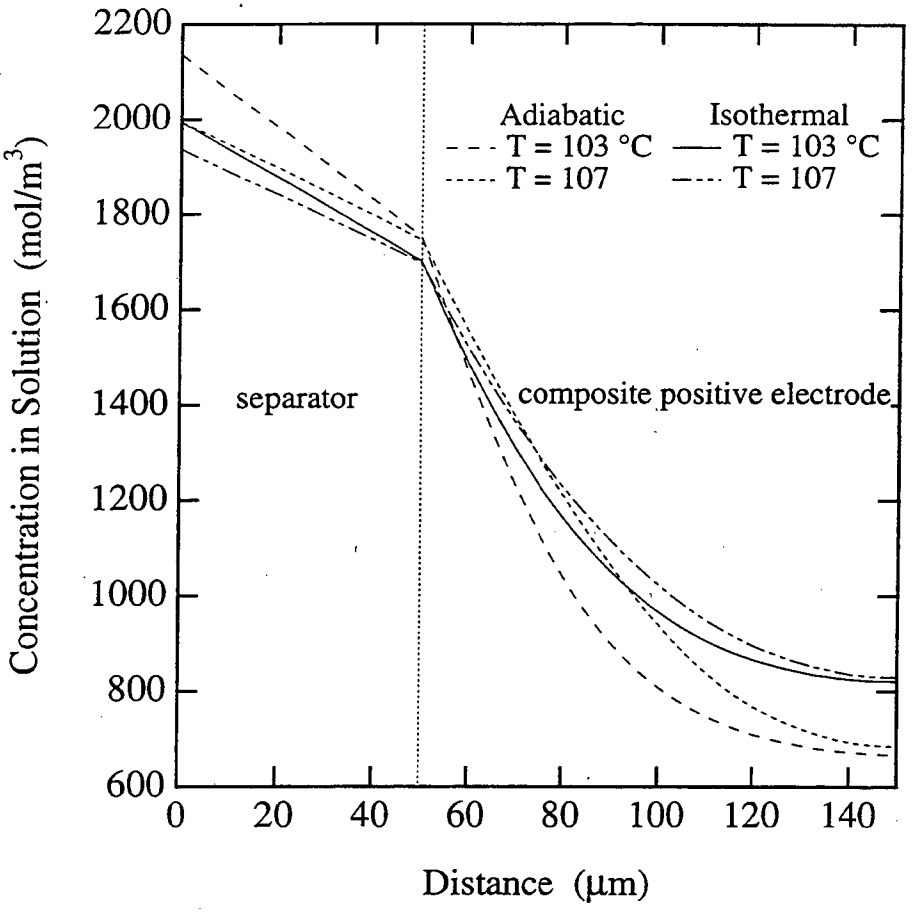

Figure 4-5. Solution-phase concentration profiles for adiabatic discharge at the three-hour rate for two times and temperatures, and for isothermal discharge at corresponding times and temperatures. For the 103.0 °C data, the time is 31.5 minutes, and for the 107.5 °C data the time is 61.5 minutes.

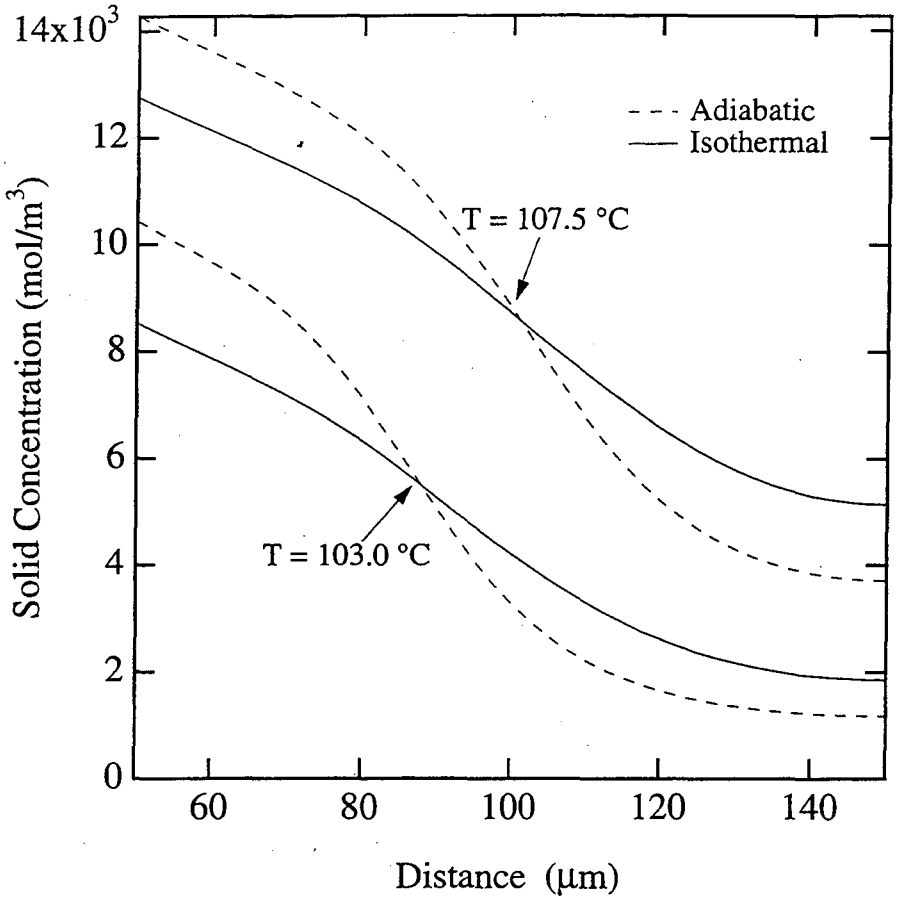

Figure 4-6. Lithium ion concentration in the solid insertion particles for specified times and temperatures of adiabatic and isothermal discharge. The separator region is not included in the figure.

in the figure. This figure shows that uniformity in the solid insertion particles is also greater throughout the discharge for isothermal runs.

Figure 4-7 shows the local pore wall flux of lithium ions across the composite positive electrode for the same conditions shown in Figure 4-5. As in Figure 4-6, the separator region is not shown in the figure. As defined in Chapter 2, the pore wall flux is the flux of lithium ions entering the solution phase from the solid phase, and is thus a measure of the electrode reaction rate. This figure shows that for the isothermal runs, where the temperature is higher at the beginning of the reaction, the reaction distribution is more uniform. The isothermal discharges react more at the back of the electrode earlier in the discharge because the diffusion coefficient and ionic conductivity are initially higher than those of the adiabatic discharge because the temperature is initially higher. Due to transport limitations induced by the lower temperature at the beginning of the adiabatic discharge, the reaction starts in the front of the electrode and proceeds to the back as the ionic conductivity and diffusion coefficient increase.

Now that the errors caused by the heat-generation assumption have been explained, the results for the temperature of the cell stack as a function of time will be presented. Figure 4-8 shows the temperature of the cell stack as a function of time for the same conditions shown in Figure 4-2. As with the heat-generation rates, the temperature is most accurate for lower discharge rates. This behavior is expected because, as discussed previously, the heat-generation rates are estimated from isothermal data, and as a result, the estimation will be more accurate as the temperature stays more nearly constant. For the 0.5 mA/cm^2 discharge, the error at the end of discharge is less than 1° C, and for the 2.0 mA/cm^2 rates, the error at the end of discharge is approximately 5 °C. For each of the discharge rates shown, the difference is approximately ten percent of the overall temperature rise.

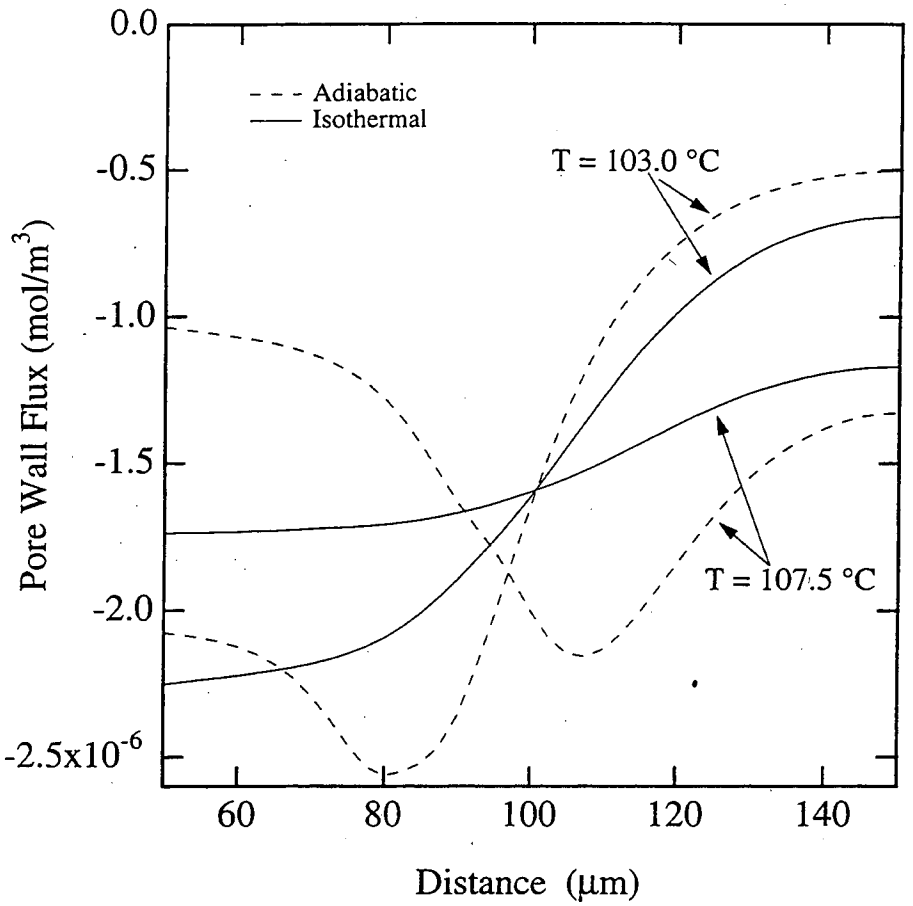

Figure 4-7. Pore wall flux in the composite insertion particles as a function of distance in the full cell for adiabatic and isothermal runs at specified times and temperatures. The separator region is not shown.

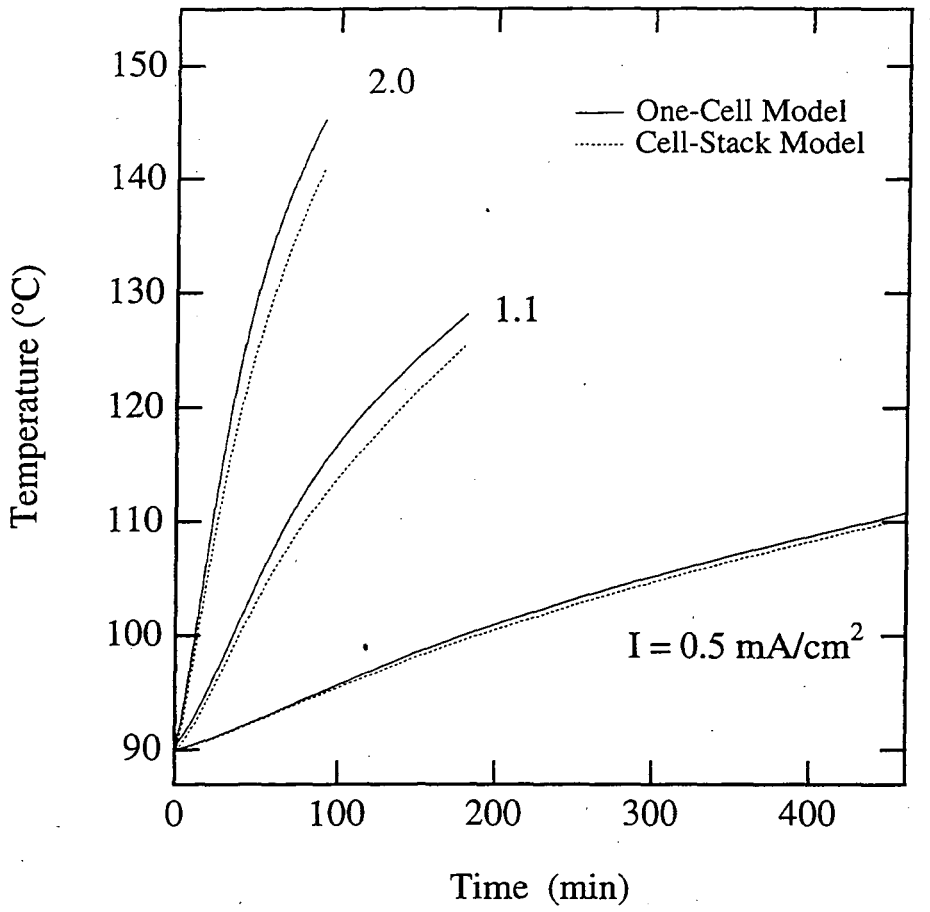

Figure 4-8. Comparison of temperature as a function of time as calculated from the one-cell model and cell-stack model for adiabatic discharge at several discharge rates. Discharge rates are given in the figure.

Figure 4-9 shows the heat-generation rate as a function of time for a single cell as calculated by the one-cell and cell-stack models for discharge at 1.1 mA/cm^2 under various heat-transfer conditions. Figure 4-10 shows the temperature as a function of time for the same simulations. The per-cell heat-transfer coefficient, h*, which is related to the convective heat-transfer coefficient by Equation 3-10, is given as a parameter. The figures show that for one cell, the estimated heat-generation rates and corresponding temperatures are most accurate for higher heat-transfer coefficients where the temperature remains more nearly constant. The variation between model calculations is greatest for the adiabatic case where the difference between the one-cell model and cell stack model is 3 °C at the end of discharge. For a single cell, or for a cell stack with a negligible temperature gradient as predicted by Equations 3-12 or 3-14, the cell-stack model is more accurate for higher values of the per-cell heat-transfer coefficient, h*. Temperature profiles calculated from the cell-stack model, along with a comparison of results calculated by the one-cell and cell-stack models are presented in Chapter 5.

4.6 Conclusion

This chapter presents the cell-stack model, a model which calculates temperature profiles in cell stacks using an approximation for the time- and position-variant heat-generation rate. This model allows us to calculate temperature profiles in cell stacks in order to assess the effect of temperature variations on battery performance.

For a single cell, differences between calculated heat-generation rates and temperatures from the one-cell model and the cell-stack model are due to the heat-generation-rate approximation. Cells discharged for nonisothermal conditions do not show the same behavior as cells discharged isothermally at the same temperature and time, and therefore the heat-generation-rate approximation, along with the cell-stack model calculations, will not be exact.

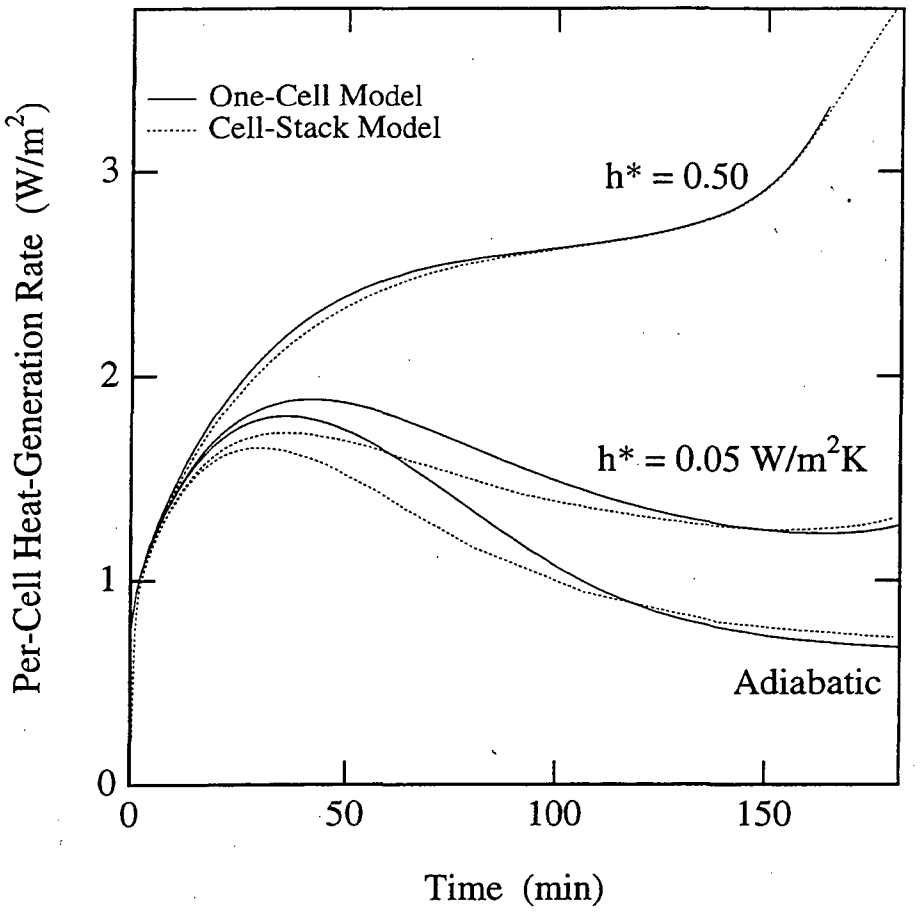

Figure 4-9. Comparison of per-cell heat-generation rates as calculated from the one-cell model and the cell-stack model for galvanostatic discharges of one cell at the three-hour rate. Values of the per-cell heat-transfer coefficient are given on the figure.

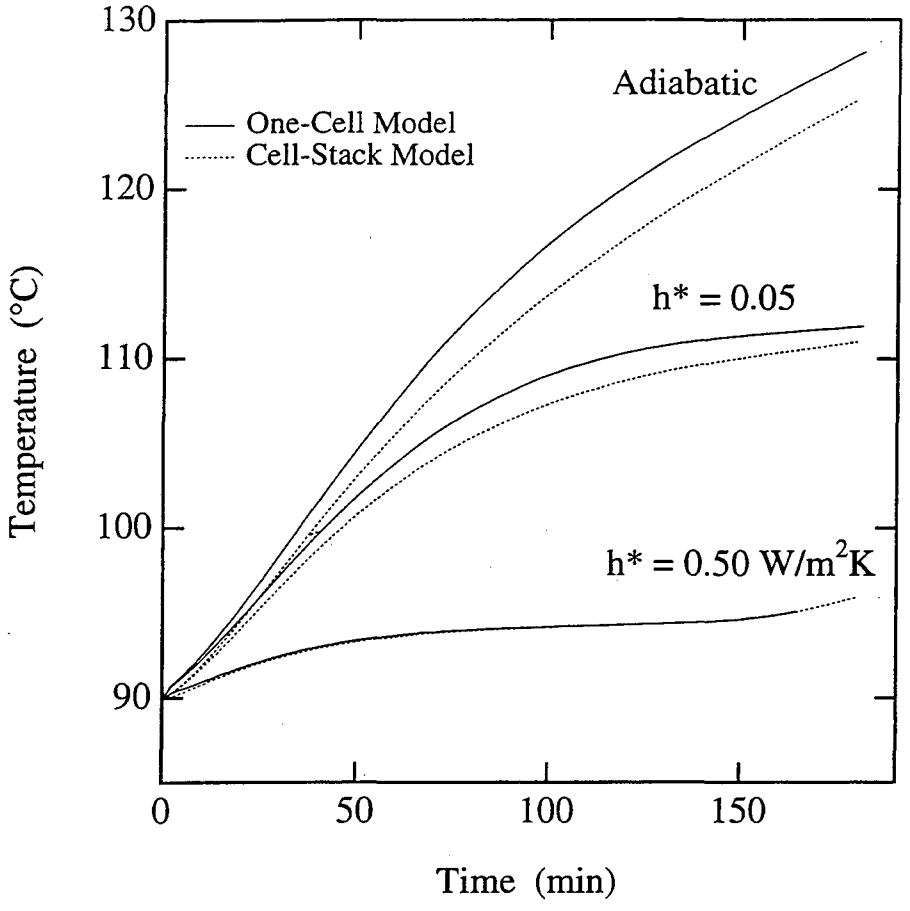

Figure 4-10. Comparison of temperatures as calculated from the one-cell model and from the cell-stack model for the same conditions shown in Figure 4-9. Per-cell heat transfer coefficients are given on the figure.

