

ANSYS FLUENT Fuel Cell Modules Manual



ANSYS, Inc.
Southpointe
275 Technology Drive
Canonsburg, PA 15317
ansysinfo@ansys.com
<http://www.ansys.com>
(T) 724-746-3304
(F) 724-514-9494

Release 14.0
November 2011

ANSYS, Inc. is
certified to ISO
9001:2008.

Copyright and Trademark Information

© 2011 SAS IP, Inc. All rights reserved. Unauthorized use, distribution or duplication is prohibited.

ANSYS, ANSYS Workbench, Ansoft, AUTODYN, EKM, Engineering Knowledge Manager, CFX, FLUENT, HFSS and any and all ANSYS, Inc. brand, product, service and feature names, logos and slogans are registered trademarks or trademarks of ANSYS, Inc. or its subsidiaries in the United States or other countries. ICEM CFD is a trademark used by ANSYS, Inc. under license. CFX is a trademark of Sony Corporation in Japan. All other brand, product, service and feature names or trademarks are the property of their respective owners.

Disclaimer Notice

THIS ANSYS SOFTWARE PRODUCT AND PROGRAM DOCUMENTATION INCLUDE TRADE SECRETS AND ARE CONFIDENTIAL AND PROPRIETARY PRODUCTS OF ANSYS, INC., ITS SUBSIDIARIES, OR LICENSORS. The software products and documentation are furnished by ANSYS, Inc., its subsidiaries, or affiliates under a software license agreement that contains provisions concerning non-disclosure, copying, length and nature of use, compliance with exporting laws, warranties, disclaimers, limitations of liability, and remedies, and other provisions. The software products and documentation may be used, disclosed, transferred, or copied only in accordance with the terms and conditions of that software license agreement.

ANSYS, Inc. is certified to ISO 9001:2008.

U.S. Government Rights

For U.S. Government users, except as specifically granted by the ANSYS, Inc. software license agreement, the use, duplication, or disclosure by the United States Government is subject to restrictions stated in the ANSYS, Inc. software license agreement and FAR 12.212 (for non-DOD licenses).

Third-Party Software

See the [legal information](#) in the product help files for the complete Legal Notice for ANSYS proprietary software and third-party software. If you are unable to access the Legal Notice, please contact ANSYS, Inc.

Published in the U.S.A.

Table of Contents

Using This Manual	v
1