List of Symbols

A_s	separator area, cm^2
$\hat{C}_p$	mass-averaged constant-pressure heat capacity of cell, J/kg·K
h	convective heat-transfer coefficient, $W/m^2 \cdot K$
h*	per-cell heat-transfer coefficient, $W/m^2 \cdot K$
i	index for cell number
k_{cell}	thermal conductivity of cell, W/m·K
l_{cell}	thickness of the cell, m
M	mass of a single cell, kg
n	$N_{cell}/2$
N_{cell}	number of cells in a cell stack
q	heat transfer to surroundings, W/m^2
Q	heat-generation rate, W/m^2
t	time, s
T	temperature, K
Ta	ambient temperature, K

Subscripts

i	refers to cell number
n	refers to cell n or $N_{cell}/2$

References

1. J. Newman and W. Tiedemann, "Temperature Rise in a Battery Module with Constant Heat Generation," Submitted to *J. Electrochem. Soc.*

2. Y. Chen and J. Evans, "Heat Transfer Phenomena in Lithium/Polymer-Electrolyte Batteries for Electric Vehicle Application," *J. Electrochem Soc.*, **140**, 1833 (1993).

3. Y. Chen and J. Evans, "Thermal Analysis of Lithium Polymer Electrolyte Batteries by a Two Dimensional Model--Thermal Behaviour and Design Optimization," *Electrochimica Acta,* **39**, 517 (1994).

4. J. Newman, *Electrochemical Systems*, Prentice-Hall, Englewood Cliffs, NJ (1991).

Chapter 5: Cell-Stack Model Results

5.1 Introduction

Now we will examine temperature profiles in cell stacks as calculated by the cell-stack model presented in Chapter 4. Predicting temperature profiles is important because we would like to design a thermal-management system for the battery, as well as predict how the temperature variation in the battery will affect its performance. Several variables in the operation of the battery will be considered. We are interested in the temperature behavior of the battery as a function of the cell-stack thickness, heat-transfer coefficient, and discharge rate. All simulations were made using 90 °C for the initial cell temperature and 89 °C for the ambient air temperature.

We will also examine a proposed design for thermal management of an electric-vehicle battery system. Several specifications will be made for its design. As mentioned before, the electric-vehicle battery will be discharged at 1.1 mA/cm^2, the nominal three-hour rate. We will also allow a temperature rise of 20 °C in the battery so that the temperature range of operation will be from 90 °C to 110°C.

5.2 Temperature Profiles

Figure 5-1 shows temperature profiles of a 288-cell (7.1 cm) stack at the end of discharge for several currents. For these discharges, the stack is subjected to a convective-heat-transfer coefficient, h, of 10 W/m^2·K at its outer faces (x=L/2). This value of h corresponds to the low range of free convection of air. For the higher discharge rates shown, the overall cell temperature increases more, and the temperature profile is steeper than for the lower discharge rates. This behavior is expected because, as seen in Chapter 3, the heat-generation rate is higher for higher discharge rates. The 2.0-mA/cm^2 discharge reaches a maximum center temperature of 129.3 °C, and its

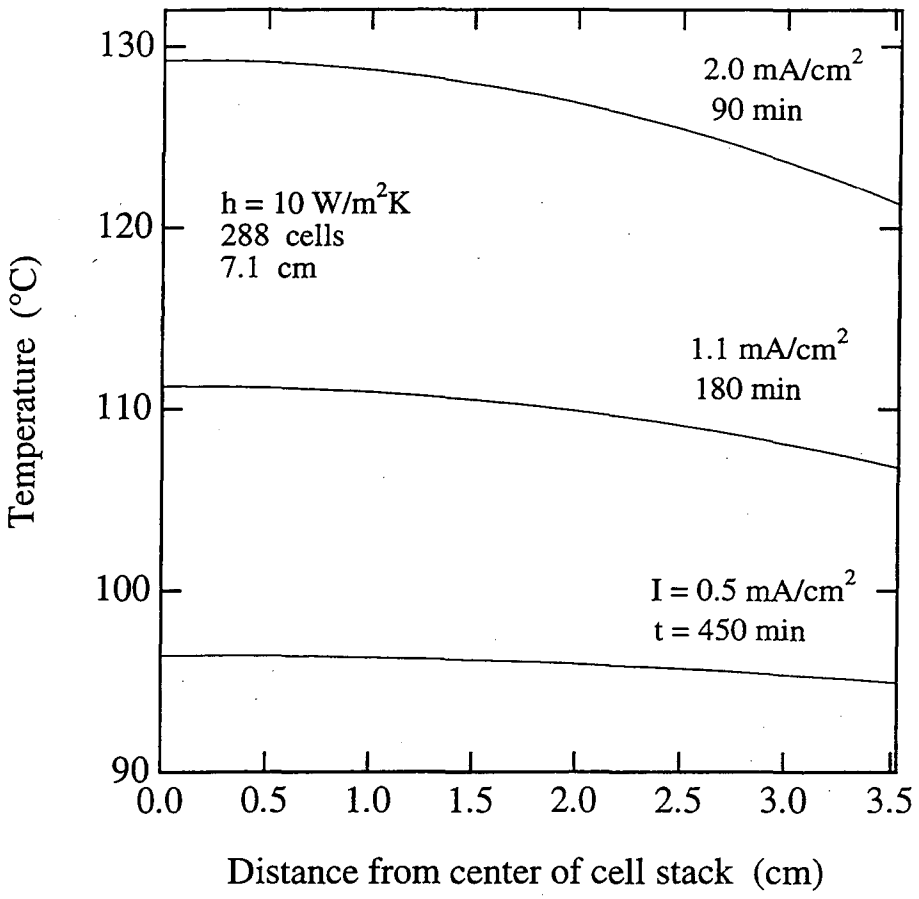

Figure 5-1. Temperature profiles at the end of discharge for galvanostatic discharge at various currents with a constant heat-transfer coefficient. The initial temperature was 90 °C, and the ambient air temperature was 89 °C.

variation from the center to the outer face of the cell stack is 9 °C. The discharge at 0.5 mA/cm^2 gives a center temperature rise of 6.4 °C with a stack-temperature variation of 1.4 °C. For the three discharge rates presented, the temperature variation at the end of discharge is approximately twenty percent of the temperature change at the center of the stack.

Since the discharge rate applicable to electric-vehicle applications is the nominal three-hour rate, the remaining results will be presented for a discharge rate of 1.1 mA/cm^2. Figure 5-2 shows temperature profiles at the end of discharge for several cell-stack thicknesses for three-hour rate discharges subject to a heat-transfer coefficient of 10 W/m^2·K. The figure shows that as the cell-stack thickness increases, the overall stack temperature increases, and the cell temperature profile gets steeper. The 144-cell stack has a 2 °C difference in temperature between the center and the outer face with a center temperature change of 15 °C, while the 576-cell stack has a 10 °C temperature difference and a center temperature change of 29 °C. This figure shows that as the stack thickness increases, the overall temperature rise, as well as the cell-stack temperature variation, increases.

Figure 5-3 shows temperature profiles in a 576-cell (14.1 cm) stack as functions of time for discharge at the three-hour rate subject to a convective-heat-transfer coefficient of 20 W/m^2·K. For early times, the temperature has not risen very much, and there is little temperature variation in the cell stack. As the discharge proceeds, however, the temperature variation increases. At the end of discharge, the temperature at the center of the stack reaches 117 °C while the temperature at the outer face reaches only 104 °C. For these conditions, the temperature variation of the cell stack will affect the battery's discharge behavior.

Figure 5-4 gives the heat-generation-rate profile for the same discharge shown in Figure 5-3. The figure shows that the heat-generation-rate profile is relatively flat at the

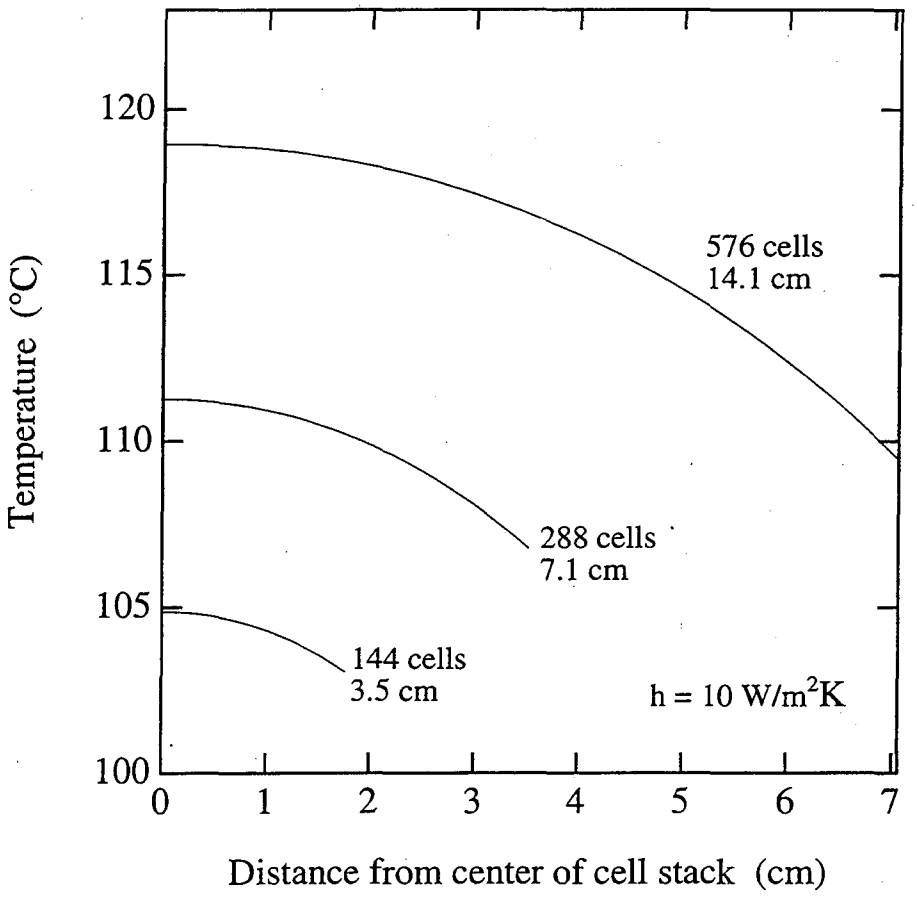

Figure 5-2. Temperature profiles at the end of discharge for cell stacks of several widths for galvanostatic discharge at 1.1 mA/cm^2. The initial stack temperature was 90 °C, and the ambient air temperature was 89 °C.

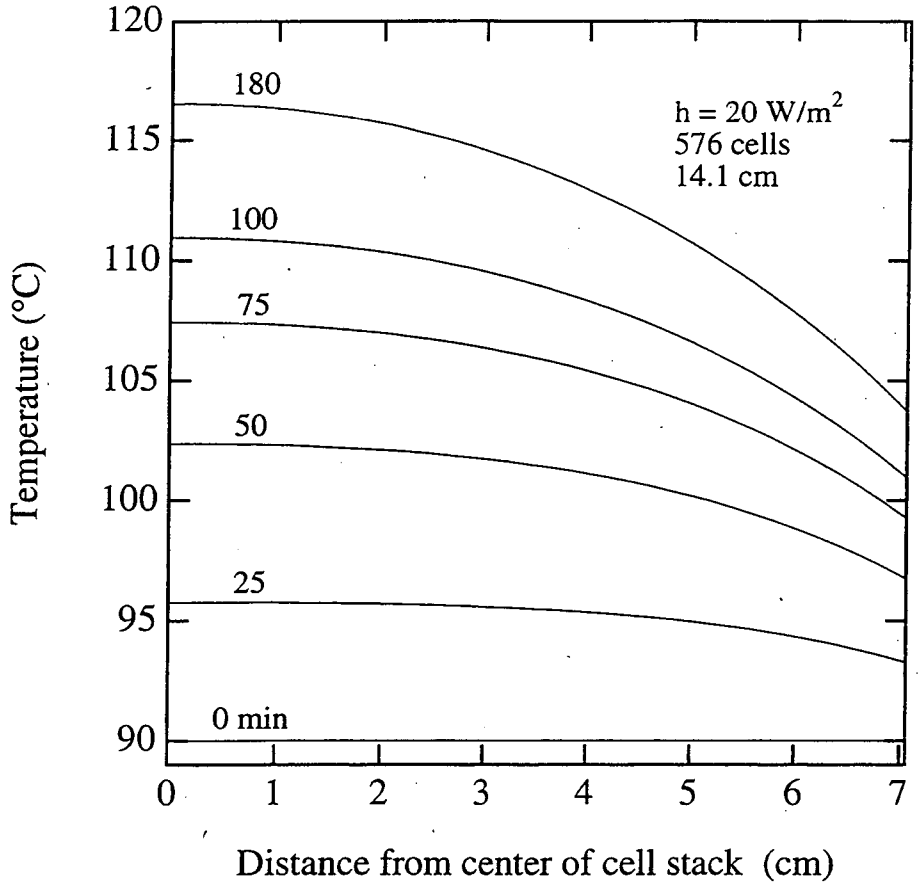

Figure 5-3. Temperature profiles as a funciton of time for galvanostatic discharge at 1.1 mA/cm^2, the three-hour rate. Times are given on the figure.

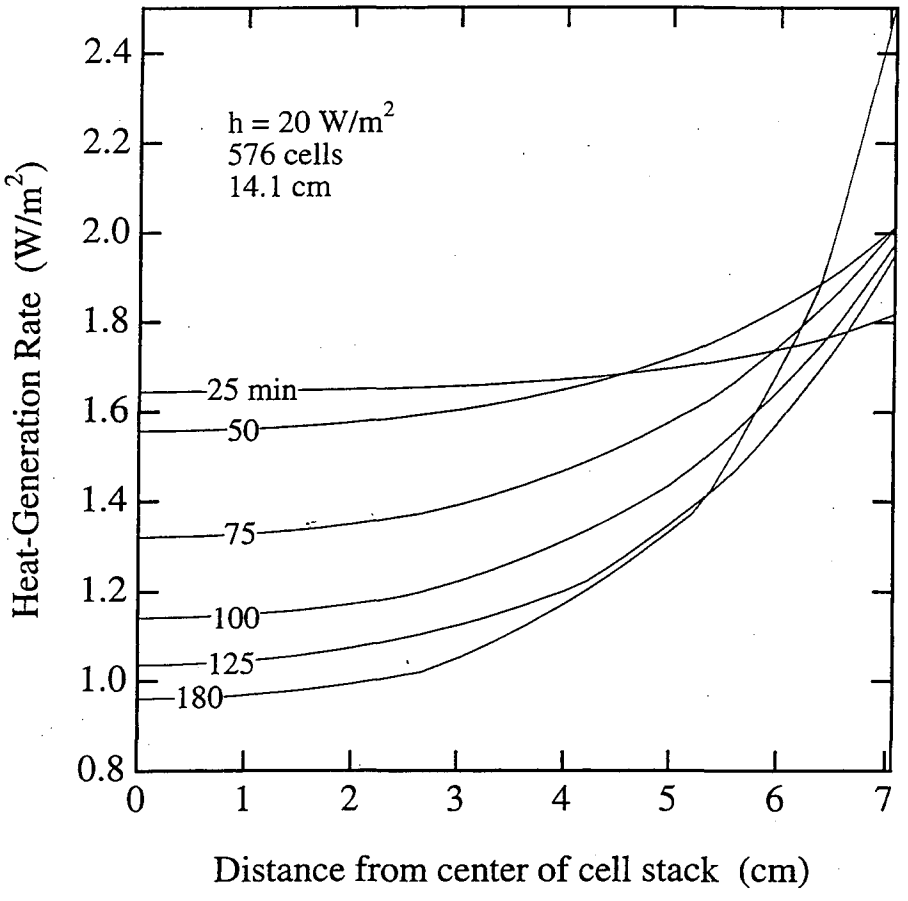

Figure 5-4. Per-cell heat-generation-rate profile as a function of time for the same discharge shown in Figure 5-3.

beginning of discharge. As the discharge proceeds, the heat-generation rate decreases through most of the cell stack, but a gradient develops. At the end of discharge, the cells at the outer face of the cell stack are generating heat at over twice the rate of the cells at the center of the stack. Because more heat is being produced in the cooler regions of the stack, this behavior tends to flatten the stack temperature profiles. Temperature profiles that are improperly calculated with a uniform heat-generation rate will therefore predict a temperature gradient that is too steep.

Figures 5-5 through 5-7 give temperature profiles at the end of discharge as functions of the convective-heat-transfer coefficient for different cell-stack thicknesses. Figure 5-5 gives temperature profiles for a 144-cell (3.5 cm) stack, Figure 5-6 for a 288-cell (7.1 cm) stack, and Figure 5-7 for a 576-cell (14.1 cm) stack. The figures show general trends of the effect of stack thickness and heat-transfer coefficient on temperatures and temperature profiles of the stack. All of these figures show that as the heat-transfer coefficient increases, the overall stack temperature rise decreases, and the temperature variation across the cell stack increases. They also show that for a fixed value of the convective-heat-transfer coefficient, as the stack thickness increases, both overall stack temperature, and stack temperature variation, increase.

5.3 Electric-Vehicle Battery

Following the specifications for an electric-vehicle battery outlined earlier, we shall propose an electric-vehicle battery configuration based on facile temperature control. Specifications for the proposed electric-vehicle battery are given in Table 5-1.

Since we have specified the stack voltage to be between 240 and 360 V, and the voltage of the TiS_2 cell ranges from 2.5 V to 1.7 V, we will set the cell-stack thickness to 144 cells. The cell stack will then have a voltage of 360 V at the beginning of discharge, and 244 V at cutoff. The time-averaged voltage will be 288 V.

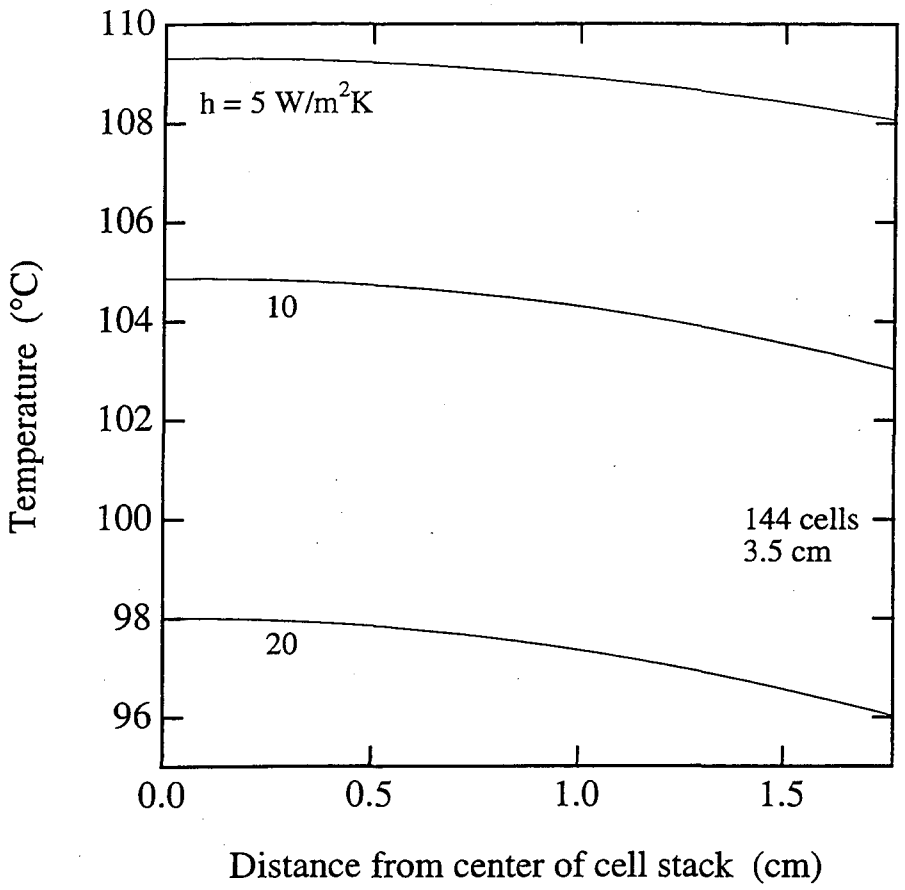

Figure 5-5. Temperature profiles for a 144-cell (3.5 cm) stack at the end of discharge at 1.1 mA/cm^2 as functions of the convective heat-transfer coefficient.

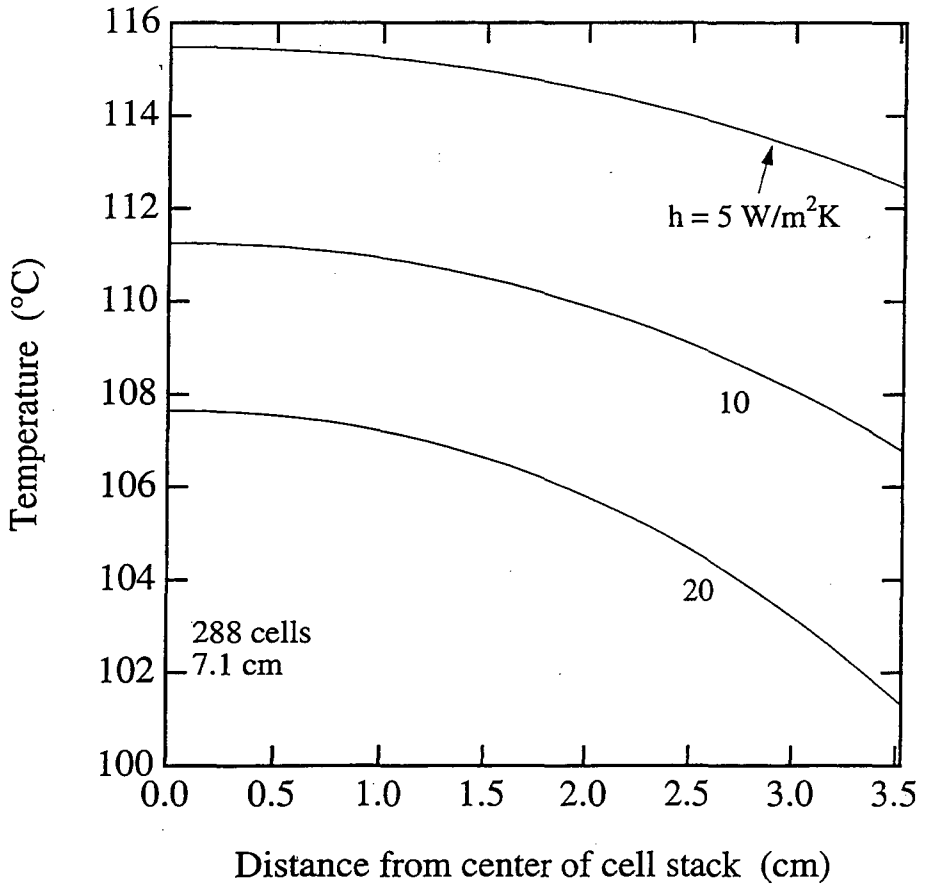

Figure 5-6. Temperature profiles for a 288-cell (7.1 cm) stack at the end of a 1.1 mA/cm^2 discharge as functions of the convective heat-transfer coefficient.

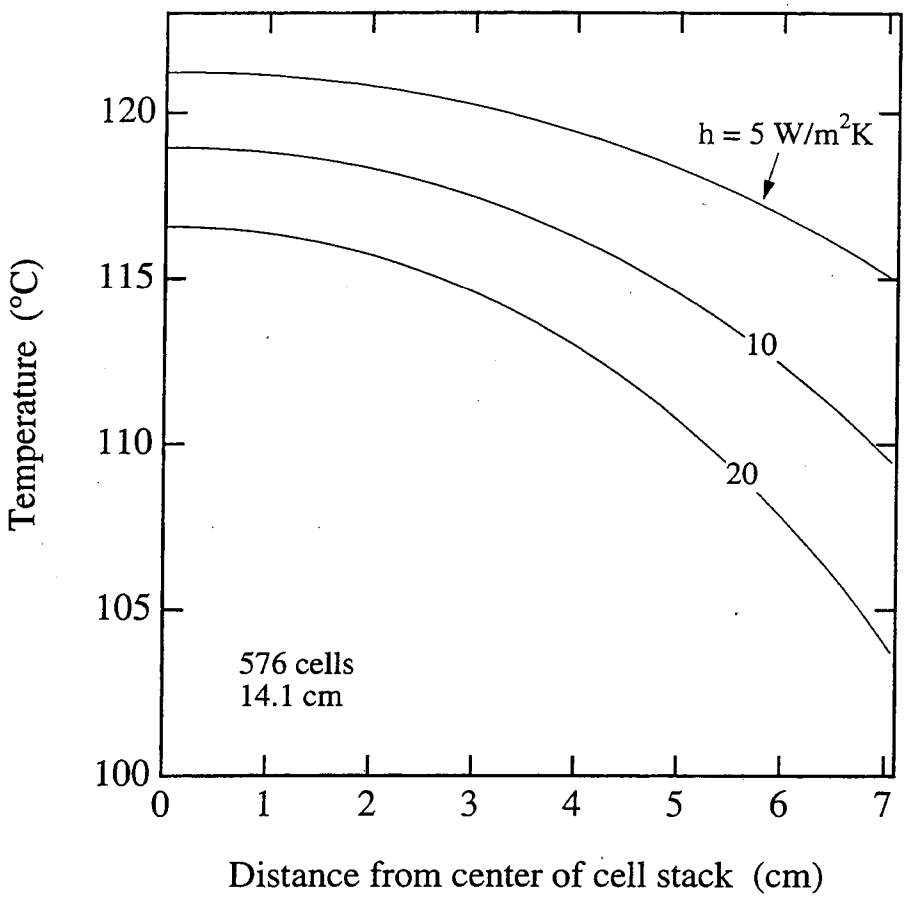

Figure 5-7. Temperature profiles for a 576-cell (14.1 cm) stack at the end of discharge at 1.1 mA/cm^2 as functions of the convective heat-transfer coefficient.

Table 5-1. Design specifications for an electric-vehicle battery.

Parameter	Specification
Voltage	240 to 360 V
Temperature rise	20 °C
Discharge Time	3 hr
Capacity	40 kW·h

Next we specify the battery capacity to be 40 kW·h. At a discharge rate of 1.1 mA/cm^2, and a time-averaged voltage of 2.0 V, we would need 606 m^2 of cells. If we set the battery to be 0.51 m by 0.51 m, we then need 2330 cells. With 144 cells per stack, we need 16 cell stacks. The cell stacks would be connected electrically in parallel. Now if we place a 1-cm cooling channel between every two cell stacks, the battery dimensions become 0.51 m by 0.51 m by 0.64 m. The total battery volume for this design is, therefore, 0.166 m^3, and its weight is 280 kg. The volume and weight include cell elements shown in Figure 2-1: electrodes, separators, and bipolar partitions, but not casing and other peripherals.

For a 288-cell stack (two 144-cell stacks connected thermally in series, but electrically in parallel), a heat-transfer coefficient that allows a temperature rise of less than 20 °C in the center of the stack is 17 $W/m^2 \cdot K$. This value of the convective heat-transfer coefficient corresponds to the free-convection-of-air range. Figure 5-8 shows temperature profiles as functions of time for discharge of a 288-cell stack at the three-hour rate subjected to a convective-heat-transfer coefficient of 17 $W/m^2 \cdot K$. The temperature profile is relatively flat throughout discharge. After 25 minutes, the stack temperature has risen to just over 95 °C with a 2 °C temperature variation. At 180

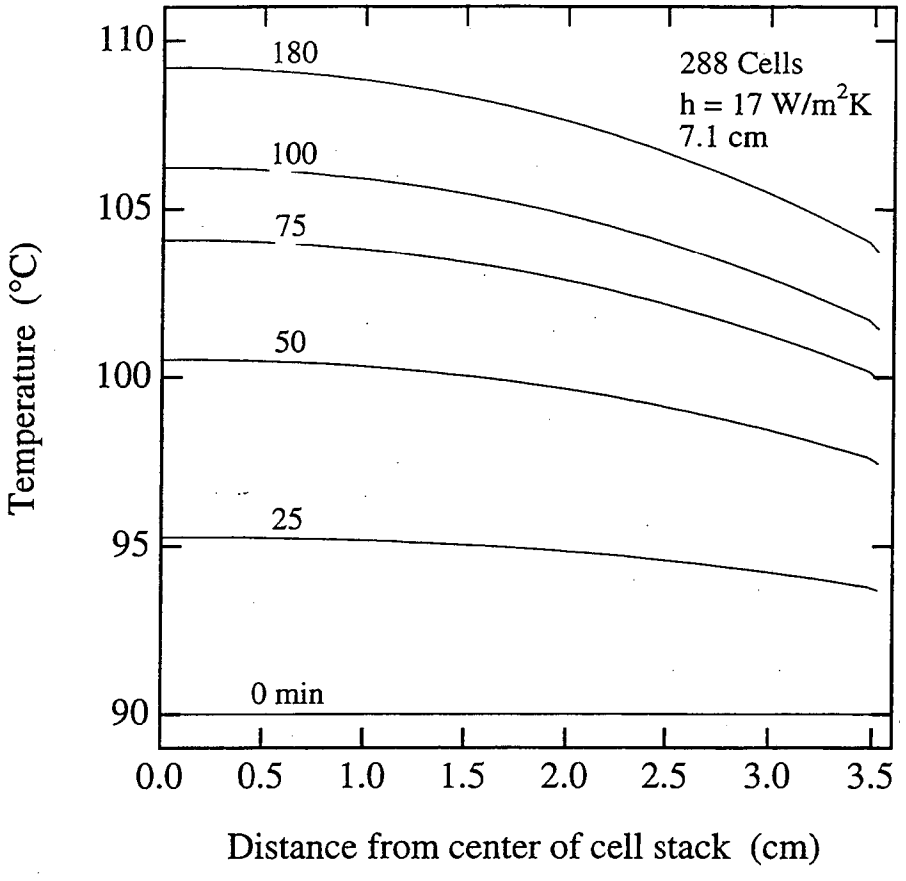

Figure 5-8. Temperature profiles as a function of time for the proposed electric-vehicle battery design. Discharge time is given on the figure.

minutes, the average temperature in the stack is 107 °C, and the variation from the outer face to the center of the stack is 5 °C.

Figure 5-9 and 5-10 give the temperature and per-cell heat-generation rate, respectively, as functions of time as calculated by the one-cell and cell-stack models. The per-cell heat-transfer coefficients for the given cells are calculated from Equations 3-9 through 3-11. For the cells in the middle of the cell stack, x=0 and x=L/4, the models show good agreement for both heat-generation rate and temperature. The models predict a center temperature of 108 °C at the end of discharge, with a variation between models of less than 0.5 °C.

The models disagree more strongly on both temperature and heat-generation rate for the outer face of the cell stack (x=L/2). The one-cell model predicts a final temperature of 105 °C, while the cell-stack model predicts a final temperature of 102.5 °C. This temperature variation between models is apparent throughout the discharge. At the beginning of discharge the x=L/2 curve calculated by the cell-stack model deviates from the other curves, which all have approximately the same slope for the first 30 minutes, corresponding to an adiabatic temperature rise. By about 60 minutes into the discharge, the cell-stack temperature for x=L/2 has reached its maximum difference from the calculations of the one-cell model, and after this time the difference remains relatively constant at 2.5 °C to the end of discharge.

The difference between the cell-stack and the one-cell calculations for the temperature of the cell at the outer face of the cell stack can be explained by examining differences in the development and use of the models. First, the cell-stack model considers a system of different cells connected thermally in series and exposed to a heat-transfer coefficient only on its edges. The cells in the cell-stack model vary in both temperature and heat-generation rate from their neighboring cells, and therefore a temperature gradient is established in the cell stack. The one-cell model, on the other

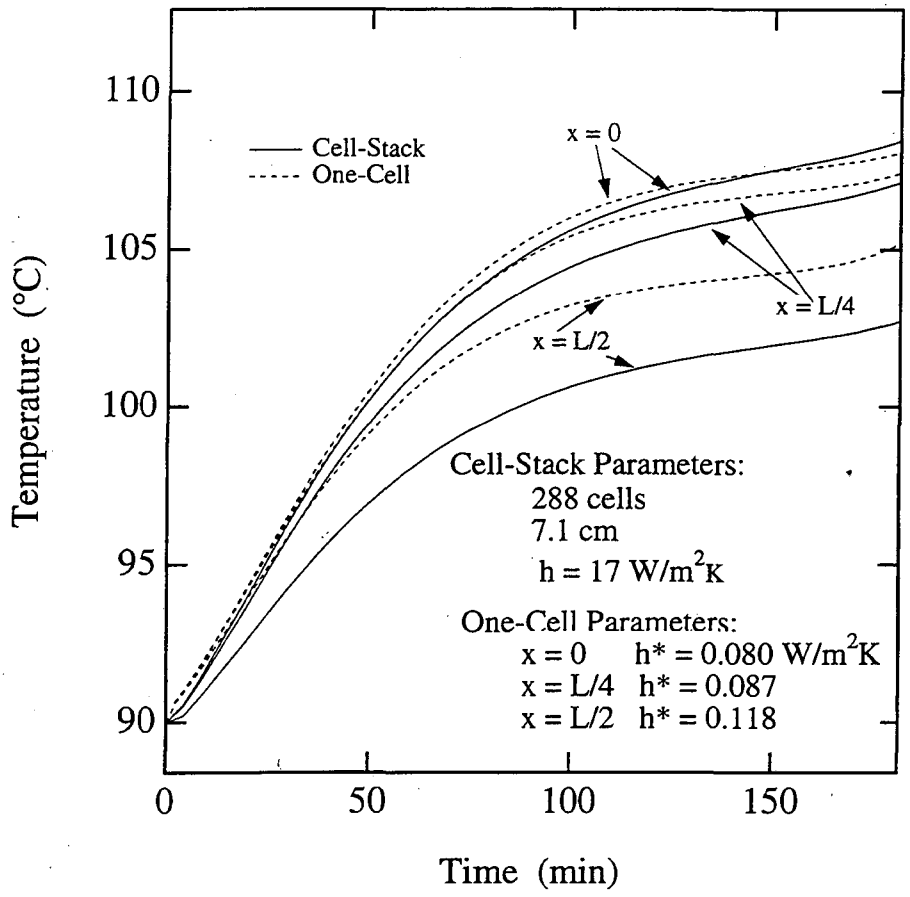

Figure 5-9. Comparison of temperature as a function of time for the proposed electric-vehicle battery design discharged at 1.1 mA/cm^2 as calculated by the one-cell and cell-stack models.

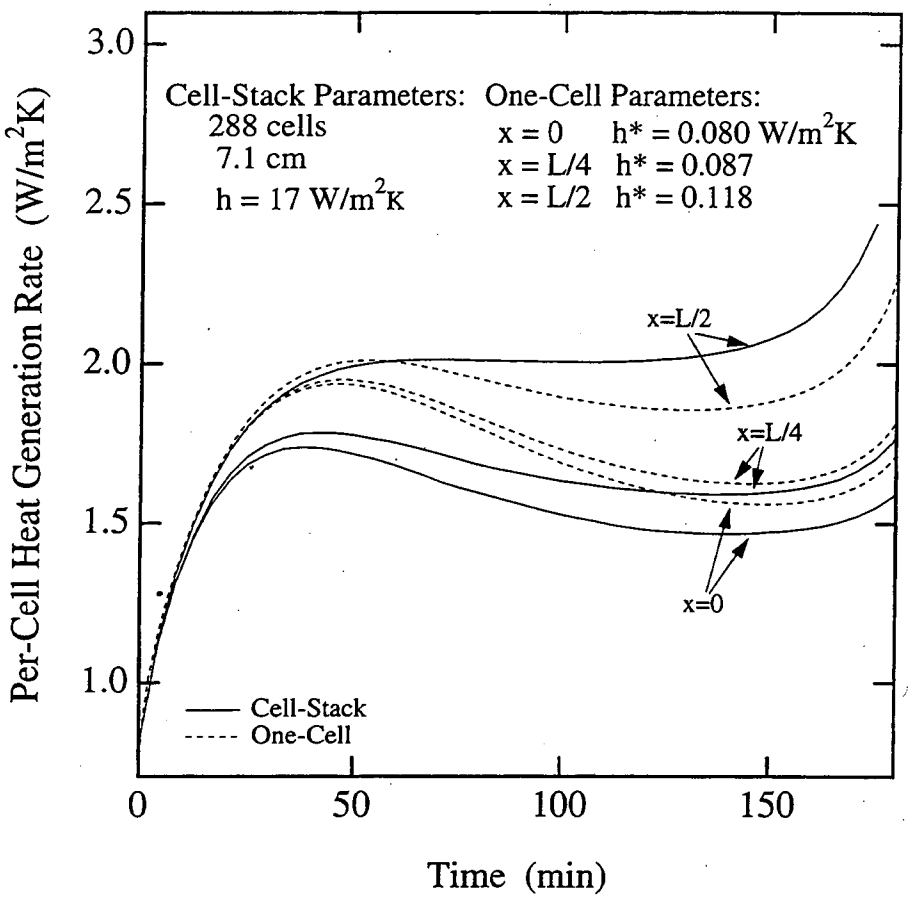

Figure 5-10. Comparison of heat-generation rate as a function of time for the same simulations presented in Figure 5-9.

hand, considers one cell with a uniform temperature and heat-generation rate with heat transfer from the cell calculated using a per-cell heat-transfer coefficient based on the position of the cell in the cell stack.

Figure 5-9 shows that the variation between the temperature of the outer-face (x=L/2) cell calculated by the two models begins early in the discharge. This happens even though the heat-generation rates for all of the cells in the cell stack are equal at this time. This is because in the cell-stack model, the cell at the outer face of the stack is exposed to the full value of the heat-transfer coefficient and is transferring heat to the surroundings, but it does not receive appreciable heat from the internal cells until later in discharge when temperature profiles develop. This causes the temperature to rise much more slowly than for the one-cell model where the heat-transfer coefficient for the end cell is much lower. For the one-cell model, the time constant for heat transfer to the surroundings is $\frac{M\hat{C}_p}{h^*}$, a very large number, while for the cell-stack model, the time constant is $\frac{M\hat{C}_p}{h}$, a relatively small number. This means that in the cell-stack model, the cell at x=L/2 transports more heat to the surroundings earlier in discharge and will not appreciably heat up until the temperature profiles in the cell stack are developed and it can receive heat from the internal cells. Thus the edge-cell temperature calculated by the cell-stack model rises more slowly and reaches a lower value than its corresponding temperature calculated by the one-cell model. Since it takes into account all of the cells, and not one cell with similar heat transfer conditions, the calculation for the edge temperature by the cell stack-model is more accurate than that by the one-cell model.

Figure 5-11 gives the cell potential as a function of time for the proposed design as calculated by the one-cell model. The figure shows that the potential variation for the selected conditions is very small throughout the discharge. At the end of discharge, the potential for the cell at the center of the stack is 1.79 V while that at the edge of the stack

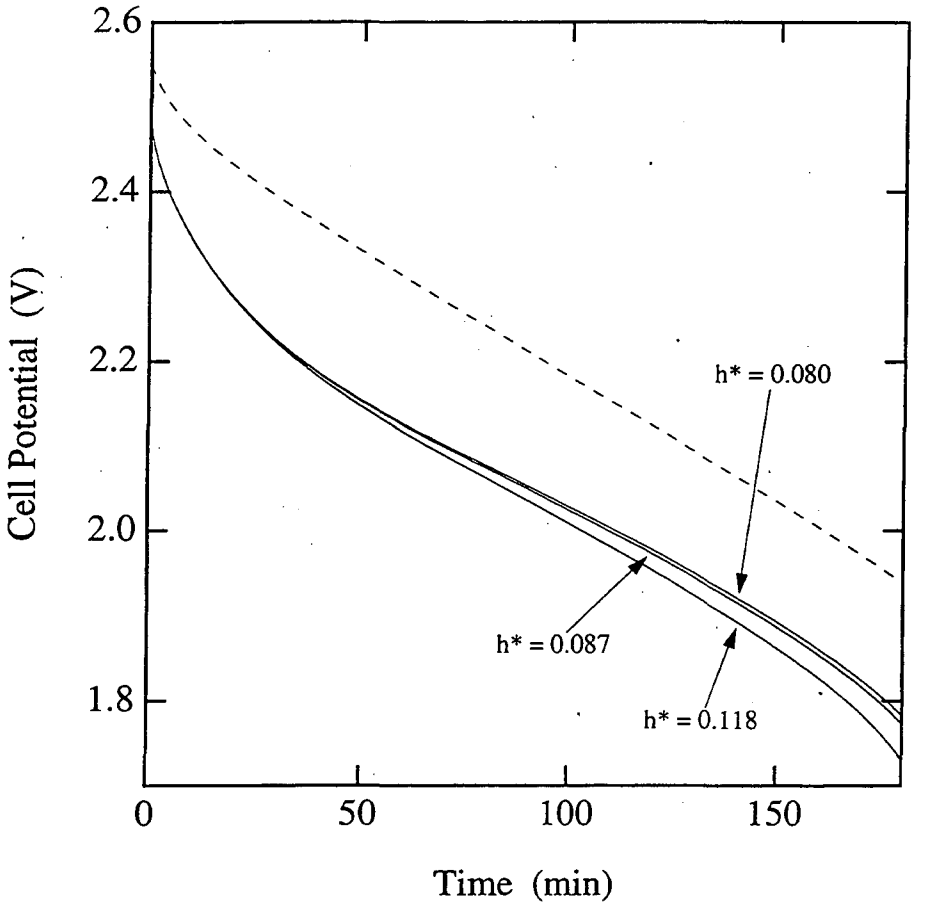

Figure 5-11. Cell potential as a function of time and position in the cell stack as calculated by the one-cell model for the proposed electric-vehicle battery configuration. The dashed line is the open-circuit potential of the cell. The position of the cell determines the value of its per-cell heat-transfer coefficient.

is 1.74 V. Although this temperature gradient in the cell stack may affect cell life, it will not affect to a great extent the operation of the battery during discharge.

5.4 Conclusion

Using the cell-stack model presented in Chapter 4, temperature profiles have been calculated and presented for cell stacks of varying thickness subject to varying heat-transfer conditions. From examination of a heat-generation-rate profile, it was concluded that heat was generated at lower rates in higher-temperature areas of the stack, and at higher-rates in lower-temperature areas of the stack. This type of non-uniform heat-generation-rate profile tends to flatten the temperature profile of the cell stack.

A design for thermal management of an electric-vehicle battery has been proposed utilizing the $LiPEO_{18}|LiCF_3SO_3|TiS_2$ cell. For a specified capacity of 40 kW·h, the battery would require 606 m^2 of separator area, and if cells were 0.51 m by 0.51 m, 2330 cells would be needed. If the cells were arranged in 16 stacks of 144 cells with 1-cm cooling channels between every two stacks, the battery would then be 0.64 m long. For this configuration, a heat-transfer coefficient of 17 $W/m^2 \cdot K$, corresponding to the lower range of the free-convection-of-air regime, would be required to keep the center temperature from exceeding 110 °C. For the above conditions, the center temperature would reach 108 °C, and the edges would reach 103 °C. The temperature gradient in the stack would be only 5 °C. Since the lower-temperature cells produce heat at a higher rate than the higher-temperature cells, the temperature profile is not as steep as would be calculated by a model that assumes a uniform energy-generation rate.

Calculations from the cell-stack model were compared to calculations from the one-cell model for the proposed electric-vehicle battery configuration. The one-cell- and

cell-stack-model calculations agreed well for temperatures in the center of the cell stack, but disagreed for cells on the edge of the cell stack. The calculated temperature at the edge of the cell stack was lower for the cell-stack model than for the one-cell model because of differences in the time-constants for heat transfer. Because of the methods in which the models were developed, the one-cell model is more accurate for calculating temperatures than the cell-stack model for cell stacks with negligible temperature gradients, and the cell-stack model is more accurate for calculating temperature profiles in stacks with appreciable temperature gradients. For stacks with appreciable temperature gradients, the one-cell model can still be used for the calculation of cell potentials for different cells in the cell stack, but the error due to the higher calculated temperature at the edge of the stack must be taken into account.

Appendix A: One-Cell Program and Input Files

A.1 Program for One-Cell Model: tfoil1.f

```
c*****************************************************
c
c     tfoil1.f
c
c     version 1.0x August 15, 1994
c
c     Thermal model of the lithium/polymer battery
c
c*****************************************************
      implicit double precision*8(a-h,o-z)
      common /n/ nx,nt,n1,n2,nj
      common /calc/ ai(1600),u(222,1600),ts(1600),h,h1,h2,h3,rr
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/power/ ed,Vold
      common/ssblock/ xp0(5),xx0(5,221),term(221)
      common/var/ xp(10),xx(5,221),xi(5,221),xt(5,221,1600)
      common/cprop/ sig,area,ct,dfs,Rad,cmax,rka,rkf
      common/tprop/df(221),cd(221),tm(221),
     1ddf(221),dcd(221),dtm(221),dfu(221),d2df(221),
     1d2tm(221),d2fu(221)
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
      dimension tt(16),cu(16)
      open(unit=10, file = 'tfoil.in', status='old')

   44 format(/'          mass  = ',f7.4,' kg/m2')
   45 format('specific energy  = ',f8.2,' W-h/kg')
   46 format('specific power   = ',f8.2,' W/kg')
c
      n=5
c     n is number of equations
      rr=1.0d0
c     intial time step is 1 second
      data fc/96487.0d0/, r/8.314d0/, pi/3.141592653589d0/
c
c*****************************************************
c     read in parameters and boundary conditions
c
      read(10,*)lim,h1,h2,h3,n1,n2,t,tsmax,tsmin
c
      n1=n1+1
      nj = n1 + n2
      rr = tsmin
c
      read(10,*)(xi(i,1), xi(i,nj), i=1,2)
      read(10,*)(xi(i,n1), xi(i,nj), i=3,n),cur
      read(10,*)dfs,Rad,ep,epf,eps,sig
      read(10,*)ct,cmax,rka,rkf,il4
      read(10,*)re,rs,rf,rc,rp,ef
      read(10,*)il1,il2,il3
```

```
      read(10,*)htc,dudt,Cp,dens,tam,ncell,lht
      read(10,*) tmax
      read(10,*)(cu(i),tt(i), i=1,tmax)
c
      tam=tam+273.15
      t=t+273.15
      tin=t

      tt(1)=6.0d01*tt(1)
      do i=2,tmax
      tt(i)=tt(i-1)+6.0d01*tt(i)
      end do
c
c     rr is the size of a time step and should equal dt times
c     the radius squared over dfs, nt the number of steps
      dt=rr*dfs/Rad/Rad
      ts(1)=0.0d0
c
      area=3.0d0*(1.0d0-ep-epf)/Rad
      sig=sig*((1.0d0-ep-epf)**(1.5d0))
      hl=h1+h2
      h1=h1/(n1-1)
      h2=h2/n2
      h=h1
      frt=fc/(r*t)
c
      print*,'   FOIL VERSION 1.0x'
      print*,' '
      print*,' util  ','  cell pot  ',' material','  time'
      print*,'  y   ','    (V)     ','balance','   (min)'
c
c*************************************************
c
      call guess(n)
c
c%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
c
      cur=cu(1)
      call zts(n,lim,2,dt,1)
      k=1
c
      do l=1,tmax
  123 k=k+1
      nt=k-1
c
c     adjust time step to match time of change in current
c
      if (time .gt. tt(l)) then
      dt=tt(l)*dfs/Rad/Rad-ts(k-1)
      rr=dt*Rad*Rad/dfs
      iflag=1
      else
      continue
      end if
```

```
c
      ts(k)=ts(k-1)+dt
      call calca(k)
      call comp(n,lim,k,dt,1)
c     sets values for time step k
c
      do 10 i=1,n
      do 10 j=1,nj
   10 xt(i,j,k)=xx(i,j)
c
      call cellpot(k,il4,vv,1)
c
c     anticipate time at next time step
      time=(ts(k)+dt)*Rad*Rad/dfs
c
      if (iflag .eq. 0) go to 123
      cur=cu(l+1)
      iflag=0
c     calculate zero time solution for change in current
      if (l .lt. tmax) call zts(n,lim,k,dt,k)
      rr=2.0d0
      dt=rr*dfs/Rad/Rad
      end do
c
c%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
c
c      call peak(n,lim,il4)
c     call util
      if(il1 .eq. 1) call nucamb(n,il2,il3)
      call mass(tw,re,rs,rf,rc,rp,ef)
      ed=ed/tw/3.6d03
      pow=3.6d03*dfs*ed/ts(nt+1)/Rad/Rad
      print44,tw
      print45,ed
      print46,pow
      end
c
c*****************************************************
c*****************************************************
      subroutine comp(n,lim,kk,tau,li)
      implicit double precision*8(a-h,o-z)
      common /n/ nx,nt,n1,n2,nj
      common /calc/ ai(1600),u(222,1600),ts(1600),h,h1,h2,h3,rr
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/ssblock/ xp0(5),xx0(5,221),term(221)
      common/var/ xp(10),xx(5,221),xi(5,221),xt(5,221,1600)
      common/cprop/ sig,area,ct,dfs,Rad,cmax,rka,rkf
      common/tprop/df(221),cd(221),tm(221),
     1ddf(221),dcd(221),dtm(221),dfu(221),d2df(221),
     1d2tm(221),d2fu(221)
      common/mat/ b,d
      common/bnd/ a,c,g,x,y
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
      dimension b(10,10),d(10,21)
```

```
      dimension a(10,10),c(10,221),g(10),x(10,10),y(10,10)
c
   99 format (1h ,//5x,'this run did not converge'//)
      nx=n
c
  666 continue
      if ( li .eq. 1) then
      do 1 j=1,nj
      do 1 i=1,n
          c(i,j)=xt(i,j,kk-1)
    1     xx(i,j)=xt(i,j,kk-1)
      else
      do 81 j=1,nj
      do 81 i=1,n
          c(i,j)=xt(i,j,kk)
   81     xx(i,j)=xt(i,j,kk)
c  81     c(i,j)=xx(i,j)
      endif
c     sets first guess to last time-step values
      jcount=0
      do 4 i=1,n
    4     xp(i)=0.0d0
c     initialize variables to begin each iteration (jcount is iteration #)
    8 j=0
      jcount=jcount+1
      call prop(nj,n1)
c     obtains physical properties at this specific point
      do 9 i=1,n
      do 9 k=1,n
          x(i,k)=0.0d0
    9     y(i,k)=0.0d0
c
c     store previous iteration of  (xp in xp0)  &  (xx in xx0)
      do 6 i=1,n
         xp0(i)=xp(i)
    6     xx0(i,n1+10)=xx(i,n1+10)
          xx0(n,n1+10)=xx(n,n1+10)
c
c     for a given iteration, set up governing equations and bc's
c     start at the left interface and move across polymer
c
   10 j=j+1
c
      do 11 i=1,n
          g(i)=0.0d0
          xx(i,j)=c(i,j)
      do 11 k=1,n
          a(i,k)=0.0d0
          b(i,k)=0.0d0
   11     d(i,k)=0.0d0
c     clears all arrays before use
      if(j.ne.1) go to 13
c
c     ____________________________________________________
```

```
c
c    specify boundary conditions at left interface (j=1)
c    boundary conditions at negative electrode
     h=h1
c
     dcf=(-xx(1,j+2) + 4.0d0*xx(1,j+1) - 3.0d0*xx(1,j))/2.0d0/h
     b(1,1)=-df(j)*1.5d0/h + ddf(j)*dcf
     d(1,1)=df(j)*2.0d0/h
     x(1,1)=-df(j)*0.5d0/h
     g(1)=(tm(j)-1.0d0)*cur/fc + ddf(j)*dcf*xx(1,j)
c
     dcf=(xx(1,j+1)-xx(1,j))/h
     r1=(xx(1,j+1)+xx(1,j))/2.0d0
     p1=(tm(j)+tm(j+1))/2.0d0
     p2=(cd(j)+cd(j+1))/2.0d0
     p3=(dtm(j)+dtm(j+1))/2.0d0
     p4=(dfu(j)+dfu(j+1))/2.0d0
     p6=(d2fu(j)+d2fu(j+1))/2.0d0
     d(2,1)=(1.0d0-p1)*(1.0d0/r1+p4)/h
     b(2,1)=-d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
    1      -(1.0d0/r1+p4)*dcf*p3)/2.0d0
     d(2,1)=d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
    1      -(1.0d0/r1+p4)*dcf*p3)/2.0d0
     d(2,2)=fc/r/t/h
     b(2,2)=-fc/r/t/h
     g(2)=fc/r/t*cur/p2+(1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf*r1 -
    1    (1.0d0/r1+p4)*dcf*p3*r1
c
     b(3,5)=1.0d0
     g(3)=0.0d0
     b(4,3)=1.0d0
     g(4)=0.0d0
     b(5,4)=1.0d0
     g(5)=cur
c
     call band(j)
     go to 10
c
c    ____________________________________________________
c    specify governing equations in polymer separator
 13  if (j .ge. n1) go to 110
c
     dc=(xx(1,j+1) - xx(1,j-1))/2.0d0/h
     d2c=(xx(1,j+1)-2.0d0*xx(1,j)+xx(1,j-1))/h/h
     a(1,1)=-df(j)/2.0d0/h/h + dc*ddf(j)/h/2.0d0 -
    1dtm(j)*xx(4,j)/fc/4.0d0/h
     b(1,1)=eps/rr + df(j)/h/h - d2df(j)*dc*dc/2.0d0
    1-d2c*ddf(j)/2.0d0 + xx(4,j)*d2tm(j)*dc/fc/2.0d0
     d(1,1)=-df(j)/2.0d0/h/h - dc*ddf(j)/h/2.0d0 +
    1dtm(j)*xx(4,j)/fc/4.0d0/h
     b(1,4)=dtm(j)*dc/2.0d0/fc
     g(1)=eps*xt(1,j,kk-1)/rr - dc*dc*d2df(j)*xx(1,j)/2.0d0
    1-dc*dc*ddf(j)/2.0d0 - d2c*ddf(j)*xx(1,j)/2.0d0 + term(j)/2.0d0+
    1xx(4,j)*d2tm(j)*dc*xx(1,j)/fc/2.0d0 + dtm(j)*xx(4,j)*dc/fc/2.0d0
```

```
c
      dcf=(xx(1,j+1)-xx(1,j))/h
      r1=(xx(1,j+1)+xx(1,j))/2.0d0
      p1=(tm(j)+tm(j+1))/2.0d0
      p2=(cd(j)+cd(j+1))/2.0d0
      p3=(dtm(j)+dtm(j+1))/2.0d0
      p4=(dfu(j)+dfu(j+1))/2.0d0
      p5=(dcd(j)+dcd(j+1))/2.0d0
      p6=(d2fu(j)+d2fu(j+1))/2.0d0
      d(2,1)=(1.0d0-p1)*(1.0d0/r1+p4)/h
      b(2,1)=-d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
     1      -(1.0d0/r1+p4)*dcf*p3)/2.0d0
      d(2,1)=d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
     1      -(1.0d0/r1+p4)*dcf*p3)/2.0d0
      d(2,2)=fc/r/t/h
      b(2,2)=-fc/r/t/h
      g(2)=fc/r/t*cur/p2+(1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf*r1 -
     1    (1.0d0/r1+p4)*dcf*p3*r1
c
      b(3,5)=1.0d0
      g(3)=0.0d0
      b(4,3)=1.0d0
      g(4)=0.0d0
      b(5,4)=1.0d0
      g(5)=cur
      call band(j)
      go to 10
c
c     Now for the boundary between cathode and separator(j=n1):
  110 if (j .ne. n1) go to 120
      sum=0.0d0
c     Material balance across box at n1 gives:
c
      dc=(xx(1,j)-xx(1,j-1))/h1
      dc2=(xx(1,j+1)-xx(1,j))/h2
      r1=(xx(4,j)+xx(4,j-1))/2.0d0
      r2=(xx(4,j+1)+xx(4,j))/2.0d0
      r3=(xx(1,j)+xx(1,j-1))/2.0d0
      r4=(xx(1,j)+xx(1,j+1))/2.0d0
      p1=(df(j-1)+df(j))/2.0d0
      p2=((ep**1.5)*df(j)/(eps**1.5)+df(j+1))/2.0d0
      p3=(dtm(j)+dtm(j+1))/2.0d0
      p4=(dtm(j-1)+dtm(j))/2.0d0
      p5=(tm(j+1)+tm(j))/2.0d0
      p6=(tm(j)+tm(j-1))/2.0d0
      p7=((ep**1.5)*ddf(j)/(eps**1.5)+ddf(j+1))/2.0d0
      p8=(ddf(j)+ddf(j-1))/2.0d0
      a(1,1)=-p1/h1/h2+dc*p8/2.0d0/h1-p4*cur/2.0d0/fc/h2
      b(1,1)=p1/h1/h2 + p2/h2/h2 + (eps*h1/h2 + ep)/rr
      b(1,1)=b(1,1)+dc*p8/2.0d0/h1-p4*cur/2.0d0/fc/h2
     1-p7*dc2/2.0d0/h2+p3*r2/2.0d0/fc/h2
      d(1,1)=-p2/h2/h2-p7*dc2/2.0d0/h2+p3*r2/2.0d0/fc/h2
      b(1,4)=p5/fc/h2/2.0d0
      d(1,4)=p5/fc/h2/2.0d0
```

```
      b(1,5)=-3.0d0*area/8.0d0
      d(1,5)=-area/8.0d0
      g(1)=(h1*eps/h2 + ep)*xt(1,j,kk-1)/rr + term(j)
     1+p6*cur/fc/h2
c
      dcf=(xx(1,j+1)-xx(1,j))/h
      r1=(xx(1,j+1)+xx(1,j))/2.0d0
      r4=(xx(4,j+1)+xx(4,j))/2.0d0
      p1=(tm(j)+tm(j+1))/2.0d0
      p2=((ep**1.50d0)*cd(j)/(eps**1.5d0)+cd(j+1))/2.0d0
      p3=(dtm(j)+dtm(j+1))/2.0d0
      p4=(dfu(j)+dfu(j+1))/2.0d0
      p5=((ep**1.50d0)*dcd(j)/(eps**1.5d0)+dcd(j+1))/2.0d0
      p6=(d2fu(j)+d2fu(j+1))/2.0d0
      d(2,1)=(1.0d0-p1)*(1.0d0/r1+p4)/h
      b(2,1)=-d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
     1      -(1.0d0/r1+p4)*dcf*p3+fc/r/t*r4*p5/p2/p2)/2.0d0
      d(2,1)=d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
     1      -(1.0d0/r1+p4)*dcf*p3+fc/r/t*r4*p5/p2/p2)/2.0d0
      d(2,2)=fc/r/t/h
      b(2,2)=-fc/r/t/h
      b(2,4)=-fc/r/t*(1.0d0/p2+1.0d0/sig)/2.0d0
      d(2,4)=b(2,4)
      g(2)=-fc/r/t*cur/sig + (1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf*r1
     1    -(1.0d0/r1+p4)*dcf*p3*r1+fc/r/t*r4*p5*r1/p2/p2
c
      call ekin(j)
c
      if (kk .gt. 2) then
      sum=0.0d0
      do i=1, kk-2
      sum=sum + (xt(3,j,i+1)-xt(3,j,i))*ai(kk-i)/(ts(i+1)-ts(i))
      end do
      else sum=0.0d0
      end if
      b(4,3)=ai(1)/rr
      b(4,5)=1.0d0/Rad
      g(4)=ai(1)*xt(3,j,kk-1)/rr -sum*dfs/Rad/Rad
      b(5,4)=1.0d0
      g(5)=cur
c
      call band(j)
      go to 10
c
  120 if (j .eq. nj) go to 16
c
c     specify governing equations [ n1+1 < j < nj ]
c     composite cathode
      sum=0.0d0
      h=h2
c
      dc=(xx(1,j+1) - xx(1,j-1))/2.0d0/h
      d2c=(xx(1,j+1)-2.0d0*xx(1,j)+xx(1,j-1))/h/h
      a(1,1)=-df(j)/2.0d0/h/h + dc*ddf(j)/h/2.0d0 -
```

```
     1dtm(j)*xx(4,j)/fc/4.0d0/h
      b(1,1)=ep/rr + df(j)/h/h - d2df(j)*dc*dc/2.0d0
     1-d2c*ddf(j)/2.0d0 + xx(4,j)*d2tm(j)*dc/fc/2.0d0
     1+area*dtm(j)*xx(5,j)/2.0d0
      d(1,1)=-df(j)/2.0d0/h/h - dc*ddf(j)/h/2.0d0 +
     1dtm(j)*xx(4,j)/fc/4.0d0/h
      b(1,4)=dtm(j)*dc/2.0d0/fc
      b(1,5)=-area/2.0d0 + area*tm(j)/2.0d0
      g(1)=ep*xt(1,j,kk-1)/rr - dc*dc*d2df(j)*xx(1,j)/2.0d0
     1- dc*dc*ddf(j)/2.0d0 - d2c*ddf(j)*xx(1,j)/2.0d0 + term(j)/2.0d0+
     1xx(4,j)*d2tm(j)*dc*xx(1,j)/fc/2.0d0 + dtm(j)*xx(4,j)*dc/fc/2.0d0
     1+dtm(j)*area*xx(5,j)*xx(1,j)/2.0d0
c
      dcf=(xx(1,j+1)-xx(1,j))/h
      r1=(xx(1,j+1)+xx(1,j))/2.0d0
      r4=(xx(4,j+1)+xx(4,j))/2.0d0
      p1=(tm(j)+tm(j+1))/2.0d0
      p2=(cd(j)+cd(j+1))/2.0d0
      p3=(dtm(j)+dtm(j+1))/2.0d0
      p4=(dfu(j)+dfu(j+1))/2.0d0
      p5=(dcd(j)+dcd(j+1))/2.0d0
      p6=(d2fu(j)+d2fu(j+1))/2.0d0
      d(2,1)=(1.0d0-p1)*(1.0d0/r1+p4)/h
      b(2,1)=-d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
     1      -(1.0d0/r1+p4)*dcf*p3+fc/r/t*r4*p5/p2/p2)/2.0d0
      d(2,1)=d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
     1      -(1.0d0/r1+p4)*dcf*p3+fc/r/t*r4*p5/p2/p2)/2.0d0
      d(2,2)=fc/r/t/h
      b(2,2)=-fc/r/t/h
      b(2,4)=-fc/r/t*(1.0d0/p2+1.0d0/sig)/2.0d0
      d(2,4)=-fc/r/t*(1.0d0/p2+1.0d0/sig)/2.0d0
      g(2)=-fc/r/t*cur/sig + (1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf*r1 -
     1    (1.0d0/r1+p4)*dcf*p3*r1+fc/r/t*r4*p5*r1/p2/p2
c
      call ekin(j)
c
      if (kk .gt. 2) then
      sum=0.0d0
      do i=1, kk-2
      sum=sum + (xt(3,j,i+1)-xt(3,j,i))*ai(kk-i)/(ts(i+1)-ts(i))
      end do
      else sum=0.0d0
      end if
      b(4,3)=ai(1)/rr
      b(4,5)=1.0d0/Rad
      g(4)=ai(1)*xt(3,j,kk-1)/rr -sum*dfs/Rad/Rad
c
      b(5,4)=-1.0d0/h
      a(5,4)=1.0d0/h
      b(5,5)=area*fc/2.0d0
      a(5,5)=area*fc/2.0d0
c
      call band(j)
      go to 10
```

```
c
c     ____________________________________________________
c
   16 continue
c     specify boundary conditions at right interface(j=nj)
c
      sum=0.0d0
c     the "irrefutable" boundary conditions:
      b(1,1)=1.5d0/h
      a(1,1)=-2.0d0/h
      y(1,1)=0.5d0/h
      g(1)=0.0d0
c
      b(2,2)=1.5d0/h
      a(2,2)=-2.0d0/h
      y(2,2)=0.5d0/h
      g(2)=-cur/sig
c
      call ekin(j)
c
      if (kk .gt. 2) then
      sum=0.0d0
      do i=1, kk-2
      sum=sum + (xt(3,j,i+1)-xt(3,j,i))*ai(kk-i)/(ts(i+1)-ts(i))
      end do
      else sum=0.0d0
      end if
      b(4,3)=ai(1)/rr
      b(4,5)=1.0d0/Rad
      g(4)=ai(1)*xt(3,j,kk-1)/rr -sum*dfs/Rad/Rad
c
      b(5,4)=1.5d0/h
      a(5,4)=-2.0d0/h
      y(5,4)=0.5d0/h
      b(5,5)=-area*fc
c
      call band(j)
c     ____________________________________________________
c
c     begin check for convergence
c
      do i=1,n
      xp(i)=(4.0d0*c(i,2)-3.0d0*c(i,1)-c(i,3))/2.0d0/h
      end do
      do 25 j=1,nj
      if(c(1,j).lt.xx(1,j)/1.0d01) c(1,j)=xx(1,j)/1.0d01
c     if(cmax-c(1,j).le.0.0d0) c(1,j)=0.989d0*cmax+xx(1,j)/1.0d02
      if(c(2,j).lt.xx(2,j)-0.3d0) c(2,j)=xx(2,j)-0.3d00
      if(c(2,j).gt.xx(2,j)+0.3d0) c(2,j)=xx(2,j)+0.3d00
c     if(c(1,j).lt.1.0d-10 ) c(1,j)=1.0d-10
c     if(c(1,j).lt.1.0d-04 ) c(5,j)=0.0d0
      if(c(3,j).lt.xx(3,j)/1.0d01) c(3,j)=xx(3,j)/1.0d01
      if(ct-c(3,j).le.0.0d0) c(3,j)=0.989d0*ct+xx(3,j)/1.0d02
      if(c(3,j).lt.1.0d-12 .and. j .ge. n1) c(3,j)=1.0d-12
```

```
      do 25 i=1,n
 25   xx(i,j)=c(i,j)
c
      if (jcount .gt. lim ) then
      tau=tau/2.0d0
      rr=rr/2.0d0
      ts(kk)=ts(kk-1)+tau
*       print*,' time step reduced to ', rr, ts(kk)*Rad*Rad/dfs
      if (tau .lt. 1.0d-02) then
c      print*,kk-1, ' this time step did not converge'
      print99
      call nucamb (n,2,30)
      stop
      else
      call calca(kk)
      go to 666
      end if
c
      else
         do 55 ii=1,n
             dxp=dabs( xp(ii)-xp0(ii) )
             dxx=dabs( xx(ii,n1+10)-xx0(ii,n1+10) )
           if ( dxx .gt. 1.0d-09*dabs( xx(ii,n1+10) ) )  go to 8
           if ( dxp .gt. 1.0d-07*dabs( xp(ii)) ) go to 8
c
  55     continue
c
c      print*,jcount,' iterations required'
c      Increasing time steps:
      if (jcount .lt. 6 .and. kk .gt. 5) then
      if (rr.lt.tsmax) then
      tau=tau*2.0d0
      rr=rr*2.0d0
c      ts(kk)=ts(kk-1)+tau
c       print*,'time step increased to ',rr, ts(kk)*Rad*Rad/dfs
c      call calca(kk)
c      go to 666
      end if
      end if
c
c- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
      do j=2,nj-1
c
      if (j .eq. n1) then
      dc=(xx(1,j)-xx(1,j-1))/h1
      dc2=(xx(1,j+1)-xx(1,j))/h2
      r1=(xx(4,j)+xx(4,j-1))/2.0d0
      r2=(xx(4,j+1)+xx(4,j))/2.0d0
      r3=(xx(1,j)+xx(1,j-1))/2.0d0
      r4=(xx(1,j)+xx(1,j+1))/2.0d0
      p1=(df(j-1)+df(j))/2.0d0
      p2=((ep**1.5)*df(j)/(eps**1.5) + df(j+1))/2.0d0
      p3=(dtm(j)+dtm(j+1))/2.0d0
      p4=(dtm(j-1)+dtm(j))/2.0d0
```

```
      p5=(tm(j+1)+tm(j))/2.0d0
      p6=(tm(j)+tm(j-1))/2.0d0
      p7=((ep**1.5)*ddf(j)/(eps**1.5)+ddf(j+1))/2.0d0
      p8=(ddf(j)+ddf(j-1))/2.0d0
      term(j)=area*(3.0d0*xx(5,j)/8.0d0 + xx(5,j+1)/8.0d0) +
     1cur*p6/fc/h2-p5*r2/h2/fc-p1*dc/h2+p2*dc2/h2+p8*dc*r3/h1
     1+p3*r2*r4/fc/h2-cur*r3*p4/fc/h2-p7*dc2*r4/h2
c
      else
c
      if(j .gt. n1) h=h2
      if(j .lt. n1) h=h1
      dc=(xx(1,j+1)-xx(1,j-1))/2.0d0/h
      d2c=(xx(1,j+1)-2.0d0*xx(1,j)+xx(1,j-1))/h/h
      term(j)=df(j)*d2c+ dc*dc*ddf(j) - dc*dtm(j)*xx(4,j)/fc -
     1area*xx(5,j)*(tm(j)-1.0d0)
c
      end if
c
c
      end do
c- - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - - -
c
      end if
c
      return
      end
c
c     ****************************************************************
      subroutine calca(kk)
      implicit double precision*8(a-h,o-z)
      common /n/ nx,nt,n1,n2,nj
      common /calc/ ai(1600),u(222,1600),ts(1600),h,h1,h2,h3,rr
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
      dimension ar(2,1600)
c
      s=1.644934066848d0
c
      do i=1,kk-1
      ar(1,i)=ts(kk)-ts(i)
      ar(2,i)=ts(kk)-ts(i+1)
      do m=1,2
      t1=ar(m,i)
      a1=0.0d0
c
      if(t1 .gt. 0.06d0) then
c
      do j=1,5
      y1=j*j*pi*pi*t1
      if (y1 .gt. 1.5d02) then
      da=0.0d0
      else
      da=(dexp(-y1))/j/j
```

```
c     da=(dexp(-j*j*pi*pi*t1))/j/j
      end if
      a1=a1+da
      end do
      a1=2.0d0*(s-a1)/pi/pi
c
      else
c
      if(t1 .eq. 0.0d0) then
      a1=0.0d0
      else
      do j=1,3
      z=j/dsqrt(t1)
      call erfc(z,e)
      y2=j*j/t1
      if(y2 .gt. 1.5d02) then
      da=-j*(dsqrt(pi/t1))*e
      else
      da=dexp(-y2)-j*dsqrt(pi/t1)*e
      end if
      a1=a1+da
      end do
      a1=-t1 + 2.0d0*dsqrt(t1/pi)*(1.0d0+2.0d0*a1)
      end if
c
      end if
      ar(m,i)=a1
      end do
c
      ai(kk-i)=ar(1,i)-ar(2,i)
      end do
c
      return
      end
c     ******************************************
      subroutine erfc(z,e)
      implicit double precision*8(a-h,o-z)
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
c
      a1=0.254829592d0
      a2=-0.284496736d0
      a3=1.421413741d0
      a4=-1.453152027d0
      a5=1.061405429d0
      if(z .lt. 2.747192d0) then
      t2=1.0d0/(1.0d0+0.3275911d0*z)
      e=(a1*t2+a2*t2*t2+a3*t2**3.0d0+a4*t2**4.0d0
     1+a5*t2**5.0d0)*dexp(-z*z)
      else
c
      if (z .gt. 25.0d0) then
      e=0.0d0
      else
```

```
c
      sum=0.0d0
      max=z*z + 0.5
      fac=-0.5d0/z/z
      sum=fac
      tl=fac
      n=1
   10 n=n+1
      if(n .gt. max) go to 15
      tn=tl*(2.0d0*n-1.0d0)*fac
      sum=sum + tn
      if(tn .lt. 1.0d-06) go to 15
      tl=tn
      go to 10
   15 e=(dexp(-z*z))*(1.0d0+sum)/dsqrt(pi)/z
      end if
      end if
c
      return
      end
c*****************************************************
      subroutine band(j)
      implicit double precision*8(a-h,o-z)
      common /n/ nx,nt,n1,n2,nj
      common/mat/ b,d
      common/bnd/ a,c,g,x,y
      dimension b(10,10),d(10,21)
      dimension a(10,10),c(10,221),g(10),x(10,10),y(10,10)
      dimension e(10,11,221)
  101 format (15h determ=0 at j=,i4)
      n=nx
      if (j-2)  1,6,8
    1 np1= n + 1
      do 2 i=1,n
      d(i,2*n+1)= g(i)
      do 2 l=1,n
      lpn= l + n
    2 d(i,lpn)= x(i,l)
      call matinv(n,2*n+1,determ)
      if (determ)  4,3,4
    3 print 101, j
    4 do 5 k=1,n
      e(k,np1,1)= d(k,2*n+1)
      do 5 l=1,n
      e(k,l,1)= - d(k,l)
      lpn= l + n
    5 x(k,l)= - d(k,lpn)
      return
    6 do 7 i=1,n
      do 7 k=1,n
      do 7 l=1,n
    7 d(i,k)= d(i,k) + a(i,l)*x(l,k)
    8 if (j-nj)  11,9,9
    9 do 10 i=1,n
```

```
      do 10 l=1,n
      g(i)= g(i) - y(i,l)*e(l,np1,j-2)
      do 10 m=1,n
   10 a(i,l)= a(i,l) + y(i,m)*e(m,l,j-2)
   11 do 12 i=1,n
      d(i,np1)= - g(i)
      do 12 l=1,n
      d(i,np1)= d(i,np1) + a(i,l)*e(l,np1,j-1)
      do 12 k=1,n
   12 b(i,k)= b(i,k) + a(i,l)*e(l,k,j-1)
      call matinv(n,np1,determ)
      if (determ)  14,13,14
   13 print 101, j
   14 do 15 k=1,n
      do 15 m=1,np1
   15 e(k,m,j)= - d(k,m)
      if (j-nj)  20,16,16
   16 do 17 k=1,n
   17 c(k,j)= e(k,np1,j)
      do 18 jj=2,nj
      m= nj - jj + 1
      do 18 k=1,n
      c(k,m)= e(k,np1,m)
      do 18 l=1,n
   18 c(k,m)= c(k,m) + e(k,l,m)*c(l,m+1)
      do 19 l=1,n
      do 19 k=1,n
   19 c(k,1)= c(k,1) + x(k,l)*c(l,3)
   20 return
      end
c*************************************************
      subroutine matinv(n,m,determ)
      implicit double precision*8(a-h,o-z)
      common/mat/ b,d
      dimension b(10,10),d(10,21)
      dimension id(10)
      determ=1.0
      do 1 i=1,n
    1 id(i)=0
      do 18 nn=1,n
      bmax=1.1
      do 6 i=1,n
      if(id(i).ne.0) go to 6
      bnext=0.0
      btry=0.0
      do 5 j=1,n
      if(id(j).ne.0) go to 5
      if(dabs(b(i,j)).le.bnext) go to 5
      bnext=dabs(b(i,j))
      if(bnext.le.btry) go to 5
      bnext=btry
      btry=dabs(b(i,j))
      jc=j
    5 continue
```

```
      if(bnext.ge.bmax*btry) go to 6
      bmax=bnext/btry
      irow=i
      jcol=jc
    6 continue
      if(id(jc).eq.0) go to 8
      determ=0.0
      return
    8 id(jcol)=1
      if(jcol.eq.irow) go to 12
      do 10 j=1,n
      save=b(irow,j)
      b(irow,j)=b(jcol,j)
   10 b(jcol,j)=save
      do 11 k=1,m
      save=d(irow,k)
      d(irow,k)=d(jcol,k)
   11 d(jcol,k)=save
   12 f=1.0/b(jcol,jcol)
      do 13 j=1,n
   13 b(jcol,j)=b(jcol,j)*f
      do 14 k=1,m
   14 d(jcol,k)=d(jcol,k)*f
      do 18 i=1,n
      if(i.eq.jcol) go to 18
      f=b(i,jcol)
      do 16 j=1,n
   16 b(i,j)=b(i,j)-f*b(jcol,j)
      do 17 k=1,m
   17 d(i,k)=d(i,k)-f*d(jcol,k)
   18 continue
      return
      end
c*********************************************************
      subroutine nucamb(n,il2,il3)
      implicit double precision*8(a-h,o-z)
      common /n/ nx,nt,n1,n2,nj
      common /calc/ ai(1600),u(222,1600),ts(1600),h,h1,h2,h3,rr
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/var/ xp(10),xx(5,221),xi(5,221),xt(5,221,1600)
      common/cprop/ sig,area,ct,dfs,Rad,cmax,rka,rkf
      common/tprop/df(221),cd(221),tm(221),
      1ddf(221),dcd(221),dtm(221),dfu(221),d2df(221),
      1d2tm(221),d2fu(221)
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
      dimension zz(221)
c
  109 format(f7.1,', ',f7.1,', ',f7.4,', 'g12.4,', ',g12.4,', ',g12.4)
  309 format(f8.5,', ',f8.5)
   44 format('t = ',f7.2,' min')
c
      do 5 i=1,n1
      w=i-1
    5 zz(i) = w*h1*1.0d06
```

```
      do 7 i=n1+1,nj
      w=i-n1
    7 zz(i) = zz(n1)+w*h2*1.0d06
c
      do 10 l=2,nt+1,il3
       print*,' '
      print44,ts(l)*Rad*Rad/dfs/60.0d0
      print*,'distance  concen  potential   c solid',
     1'    current            j'
      print*,'microns  (mol/m3)  (V)     (mol/m3)',
     1'    (A/m2)        (mol/m2-s)'
      do 10 j=1,nj,il2
c 10  print309,zz(j),u(j,l)
   10 print109,zz(j),xt(1,j,l),xt(2,j,l),xt(3,j,l),xt(4,j,l),xt(5,j,l)
c
      return
      end
c     ***************************************************
      subroutine guess(n)
      implicit double precision*8(a-h,o-z)
      common /n/ nx,nt,n1,n2,nj
      common /calc/ ai(1600),u(222,1600),ts(1600),h,h1,h2,h3,rr
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/var/ xp(10),xx(5,221),xi(5,221),xt(5,221,1600)
      common/cprop/ sig,area,ct,dfs,Rad,cmax,rka,rkf
      common/tprop/df(221),cd(221),tm(221),
     1ddf(221),dcd(221),dtm(221),dfu(221),d2df(221),
     1d2tm(221),d2fu(221)
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
      dimension del(5)
c
      del(1)=(xi(1,nj)-xi(1,1))/(nj-1)
      del(2)=cur*h2/2.0d-02
      del(4)=cur/(n2)
      del(5)=(xi(5,nj))/(nj-1)
      xi(4,1)=cur
c
      do i=1,n1-1
      xi(3,i)=0.0d0
      xi(4,i)=cur
      xi(5,i)=0.0d0
      end do
      do i=n1,nj
      xi(3,i)=xi(3,n1)
      xi(4,i)=xi(4,1)-del(4)*(i-n1)
      xi(5,i)=xi(5,nj)
      end do
c
      do 15 i=1,nj
      xt(5,i,1)=xi(5,i)
      xt(4,i,1)=xi(4,i)
      xt(3,i,1)=xi(3,i)
c     xi(2,i)=xi(2,1)+del(2)*(i-1)
      xt(2,i,1)=xi(2,1)
```

```
      xi(1,i)=xi(1,1)+del(1)*(i-1)
   15 xt(1,i,1)=xi(1,i)
c
      return
      end
c*********************************************************
      subroutine util
      implicit double precision*8(a-h,o-z)
      common /n/ nx,nt,n1,n2,nj
      common /calc/ ai(1600),u(222,1600),ts(1600),h,h1,h2,h3,rr
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/var/ xp(10),xx(5,221),xi(5,221),xt(5,221,1600)
      common/cprop/ sig,area,ct,dfs,Rad,cmax,rka,rkf
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
c
      do 10 j=n1,nj
      u(j,1)=0.0d0
        do k=2,nt+1
         u(j,k)=u(j,k-1)-1.5d0*(xt(5,j,k)+xt(5,j,k-1))*rr/Rad
        end do
      do k=2,nt+1
      u(j,k)=(u(j,k)+xt(3,j,1))/ct
      end do
      u(j,1)=xt(3,j,1)/ct
   10 continue
      do k=1,nt+1
      u(nj+1,k)=0.0d0
        do j=n1+1,nj-1
         u(nj+1,k)=u(nj+1,k)+u(j,k)
        end do
         u(nj+1,k)=u(nj+1,k)+0.5d0*(u(n1,k)+u(nj,k))
      u(nj+1,k)=u(nj+1,k)/(nj-n1)
      end do
c
      return
      end
c*************************************************************
      subroutine peak(n,lim,il4)
      implicit double precision*8(a-h,o-z)
      common /n/ nx,nt,n1,n2,nj
      common /calc/ ai(1600),u(222,1600),ts(1600),h,h1,h2,h3,rr
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/var/ xp(10),xx(5,221),xi(5,221),xt(5,221,1600)
      common/cprop/ sig,area,ct,dfs,Rad,cmax,rka,rkf
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
c
c     Peak power current ramp section:
      print*,' '
      print*,'    PEAK POWER '
      print*,' '
      print*,'cell pot  ',' material',' current','   power'
      print*,'  (V)  ','    balance  ',' (A/m2)','   (W/m2)'
c
c     Duration of current pulse is 30 seconds.
```

```
c
      li=0
      k=nt+2
      rr=30.0d0
      dt=dfs*rr/Rad/Rad
      ts(k)=ts(k-1)+rr*dfs/Rad/Rad
      call calca(k)
c     Ramp current:
c
      do ii=1,115
      cur=ii*1.0d0+40.0d0
      call zts(n,lim,k-1,dt,k-1)
      rr=30.0d0
      dt=dfs*rr/Rad/Rad
      ts(k)=ts(k-1)+rr*dfs/Rad/Rad
      call calca(k)
      if(ii .eq. 1) then
      call comp(n,lim,k,dt,1)
      else
      call comp(n,lim,k,dt,0)
      end if
c
      do 40 i=1,n
      do 40 j=1,nj
   40 xt(i,j,k)=xx(i,j)
c
      call cellpot(k,il4,vv,0)
c
      end do
c     do ik=1,nj,2
c     print*,xt(1,ik,k),xt(3,ik,k)
c     end do
c
      return
      end
c*********************************************************
      subroutine cellpot(kk,il4,v,li)
      implicit double precision*8(a-h,o-z)
      common /n/ nx,nt,n1,n2,nj
      common /calc/ ai(1600),u(222,1600),ts(1600),h,h1,h2,h3,rr
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/power/ ed,Vold
      common/var/ xp(10),xx(5,221),xi(5,221),xt(5,221,1600)
      common/cprop/ sig,area,ct,dfs,Rad,cmax,rka,rkf
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
c
  309 format(f8.5,', ',f8.5,', ',f7.3,', ',f12.3,',     ',i3,',',f12.4
     +   ,',',f10.5,',',f12.5,',',f10.5)
  319 format(f8.5,', ',f7.3,', ',f7.2,',     ',f7.2,', ',f7.2)
  320 format(f12.3,f9.4)

      lim=20
c
      if (il4 .eq. 1 ) then
```

```
c     For polymer PEO elyte:
      vr=cmax-xt(1,1,kk)
      if (vr .lt. 1.0d0) vr=1.0d0
      x0=rkf*(dsqrt((vr)*xt(1,1,kk)))
c      x0=7.375d-03*(dsqrt((cmax-xt(1,1,kk))*xt(1,1,kk)))
c
      else
c
c     For liquid PC elyte:
      x0=rkf*(dsqrt(xt(1,1,kk)))
c      x0=3.0d-01*(dsqrt(xt(1,1,kk)))
      end if
c
      p20=-0.01d0
      p2=p20
      jcount=0
c
    8 jcount = jcount +1
      p20=p2
      a1=0.33d0*fc/r/t
      c1=0.67d0*fc/r/t
c
      p2=-cur/x0+(1.0d0+p20*a1)*dexp(-p20*a1)
     1-(1.0d0-p20*c1)*dexp(p20*c1)
      p2=p2/(a1*dexp(-a1*p20)+c1*dexp(c1*p20))
c
      if(jcount .gt. lim) then
      print*,jcount
      stop
      else
      dx=dabs(p2-p20)
      if(dx .gt. dabs(1.0d-09*p2)) go to 8
      end if
c
      sum=0.0d0
      do j=n1+1,nj-1
      sum=sum+xt(4,j,kk)*h
      end do
      sum=sum-cur*h*(nj-n1)+0.5d0*h*(xt(4,1,kk)+xt(4,nj,kk))
      v=p2+xt(2,1,kk)+sum/sig
c
c     Material balance criteria:
      sum=0.0d0
      do j=2,n1-1
      sum=sum+xt(1,j,kk)*h1*eps
      end do
      sum=sum+(xt(1,1,kk)+xt(1,n1,kk))*h1*eps/2.0d0
      do j=n1+1,nj-1
      sum=sum+xt(1,j,kk)*ep*h2
      end do
      sum=sum+(xt(1,n1,kk)+xt(1,nj,kk))*h2*ep/2.0d0
      w=xt(1,1,1)*((n1-1)*h1*eps+n2*ep*h2)
      ca=sum/w
c
```

```
c     ut=cur*(kk-1)*rr/fc/(1.0d0-ep)/(nj-n1)/h/ct+xt(3,n1,1)/ct
      th=ts(kk)*Rad*Rad/dfs/6.0d01
c     Volume fraction filler is epf:
      if(kk .eq. 2) ut=xt(3,n1,1)/ct
      r4=(ts(kk)-ts(kk-1))*Rad*Rad/dfs
      ut=ut + cur*r4/fc/(1.0d0-ep-epf)/(nj-n1)/h/ct

      call temperature(kk,v,q,ut,Uoc)
      tprint=t-273.15
c      print309,ut,v,ca,th,t-273.15,uo,htc,kk-1
      if(li .eq. 1) print309,ut,v,ca,th,kk-1,tprint,Uoc,htc,q
c      if(li.eq.1) print320, th, q
      if(li .eq. 0) print319,v,ca,cur,cur*v,th
c     x1=(kk-1)*rr/3.6d03
      if(kk .eq. 2) ed=0.0d0
      ed=ed+(Vold+v)*(ts(kk)-ts(kk-1))*cur*Rad*Rad/dfs/2.0d0
      Vold=v
      return
      end
c*************************************************************
      subroutine sol(nmax,jj)
      implicit double precision*8(a-h,o-z)
      common /calc/ ai(1600),u(222,1600),ts(1600),h,h1,h2,h3,rr
      common/var/ xp(10),xx(5,221),xi(5,221),xt(5,221,1600)
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/cprop/ sig,area,ct,dfs,Rad,cmax,rka,rkf
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht

      dimension cs(50)
c
c     set initial value of solid concentration
      do i=1, 50
c     cs(i)=0.0d0
      cs(i)=xt(3,jj,1)
      end do
c
c     complete calculations for 50 points along radius of particle
      do 10 i=1,50
      y2=2.0d-02*i
c
c
      sum1=0.0d0
      do 20 kk=1,nmax
      k=nmax+1-kk
c
      t1=(ts(nmax+1)-ts(k))
c     print*,t1
      sum2=sum1
c
c     calculate c bar (r,t1)
c
      sum1=0.0d0
      r1=1.0d0
c
```

```
      do j=1,15
      r1=-r1
      y1=j*j*pi*pi*t1
      y3=j*pi*y2
      if (y1 .gt. 1.50d02) then
      da=0.0d0
      else
      da=dexp(-y1)
      end if
      sum1=sum1-2.0d0*r1*da*dsin(y3)/j/pi/y2
      end do
      sum1=1.0d0-sum1
c
c     perform superposition
c
      cs(i)=cs(i)+(xt(3,jj,k+1)+xt(3,jj,k)-2.0d0*xt(3,jj,1)
     1)*(sum1-sum2)/2.0d0
   20 continue
c
   10 continue
c
      print*,' '
      print*,ts(nmax)
      print*,' '
      do i=1, 50, 1
      print*,.02*i,' ',cs(i)
c     print*,' '
      end do
      return
      end
c*************************************************************
      subroutine mass(tw,re,rs,rf,rc,rp,ef)
      implicit double precision*8(a-h,o-z)
      common /n/ nx,nt,n1,n2,nj
      common /calc/ ai(1600),u(222,1600),ts(1600),h,h1,h2,h3,rr
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/var/ xp(10),xx(5,221),xi(5,221),xt(5,221,1600)
      common/cprop/ sig,area,ct,dfs,Rad,cmax,rka,rkf
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
  309 format(f8.5,', ',f8.5,', ',f8.5,', ',f8.5)
c
c     mass of positive electrode
      c1=h2*n2*(re*ep+rs*(1.0d0-ep-epf)+rf*epf)
c
c     mass of separator
      s=(re*eps+rc*(1.0d0-eps))*h1*(n1-1)
c
c     mass of negative electrode
      a1=ef*(1.0d0-ep-epf)*(ct)*h2*(n2)*6.941/1.0d03

c     mass of bipolar partition is rp*h3

      tw=c1+s+a1+rp*h3
      print309,a1,s,c1,rp*h3,tw
```

```
c
      return
      end
c*****************************************************
      subroutine zts(n,lim,kk,tau,li)
      implicit double precision*8(a-h,o-z)
      common /n/ nx,nt,n1,n2,nj
      common /calc/ ai(1600),u(222,1600),ts(1600),h,h1,h2,h3,rr
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/ssblock/ xp0(5),xx0(5,221),term(221)
      common/var/ xp(10),xx(5,221),xi(5,221),xt(5,221,1600)
      common/cprop/ sig,area,ct,dfs,Rad,cmax,rka,rkf
      common/tprop/df(221),cd(221),tm(221),
     1ddf(221),dcd(221),dtm(221),dfu(221),d2df(221),
     1d2tm(221),d2fu(221)
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
      common/mat/ b,d
      common/bnd/ a,c,g,x,y
      dimension b(10,10),d(10,21)
      dimension a(10,10),c(10,221),g(10),x(10,10),y(10,10)
c
   99 format (1h ,//5x,'this run did not converge'//)
      nx=n
c
      do 1 j=1,nj
      do 1 i=1,n
         c(i,j)=xt(i,j,kk-1)
    1    xx(i,j)=xt(i,j,kk-1)
c     sets first guess to last time step values
      jcount=0
      do 4 i=1,n
    4    xp(i)=0.0d0
c     initialize variables to begin each iteration (jcount is iteration #)
    8 j=0
      jcount=jcount+1
      call prop(nj,n1)
c     obtains physical properties at this specific point
      do 9 i=1,n
      do 9 k=1,n
         x(i,k)=0.0d0
    9    y(i,k)=0.0d0
c
c     store previous iteration of  (xp in xp0)  &  (xx in xx0)
      do 6 i=1,n
         xp0(i)=xp(i)
    6    xx0(i,n1+10)=xx(i,n1+10)
         xx0(n,n1+10)=xx(n,n1+10)
c
c     for a given iteration, set up governing equations and bc's
c     start at the left interface and move across polymer
c
   10 j=j+1
c
      do 11 i=1,n
```

```
      g(i)=0.0d0
      xx(i,j)=c(i,j)
   do 11 k=1,n
      a(i,k)=0.0d0
      b(i,k)=0.0d0
 11     d(i,k)=0.0d0
c  clears all arrays before use
   if(j.ne.1) go to 13
c
c  ____________________________________________________
c
c  specify boundary conditions at left interface (j=1)
c  boundary conditions at negative electrode
   h=h1
c
   b(1,1)=1.0d0
   g(1)=xt(1,j,kk-1)
c
   dcf=(xx(1,j+1)-xx(1,j))/h
   r1=(xx(1,j+1)+xx(1,j))/2.0d0
   p1=(tm(j)+tm(j+1))/2.0d0
   p2=(cd(j)+cd(j+1))/2.0d0
   p3=(dtm(j)+dtm(j+1))/2.0d0
   p4=(dfu(j)+dfu(j+1))/2.0d0
   p6=(d2fu(j)+d2fu(j+1))/2.0d0
   d(2,1)=(1.0d0-p1)*(1.0d0/r1+p4)/h
   b(2,1)=-d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
  1     -(1.0d0/r1+p4)*dcf*p3)/2.0d0
   d(2,1)=d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
  1     -(1.0d0/r1+p4)*dcf*p3)/2.0d0
   d(2,2)=fc/r/t/h
   b(2,2)=-fc/r/t/h
   g(2)=fc/r/t*cur/p2+(1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf*r1 -
  1    (1.0d0/r1+p4)*dcf*p3*r1
c
   b(3,5)=1.0d0
   g(3)=0.0d0
   b(4,3)=1.0d0
   g(4)=0.0d0
   b(5,4)=1.0d0
   g(5)=cur
c
   call band(j)
   go to 10
c
c  ____________________________________________________
c  specify governing equations in polymer separator
 13 if (j .ge. n1) go to 110
c
   b(1,1)=1.0d0
   g(1)=xt(1,j,kk-1)
c
   dcf=(xx(1,j+1)-xx(1,j))/h
   r1=(xx(1,j+1)+xx(1,j))/2.0d0
```

```
      p1=(tm(j)+tm(j+1))/2.0d0
      p2=(cd(j)+cd(j+1))/2.0d0
      p3=(dtm(j)+dtm(j+1))/2.0d0
      p4=(dfu(j)+dfu(j+1))/2.0d0
      p5=(dcd(j)+dcd(j+1))/2.0d0
      p6=(d2fu(j)+d2fu(j+1))/2.0d0
      d(2,1)=(1.0d0-p1)*(1.0d0/r1+p4)/h
      b(2,1)=-d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
     1      -(1.0d0/r1+p4)*dcf*p3)/2.0d0
      d(2,1)=d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
     1      -(1.0d0/r1+p4)*dcf*p3)/2.0d0
      d(2,2)=fc/r/t/h
      b(2,2)=-fc/r/t/h
      g(2)=fc/r/t*cur/p2+(1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf*r1 -
     1    (1.0d0/r1+p4)*dcf*p3*r1
c
      b(3,5)=1.0d0
      g(3)=0.0d0
      b(4,3)=1.0d0
      g(4)=0.0d0
      b(5,4)=1.0d0
      g(5)=cur
      call band(j)
      go to 10
c
c     Now for the boundary between cathode and separator(j=n1):
  110 if (j .ne. n1) go to 120
c
      b(1,1)=1.0d0
      g(1)=xt(1,j,kk-1)
c
      dcf=(xx(1,j+1)-xx(1,j))/h
      r1=(xx(1,j+1)+xx(1,j))/2.0d0
      r4=(xx(4,j+1)+xx(4,j))/2.0d0
      p1=(tm(j)+tm(j+1))/2.0d0
      p2=((ep**1.50d0)*cd(j)/(eps**1.5d0)+cd(j+1))/2.0d0
      p3=(dtm(j)+dtm(j+1))/2.0d0
      p4=(dfu(j)+dfu(j+1))/2.0d0
      p5=((ep**1.50d0)*dcd(j)/(eps**1.5d0)+dcd(j+1))/2.0d0
      p6=(d2fu(j)+d2fu(j+1))/2.0d0
      d(2,1)=(1.0d0-p1)*(1.0d0/r1+p4)/h
      b(2,1)=-d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
     1      -(1.0d0/r1+p4)*dcf*p3+fc/r/t*r4*p5/p2/p2)/2.0d0
      d(2,1)=d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
     1      -(1.0d0/r1+p4)*dcf*p3+fc/r/t*r4*p5/p2/p2)/2.0d0
      d(2,2)=fc/r/t/h
      b(2,2)=-fc/r/t/h
      b(2,4)=-fc/r/t*(1.0d0/p2+1.0d0/sig)/2.0d0
      d(2,4)=b(2,4)
      g(2)=-fc/r/t*cur/sig + (1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf*r1
     1    -(1.0d0/r1+p4)*dcf*p3*r1+fc/r/t*r4*p5*r1/p2/p2
c
      call ekin(j)
c
```

```
      b(4,3)=1.0d0
      g(4)=xt(3,j,kk-1)
c
      b(5,4)=1.0d0
      g(5)=cur
c
      call band(j)
      go to 10
c
 120 if (j .eq. nj) go to 16
c
c     specify governing equations [ n1+1 < j < nj ]
c     composite cathode
      sum=0.0d0
      h=h2
c
      b(1,1)=1.0d0
      g(1)=xt(1,j,kk-1)
c
      dcf=(xx(1,j+1)-xx(1,j))/h
      r1=(xx(1,j+1)+xx(1,j))/2.0d0
      r4=(xx(4,j+1)+xx(4,j))/2.0d0
      p1=(tm(j)+tm(j+1))/2.0d0
      p2=(cd(j)+cd(j+1))/2.0d0
      p3=(dtm(j)+dtm(j+1))/2.0d0
      p4=(dfu(j)+dfu(j+1))/2.0d0
      p5=(dcd(j)+dcd(j+1))/2.0d0
      p6=(d2fu(j)+d2fu(j+1))/2.0d0
      d(2,1)=(1.0d0-p1)*(1.0d0/r1+p4)/h
      b(2,1)=-d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
     1      -(1.0d0/r1+p4)*dcf*p3+fc/r/t*r4*p5/p2/p2)/2.0d0
      d(2,1)=d(2,1)+((1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf
     1      -(1.0d0/r1+p4)*dcf*p3+fc/r/t*r4*p5/p2/p2)/2.0d0
      d(2,2)=fc/r/t/h
      b(2,2)=-fc/r/t/h
      b(2,4)=-fc/r/t*(1.0d0/p2+1.0d0/sig)/2.0d0
      d(2,4)=-fc/r/t*(1.0d0/p2+1.0d0/sig)/2.0d0
      g(2)=-fc/r/t*cur/sig + (1.0d0-p1)*(p6-1.0d0/r1/r1)*dcf*r1 -
     1    (1.0d0/r1+p4)*dcf*p3*r1+fc/r/t*r4*p5*r1/p2/p2
c
      call ekin(j)
c
      b(4,3)=1.0d0
      g(4)=xt(3,j,kk-1)
c
      b(5,4)=-1.0d0/h
      a(5,4)=1.0d0/h
      b(5,5)=area*fc/2.0d0
      a(5,5)=area*fc/2.0d0
c
      call band(j)
      go to 10
c
c
```

```
c
  16 continue
c    specify boundary conditions at right interface(j=nj)
c
     sum=0.0d0
c    the "irrefutable" boundary conditions:
     b(1,1)=1.0d0
     g(1)=xt(1,j,kk-1)
c
     b(2,2)=1.5d0/h
     a(2,2)=-2.0d0/h
     y(2,2)=0.5d0/h
     g(2)=-cur/sig
c
     call ekin(j)
c
     b(4,3)=1.0d0
     g(4)=xt(3,j,kk-1)
c
     b(5,4)=1.5d0/h
     a(5,4)=-2.0d0/h
     y(5,4)=0.5d0/h
     b(5,5)=-area*fc
c
     call band(j)
c    ------------------------------------------------------------
c
c    begin check for convergence
c
     do i=1,n
     xp(i)=(4.0d0*c(i,2)-3.0d0*c(i,1)-c(i,3))/2.0d0/h
     end do
     do 25 j=1,nj
     if(c(2,j).lt.xx(2,j)-0.3d0) c(2,j)=xx(2,j)-0.3d00
     if(c(2,j).gt.xx(2,j)+0.3d0) c(2,j)=xx(2,j)+0.3d00
     do 25 i=1,n
 25   xx(i,j)=c(i,j)
c
     if (jcount .gt. lim ) then
     print99
     stop
c
     else
        do 55 ii=1,n
           dxp=dabs( xp(ii)-xp0(ii) )
           dxx=dabs( xx(ii,n1+10)-xx0(ii,n1+10) )
          if ( dxx .gt. 1.0d-09*dabs( xx(ii,n1+10) ) )  go to 8
          if ( dxp .gt. 1.0d-07*dabs( xp(ii)) ) go to 8
c
  55    continue
c
c    print*,jcount,' iterations required'
c
     do ll=1, nj, 1
```

```
      do lk=1,n
      xt(lk,ll,li)=xx(lk,ll)
      end do
      end do
      end if
c
      return
      end
c
c*****************************************************
c*****************************************************
      subroutine ekin(j)
      implicit double precision*8(a-h,o-z)
      common /n/ nx,nt,n1,n2,nj
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/var/ xp(10),xx(5,221),xi(5,221),xt(5,221,1600)
      common/cprop/ sig,area,ct,dfs,Rad,cmax,rka,rkf
      common/mat/ b,d
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
      common/bnd/ a,c,g,x,y
      dimension b(10,10),d(10,21)
      dimension a(10,10),c(10,221),g(10),x(10,10),y(10,10)
c
c%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
c
c     OPEN-CIRCUIT POTENTIAL FUNCTIONS:
c
c     g0 is the open-circuit potential in terms of the solid
c     concentration, xx(3,j),
c     g1 is the derivative of the open-circuit potential wrt
c     the solid concentration
c
c &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
c
c     TiS2:
c
      delt=-5.58d-04
      zeta=8.1d0
      cT=2.9d04
      g0=2.17+(dlog((cT-xx(3,j))/xx(3,j))+delt*xx(3,j)+zeta)*r*353.15/fc
      g1=(delt-cT/xx(3,j)/(cT-xx(3,j)))*r*353.15/fc

c &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
c
c     Spinel Mn2O4 (lower plateau)
c
c      c1=2.06307d0
c      c2=-0.869705d0
c      c3=8.65375d0
c      c4=0.981258d0
c      a1=c3*(xx(3,j)/ct-c4)
c      a11=0.1d0
c      a12=100.0d0
c      a13=1.0d-02
```

```
c     g0=c1+c2*(dtanh(a1))
c     g0=g0+a11*dexp(-a12*(xx(3,j)/ct-a13))
c     g1=c2*c3/ct/(dcosh(a1))/(dcosh(a1))
c     g1=g1-a11*a12/ct*(dexp(-a12*(xx(3,j)/ct-a13)))
c
c &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
c
c    Spinel Mn2O4 (upper plateau)
c
c     a1=4.06279d0
c     a2=0.0677504d0
c     a3=21.8502d0
c     a4=12.8268d0
c     a5=0.105734d0
c     a6=1.00167d0
c     a7=-0.379571d0
c     a8=1.575994d0
c     a9=0.045d0
c     a10=71.69d0
c     a11=0.01d0
c     a12=200.0d0
c     a13=0.19d0
c     g0=a1+a2*dtanh(-a3*xx(3,j)/cT+a4)-a5*((a6-xx(3,j)/cT)**a7-
c    1a8)-a9*dexp(-a10*((xx(3,j)/cT)**8.0d0))+a11
c    1*dexp(-a12*(xx(3,j)/cT-a13))
c     g1=(1.0d0/cT)*(-a2*a3/dcosh(-a3*xx(3,j)/cT+a4)/dcosh(-a3
c    1*xx(3,j)/cT+a4)+a5*a7*(a6-xx(3,j)/cT)**(-1.0d0+a7)+
c    1a9*a10*8.0d0*((xx(3,j)/cT)**7.0d0)*dexp(-a10*
c    1(xx(3,j)/cT)**8.0d0))-a11*a12/cT*dexp(-a12*(xx(3,j)/cT-a13))
c
c%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
c
c    KINETIC EXPRESSIONS
c
c    h0 is the exchange current density (A/m2)
c    h1 is the derivative of io wrt solid concentration, xx(3,j)
c    h2 is the derivative of io wrt electrolyte concen., xx(1,j)
c
c &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
c
c
c    NONAQUEOUS LIQUIDS
c
c    alpha=0.5d0
c    alphc=0.5d0
c    h0=rka*dsqrt(xx(1,j))*dsqrt(cT-xx(3,j))*dsqrt(xx(3,j))
c    h1=-rka*dsqrt(xx(1,j))*dsqrt(cT-xx(3,j))*dsqrt(xx(3,j))*(1.0d0/
c    1(cT-xx(3,j))-1.0d0/xx(3,j))/2.0d0
c    h2=rka*dsqrt(cT-xx(3,j))*dsqrt(xx(3,j))/dsqrt(xx(1,j))/2.0d0
c
c &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
c
c    POLYMER
c
```

```
      alpha=0.5d0
      alphc=0.5d0
c       if(cmax-xx(1,j) .le. 0) print*,'cmax',j
c       if(ct-xx(3,j) .le. 0) print*,'ct',j
c       if(xx(1,j) .lt. .01d0) print*,'c',j
      h0=rka*dsqrt(xx(1,j))*dsqrt(cmax-xx(1,j))*dsqrt(ct-xx(3,j))
     1*dsqrt(xx(3,j))
      h1=-rka*dsqrt(xx(1,j))*dsqrt(cmax-xx(1,j))*dsqrt(ct-xx(3,j))
     1*dsqrt(xx(3,j))*(1.0d0/(ct-xx(3,j))-1.0d0/xx(3,j))/2.0d0
      h2=-rka*dsqrt(xx(3,j))*dsqrt(ct-xx(3,j))*dsqrt(cmax-xx(1,j))
     1*dsqrt(xx(1,j))*(1.0d0/(cmax-xx(1,j))-1.0d0/xx(1,j))/2.0d0
c
c &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
c
      r1=alpha*fc/r/t
      r2=r1*(xx(2,j)-g0)
c
      b(3,1)=h2*(dexp(-r2)-dexp(r2))
      b(3,2)=-h0*r1*(dexp(r2)+dexp(-r2))
      b(3,3)=-h1*(dexp(r2)-dexp(-r2))+h0*r1*g1*(dexp(r2)+dexp(-r2))
      b(3,5)=1.0d0
      g(3)=h0*(dexp(r2)-dexp(-r2))+b(3,1)*xx(1,j)+b(3,2)*xx(2,j)+
     1b(3,3)*xx(3,j)
c
      return
      end
c*********************************************************************
      subroutine prop(nj,n1)
      implicit double precision*8(a-h,o-z)
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/var/ xp(10),xx(5,221),xi(5,221),xt(5,221,1600)
      common/tprop/df(221),cd(221),tm(221),
     1ddf(221),dcd(221),dtm(221),dfu(221),d2df(221),
     1d2tm(221),d2fu(221)
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
c
      do j=1,nj
      if(j .gt. n1) ee=ep
      if(j .le. n1) ee=eps
c
c &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
c
c     AsF6 in methyl acetate
c
c     diffusion coefficient of the salt (m2/s)
c     df(j)=(ee**1.5d0)*1.54d-09
c     ddf(j)=0.0d0
c     d2df(j)=0.0d0
c     conductivity of the salt (S/m)
c     cd(j)=2.5d0*(ee**(1.5d0))
c     dcd(j)=0.0d0
c     transference number of lithium
c     tm(j)=0.20d0
c     dtm(j)=0.0d0
```

```
c    d2tm(j)=0.0d0
c    activity factor for the salt
c    dfu(j)=0.0d0
c    d2fu(j)=0.0d0
c
c &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
c
c    Perchlorate in PEO
c
c    diffusion coefficient of the salt (m2/s)
c    df(j)=(ee**1.5d0)*1.78d-12
c    ddf(j)=0.0d0
c    d2df(j)=0.0d0
c    conductivity of the salt (S/m)
c    cd(j)=1.6d-02*ee**(1.5d0)
c    dcd(j)=0.0d0
c    transference number of lithium
c    tm(j)=0.10d0
c    dtm(j)=0.0d0
c    d2tm(j)=0.0d0
c    activity factor for the salt
c    dfu(j)=0.0d0
c    d2fu(j)=0.0d0
c
c &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
c
c    Perchlorate in PC (West's simulation)
c
c    diffusion coefficient of the salt (m2/s)
c    df(j)=(ee**1.5d0)*2.58d-10
c    ddf(j)=0.0d0
c    d2df(j)=0.0d0
c    conductivity of the salt (S/m)
c    pmax=5.42d0
c    pu=0.6616d0
c    aa=0.855d0
c    bb=-0.08d0
c    rho=1.2041d03
c    fun=pmax*((1.0d0/rho/pu)**aa)*dexp(bb*((xx(1,j)/rho-pu)**2.0)
c   1-(aa/pu)*(xx(1,j)/rho-pu))
c    fun2=2.0d0*(bb/rho)*(xx(1,j)/rho-pu)-aa/pu/rho
c    cd(j)=0.0001+(ee**1.5d0)*((xx(1,j))**aa)*fun
c    dcd(j)=(ee**1.5d0)*fun*(aa*(xx(1,j)**(aa-1.0d0))+(xx(1,j)**aa)
c   1*fun2)
c    transference number of lithium
c    tm(j)=0.20d0
c    dtm(j)=0.0d0
c    d2tm(j)=0.0d0
c    activity factor for the salt
c    dfu(j)=0.0d0
c    d2fu(j)=0.0d0
c
c &&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&&
c
```

```
c    Triflate in PEO
c
      r0=3.18466d-05
      r1=1.9915633d-06
      r2=-1.9504432d-09
      r3=6.307483d-13
      r4=-6.752235d-17

c    diffusion coefficient of the salt (m2/s)
c    I have added a temperature function to the diffusion coefficient
c    From equation in Newman relating diffusion coefficient to viscosity
c    and equation in Billmeyer (polymer science) for viscosity of a
c    polymer as a function of temperature. The reference diffusion coefficient
c    is 8*e-12 at 90 C.

      tb = 363.15
      vist=(17.44*(t-213)/(51.6+t-213)-17.44*(tb-213)/(51.6+tb-213))
      df(j)=(ee**1.5d0)*8.0d-12*t/tb*10**vist
      ddf(j)=0.0d0
      d2df(j)=0.0d0
c         print*,'the diffusion coeff is ',df(j),j

c    conductivity of the salt (S/m)
c     cd(j)=(ee**(1.5d0))*100.0d0*(r0 + r1*xx(1,j)+r2*xx(1,j)*xx(1,j)
c     1+r3*xx(1,j)*xx(1,j)*xx(1,j)+r4*xx(1,j)**4.0d0)
c     dcd(j)=(ee**1.5d0)*100.0d0*(r1 + 2.0d0*r2*xx(1,j) +
c     13.0d0*r3*xx(1,j)**2.0d0+4.0d0*r4*xx(1,j)**3.0d0)


* Conductivity as a Function  (S/m) temperature and concentration********
      rr0= 32.1374d0
      rr1= -6.60932d0
      rr2= -0.0251516d0
      rr3=  0.0259545d0
      rr4= -0.0019001d0
      rr5=  3.41711d-5
      rr6= 32.1262d0
      rr7=-44.5889d0
      rr8=  2.59756d0
      rr9= -0.0631626d0
      rr10= 0.000640897d0
      rr11=-2.35565d-6

      cx=xx(1,j)

      ff1=((1.0d0+dtanh(rr0-cx**.5))/2.0d0)*
     + (rr1 + rr2*cx**.5 + rr3*cx + rr4*cx**1.5 + rr5*cx**2)

      ff2=((1.0d0+dtanh(-rr6+cx**.5))/2.0d0)*
     + (rr7 + rr8*cx**.5 + rr9*cx + rr10*cx**1.5 + rr11*cx**2)

      bb1=-1052.65d0
      bb2=-182.053d0
      bb3=3.74936d0
```

```
      bb4=0.0242493d0

      gg=(bb1+bb2*cx**.5+bb3*cx+bb4*cx**1.5)*(1.0/t-1.0/358.0d0)


      cd(j)=(ee**(1.5d0))*100.0d0*cx*dexp(dlog(10.)*(ff1+ff2+gg))

      dff1=((1.0d0+dtanh(rr0-cx**.5))/2.0d0)*
     + (.5*rr2/cx**.5 + rr3 + 1.5*rr4*cx**.5 + 2.0*rr5*cx)

      dff2=(((1.0/(dcosh(rr0-cx**.5))**2))/(-4.0d0*cx**0.5))*
     + (rr1 + rr2*cx**.5 + rr3*cx + rr4*cx**1.5 + rr5*cx**2)

      dff3=((1.0d0+dtanh(-rr6+cx**.5))/2.0d0)*
     + (.5*rr8/cx**.5 + rr9 + 1.5*rr10*cx**.5 + 2.0*rr11*cx)

      dff4=(((1.0/(dcosh(-rr6+cx**.5))**2))/(4.0d0*cx**0.5))*
     + (rr7 + rr8*cx**.5 + rr9*cx + rr10*cx**1.5 + rr11*cx**2)

      dgg=(0.5*bb2/cx**0.5 + bb3 + 1.5*bb4*cx**0.5)*
     + (1.0d0/t - 1.0d0/358.0d0)

      b=dlog(10.)*(ff1+ff2+gg)
      dbdc=dlog(10.)*(dff1+dff2+dff3+dff4+dgg)

      dcd(j)=(ee**(1.5d0))*100.0d0*dexp(b)*(1.0d0+cx*dbdc)


c     transference number of lithium
c      tm(j)=0.0107907d0 + 1.48837d-04*xx(1,j)
c      dtm(j)=1.48837d-04
      tm(j)=0.2d0
      dtm(j)=0.0d0
      d2tm(j)=0.0d0
c     activity factor for the salt
      dfu(j)=0.0d0
      d2fu(j)=0.0d0

      end do
c
      return
      end
*************************************************************
      subroutine temperature(kk,v,q,ut,Uoc)
      implicit double precision*8(a-h,o-z)
      common/const/ fc,r,t,frt,cur,ep,epf,eps,pi,tsmax
      common/temp/htc,dudt,Cp,dens,tam,ncell,g0,tin,lht
      common/cprop/ sig,area,ct,dfs,Rad,cmax,rka,rkf
      common /calc/ ai(1600),u(222,1600),ts(1600),h,h1,h2,h3,rr


      delt=-5.58d-04
```

```
      zeta=8.1d0
      cT=2.9d04
      Uoc=2.17+(dlog(1.0d0/ut-1.0d0)+delt*cT*ut+zeta)*r*353.15/fc

      thk = 2.45d-4
      if (kk.eq.2) print*,htc,dudt,ncell,cur,tam-273.15
c     per cell heat generation

      q = cur*(Uoc-v-t*dudt)

c The heat transfer coefficient, htc, is defined based on the separator area
c  of a single cell for Ncell = 1.


      if (lht.eq.0) then
       t11 = Ncell*dens*Cp*thk/rr
       t12 = t11*t + (Uoc-v)*cur*Ncell + htc*tam
       t13 = t11 + dudt*cur*Ncell + htc
        if (kk.ne.2) then
          t = t12/t13

        end if


     else

**********Calculate htc instead of Temp **********************
c For this case the temperature is assumed constant, and the
c heat transfer coefficient required to keep it constant is
c calculated as a function of time.  The heat transfer coefficient,
c htc, is defined based on the separator area of a single cell for
c Ncell = 1.

        if (t.ne.tam) then
         htc = cur*Ncell*(Uoc-v-t*dudt)/(t-tam)
        else
         htc = 0
        endif

*******************************************************
      endif

      return
      end
```

A.2 Input File for One-Cell Program: tfoil.in

```
21 5.0d-05 1.0d-04 2.0d-5 40 80 90  90  1
1.5d03  1.5d03 2.9d0 2.95d0
2.9d02 2.9d02    7.5d00 0.0d00   -1.0d-06 -1.0d-06   7.5d00
1.0d-13   2.0d-06   0.4  0.1  1.0d0   6.0d01
2.372d04 3.92d03  1.0d-12  3.750d-02  1
1.2041d03   3.22d03   2.0d03   2.0d03 5.796d03  5.0d0
1  2  10
.118  0  746.7d0  1868.0d0  89.0d0  1  0
1
11.0d0    200
```

line 1: lim,h1,h2,n1,n2,t
- lim limit on number of iterations
- h1 thickness of separator (m)
- h2 thickness of positive electrode (m)
- h3 thickness of bipolar partition (m)
- n1 number of nodes in separator
- n2 number of nodes in positive electrode
- t initial temperature (C)
- tmax maximum time step size, (s)
- rr initial time step size

line 2: xi(1,1),xi(1,nj),xi(2,1),xi(2,nj)
- xi(1,j) initial concentration (mol/m3)
- xi(2,j) initial guess for potential, PHI1 - PHI2 (V)

line 3: xi(3,1),xi(3,nj),xi(4,1),xi(4,nj),xi(5,1),xi(5,nj)
- xi(3,j) initial solid concentration (mol/m3)
- xi(4,j) initial guess for current density (A/m2)
- xi(5,j) initial guess for pore wall flux (mol/m2-s)

line 4: dfs,Rad,ep,epf,eps,sig
- dfs diffusion coefficient in solid (m2/s)
- Rad radius of particles (m)
- ep volume fraction of electrolyte in composite electrode
- epf volume fraction of inert filler in composite electrode
- eps volume fraction of electrolyte in separator
- sig conductivity of solid matrix (S/m)

line 5: ct,cmax,rka,rkf,il4
- ct maximum concentration in insertion material (mol/m3)
- cmax maximum concentration in electrolyte (mol/m3)
- rka reaction rate constant for insertion reaction
- rkf exchange current density for lithium foil
- il4 1 for polymer, 0 for liquid electrolyte

line 6: re,rs,rf,ef
- re density of electrolyte (kg/m3)
- rs density of insertion material (kg/m3)

rf density of inert filler (kg/m3)
rc density of separator material (kg/m3)
rp density of bipolar partition (kg/m3)
ef excess capacity of lithium foil

line 7: il1,il2,il3
il1 1 for long print-out 0 for short print-out
il2 1/il2 = fraction of nodes in long print-out
il3 1/il3 = fraction of time steps in long print-out

line 8: htc, dudt, Cp, dens, tam, ncell,lht
htc per-cell heat-transfer coefficient
dudt temperature coefficient of open circuit potential
Cp average heat capacity of system
dens average density of system
tam ambient air temperature
ncell number of cells in a cell stack with negligible temperature gradient
lhc 0 for varying temperature, 1 for varying heat transfer coefficient

line 9: tmax
tmax number of current changes

line 10: cu(i) tt(i)
cu(i) operating current density (A/m2)
tt(i) time (min)

Appendix B: Cell-Stack Program and Input Files

B.1 Cell-Stack Program: Temp11.f

```
c*******************************************************
c
c   Temp11.f   Program for heat transfer
c              in a cell stack with variable rates of energy
c              generation. The energy generation is read
c              from an input file of compiled data from
c              several isothermal runs of tfoil1.f
c*******************************************************

      implicit double precision*8(a-h,o-z)
      dimension B1(100,100), D(100), D1(100,2001),B(100,100)
      dimension To(100,2001),Tq(25)
      dimension tiq(600),qi(600,25)
      dimension qsp(100,2001),tm(100,2001)
      common/int/Tq,qsp,tm
      common/inta/tiq
      common/intb/qi
      common/intc/To
      common/intd/ntem,nmq


c  Definition of terms:

c    cur   cell current
c    tin   initial temperature
c    tam   ambient temperature
c    dudt  entropic term
c    h     heat transfer coefficient based on x-sectional
c          area of cell stack
c    delt  time step
c    sthk  thickness of cell stack (m)
c    cthk  thickness of cell (m)
c    delx  step length in x-direction (fm*cthk)
c    fm    number of cells per mesh point (delx/cthk)
c    dens  composite density
c    Cp    heat capacaty
c    tk    thermal conductivity
c    nts   number of time steps in simulation
c    Ncell number of cells in stack
c    n     = Ncell/2
c    qint  interpolated value of q from input file
c    kc1   print out every kc1 cell for temp vs time
c    kc2   print out every kc2 cell for t or Q vs cell
c    kt1   print out every kt1 time step for temp vs time
c    kt2   print out every kt2 time step for t or Q vs cell

      open (unit=10, file='temp11.in', status='old')
      open (unit=15, file='heat11.in', status='old')
```

```
      read(10,*) tin, tam
      print*,tin,tam
      read(10,*) h, tk, Cp, dens
      print*,h,tk,Cp,dens
      read(10,*) sthk, time,delx,cthk,delt
      print*,sthk,time,delx,cthk,delt
      read(10,*) cur,.dudt
      print*,cur,dudt
      read(10,*) kc1,kt1,kc2,kt2
      print*,kc1,kt1,kc2,kt2

      tin=tin+273.15
      tam=tam+273.15

      Ncell = sthk/delx + 1
      n = Ncell/2
    nts = int(time*60/delt) + 2
    fm = delx/cthk

c   Reading temperatures from input file.  Ntem is number of temps.
    it = 1
    do while (Tcount.ge.0)
      read(15,*),Tq(it)
      Tcount=Tq(it)
      Tq(it) = Tq(it)+273.15
c       print*,Tq(it),it
      it=it+1
    end do

    ntem=it-2


c   Reading heat generation rates and times from input file.
    do 5 iw = 1,ntem
    iq=1
c     print*,Tq(iw),iw,ntem
    do while(dm6.ge.0)
       read(15,*),dm1,dm2,dm3,tiq(iq),icount,dm6,dm7,dm8,qi(iq,iw)
c       print*,tiq(iq),qi(iq,iw),icount,dm1
      iq=iq+1
    end do
    dm6=100
c     print*,' '
 5  continue


c Fill first column of temperature array with initial temperatures.

   do 10 jk = 1,n
     To(jk,1) = tin
```

```
 10 continue

c Fill equation matrix with zeroes.

    do 20 ii = 1,n
     do 20 ij = 1,n
      B1(ii,ij) = 0.0
 20 continue

c Fill equation matrix with values
c First row:

    B1(1,1) = 1.0 + delx/tk*(delx*dens*Cp/delt)
    B1(1,2) = -1.0

c Rows 2 through n-1.
    m=0
    do 30 jj=2,n-1
     m = m + 1
     B1(jj,m) = -1.0
     B1(jj,m+1) = 2.0 + delx/tk*(delx*dens*Cp/delt)
     B1(jj,m+2) = -1.0
 30 continue

c Last Row

    B1(n,n-1) = -tk/delx
    B1(n,n) = h + tk/delx + delx*dens*Cp/delt


c     print*,((B1(i,j),j=1,n),i=1,n)


c Solve

    do 40 kk = 1,nts

     do 45 in = 1,n
      do 45 il = 1,n
       B(in,il) = B1(in,il)
45   continue


     t = (kk-1)*delt
     call interp(t,1,kk,qint)
     D(1) = delx/tk*(fm*qint + delx*dens*Cp*To(1,kk)/delt)
     D1(1,kk) = D(1)

     do 50 nn = 2,n-1
     call interp(t,nn,kk,qint)
      D(nn) = delx/tk*(fm*qint + delx*dens*Cp*To(nn,kk)/delt)
      D1(nn,kk) = D(nn)
```

```
 50    continue

       call interp(t,n,kk,qint)
       D(n) = h*Tam + fm*qint + delx*dens*Cp*To(n,kk)/delt
       D1(n,kk) = D(n)


       call matinv(n,1,determ,B,D)

c  Rewrite solution array into To matrix.

       do 60 ki = 1,n
c         print*,D1(ki,kk),D(ki)
         To(ki,kk+1) = D(ki)
c         print*,To(ki,kk+1)
 60    continue

 40  continue


c-----------------------------------------------------------------

c      Printing Routine

c  Printing temperatures as function of distance and cell number.

    print*,'Temperatures in cell stack as functions of distance'
     print*,'and cell.'
         print*
         print77,'time ----->'
         print78,'cell','x in m',((i-1)*delt/60.0,i=1,nts,kt2),
   +(nts-1)*delt/60.0


    k1=0
    j1=1
    zero=0

     print79,j1,zero,(To(1,i)-273.15, i=1,nts,kt2),To(1,nts)-273.15

    do 70 j = 1,n,kc2
      print79,j,j*delx,(To(j,i)-273.15,i=1,nts,kt2),To(j,nts)-273.15
     k1=j
 70  continue

    if (k1.ne.n) then
      print79,n,n*delx,(To(n,i)-273.15,i=1,nts,kt2),To(n,nts)-273.15
    endif

    print*
    print*
```

```
c-----------------------------------------------------------------
c  Printing heat generation rates as functions of distance and cell number.

     print*,'Heat Generation Rates in cell stack as functions of'
     print*,'distance and cell.'
     print*
     print77,'time ----->'
      print78,'cell','x in m,',((i-1)*delt/60.0,i=1,nts,kt2),
    +(nts-1)*delt/60.0


     k1=0
     j1=1
     zero=0

      print79,j1,zero,(qsp(1,i), i=1,nts,kt2),qsp(1,nts)

     do 71 j = 1,n,kc2
       print79,j,j*delx,(qsp(j,i),i=1,nts,kt2),qsp(j,nts)
      k1=j
 71  continue

     if (k1.ne.n) then
       print79,n,n*delx,(qsp(n,i),i=1,nts,kt2),qsp(n,nts)
     endif

     print*
     print*


c-----------------------------------------------------------------
c  Printing temperatures as functions of time.

     k1=0
     j1 = 1

     print*, 'Temperature of cells as functions of time'
     print*
      print 81, 'time,', (i, i=1,n,kc1), n


     do 72 j = 1,nts,kt1
      time=(j-1)*delt/60.0
       print80,time,(To(i,j)-273.15,i=1,n,kc1),To(n,j)-273.15
      k1=j
 72  continue

     if(k1.ne.nts) then
         time=(nts-1)*delt/60.0
       print80,time,(To(i,nts)-273.15,i=1,n,kc1),To(n,nts)-273.15
     endif
```

```
      print*
      print*


c-----------------------------------------------------------------
c  Printing heat generation rates as functions of time.

      k1=0
      time=0

      print*,'Heat Generation Rate in Cells'
      print*
       print 81, 'time,', (i, i=1,n,kc1), n


      do 73 j = 1,nts,kt1
       time=(j-1)*delt/60.0
        print80,time,(qsp(i,j),i=1,n,kc1),qsp(n,j)
       k1=j
 73   continue

      if(k1.ne.nts) then
       time=(nts-1)*delt/60.0
        print80,time,(qsp(i,nts),i=1,n,kc1),qsp(n,nts)
      endif

 77   format(22X,A11)
 78     format(2(A10,','),18('t'f9.4,',',X))
 79      format(I10,',',f9.6,',',X,18(f10.4,',',X))
 80   format(20(F10.4,',',X))
 81   format(A11,19(3X'cell'X,i3',',))


      end

c*****************************************************************
      subroutine interp(t,nnq,kkq,qint)
      implicit double precision*8(a-h,o-z)
      dimension To(100,2001),Tq(25)
      dimension tiq(600),qi(600,25)
      dimension qsp(100,2001),tm(100,2001)
      common/int/Tq,qsp,tm
      common/inta/tiq
      common/intb/qi
      common/intc/To
      common/intd/ntem,nmq

c      do ic=1,226,20
c        print*,(qi(ic,jc),jc=1,5)
c      end do

c     The temperature of interest is To(nnq, kkq)
```

```
      time=t/60.0d0
      tol = 0.1
      If (time.lt.tol) then
        qint = 0
c         print*,'1',time,nnq,kkq,qint
c         qsp(nnq,kkq) = qint
        return
      end if


c  Locating data points between jj and jj-1 temperatures.

      jj=1
 10   if (Tq(jj).lt.To(nnq,kkq)) then
c        print*,Tq(jj),To(nnq,kkq)
       jj = jj+1
       go to 10
      endif

c  Locating data points between ii and ii-1 time steps.
      ii=1
 20   If (tiq(ii).lt.time) then
c       print*,time, tiq(ii)
       ii = ii+1
       go to 20
      endif


c  Interpolating Data
c     print*,tiq(ii), tiq(ii-1),time,ii,qi(ii-1,jj-1),qi(ii,jj-1)
      s1 = (time-tiq(ii-1))/(tiq(ii)-tiq(ii-1))
      q1 = qi(ii-1,jj-1) + s1*(qi(ii,jj-1)-qi(ii-1,jj-1))

*       print*,q1,'before s2'
      s2 = (time-tiq(ii-1))/(tiq(ii)-tiq(ii-1))
      q2 = qi(ii-1,jj) + s2*(qi(ii,jj)-qi(ii-1,jj))

*       print*,q2,'before s3'
      s3 = (To(nnq,kkq)-Tq(jj-1))/(Tq(jj)-Tq(jj-1))
      qint = q1 + s3*(q2-q1)
*       print*,'2',time, nnq, kkq, qint

      qsp(nnq,kkq)=qint
      tm(nnq,kkq)=time
c       if(nnq.eq.1) print*,time,' ',qint,' ',To(nnq,kkq)


      return
      end
```

```
c*****************************************************************
      subroutine matinv(n,m,determ,B,D)
      implicit double precision*8(a-h,o-z)
      dimension b(100,100),d(100,1)
      dimension id(100)
      determ=1.0
      do 1 i=1,n
    1 id(i)=0
      do 18 nn=1,n
      bmax=1.1
      do 6 i=1,n
      if(id(i).ne.0) go to 6
      bnext=0.0
      btry=0.0
      do 5 j=1,n
      if(id(j).ne.0) go to 5
      if(dabs(b(i,j)).le.bnext) go to 5
      bnext=dabs(b(i,j))
      if(bnext.le.btry) go to 5
      bnext=btry
      btry=dabs(b(i,j))
      jc=j
    5 continue
      if(bnext.ge.bmax*btry) go to 6
      bmax=bnext/btry
      irow=i
      jcol=jc
    6 continue
      if(id(jc).eq.0) go to 8
      determ=0.0
      return
    8 id(jcol)=1
      if(jcol.eq.irow) go to 12
      do 10 j=1,n
      save=b(irow,j)
      b(irow,j)=b(jcol,j)
   10 b(jcol,j)=save
      do 11 k=1,m
      save=d(irow,k)
      d(irow,k)=d(jcol,k)
   11 d(jcol,k)=save
   12 f=1.0/b(jcol,jcol)
      do 13 j=1,n
   13 b(jcol,j)=b(jcol,j)*f
      do 14 k=1,m
   14 d(jcol,k)=d(jcol,k)*f
      do 18 i=1,n
      if(i.eq.jcol) go to 18
      f=b(i,jcol)
      do 16 j=1,n
```

```
16  b(i,j)=b(i,j)-f*b(jcol,j)
    do 17 k=1,m
17  d(i,k)=d(i,k)-f*d(jcol,k)
18  continue
    return
    end
```

B.2 Input File for Temp11.f: temp11.in

```
90.0  89.0
12  0.64  746.7  1868.0
.0441  180  4.9d-4 2.45d-4  50
11  0
5  5  2  30
```

Line 1: tin, tam
- tin — initial temperature of cell stack (C)
- tam — ambient air temperature (C)

Line 2: h, tk, Cp, dens
- h — heat transfer coefficient (W/m2K) based on area of one cell
- tk — thermal conductivity (W/mK) parallel to layers
- Cp — heat capacity (J/kg-K)
- dens — density of cell

Line 3: n, nts, delx, delt
- sthk — cell stack thickness (m)
- time — simulation time (min)
- delx — step length in x-direction (m)
- cthk — thickness of cell (m)
- delt — size of time step (s)

Line 4: cur, dudt
- cur — current (A/m2)
- dudt — entropic term

Line 5: printing
(good values for Ncell = 180 and time = 180 min, tstep = 10.9 sec.)
- kc1 — print out every kc1 cell for T or Q vs time (across, 10)
- kt1 — print out every kt1 time step for T or Q vs time (down, 10)
- kc2 — print out every kc2 cell for T vs cell (down, 1)
- kt2 — print out every kt2 time step for T vs cell (across, 100)

B.3 Input File for Temp11.f: heat11.in

The first block gives the temperatures that data are present for in the input file. The second block gives output from the one-cell model. The only columns that are read by the program are the fourth column, the time (in minutes), and the eighth column, the heat-generation rate (in $W/m^2 \cdot K$). Only the first page of the input file is included in this Appendix.

```
88
90
92
94
96
98
100
103.3
106.6
110
115
120
130
140
150
-100

 .01472, 2.46105,  1.000,    1.000,  1,   88.0000,   2.53716,  -16.74380,
.83719
 .01943, 2.43511,  1.000,    2.000,  2,   88.0000,   2.52623,  -20.04595,
1.00230
 .02415, 2.41669,  1.000,    3.000,  3,   88.0000,   2.51715,  -22.10137,
1.10507
 .02887, 2.40089,  1.000,    4.000,  4,   88.0000,   2.50925,  -23.83877,
1.19194
 .03359, 2.38700,  1.000,    5.000,  5,   88.0000,   2.50217,  -25.33820,
1.26691
 .03830, 2.37404,  1.000,    6.000,  6,   88.0000,   2.49570,  -26.76559,
1.33828
 .04302, 2.36213,  1.000,    7.000,  7,   88.0000,   2.48969,  -28.06288,
1.40314
 .04774, 2.35089,  1.000,    8.000,  8,   88.0000,   2.48405,  -29.29500,
1.46475
 .05246, 2.34033,  1.000,    9.000,  9,   88.0000,   2.47871,  -30.44503,
1.52225
 .05717, 2.33013,  1.000,   10.000, 10,   88.0000,   2.47362,  -31.56777,
1.57839
 .06189, 2.32029,  1.000,   11.000, 11,   88.0000,   2.46873,  -32.65675,
1.63284
 .06661, 2.31069,  1.000,   12.000, 12,   88.0000,   2.46402,  -33.73210,
1.68661
 .07133, 2.30137,  1.000,   13.000, 13,   88.0000,   2.45946,  -34.77937,
1.73897
```

LAWRENCE BERKELEY LABORATORY
UNIVERSITY OF CALIFORNIA
TECHNICAL INFORMATION DEPARTMENT
BERKELEY, CALIFORNIA 94720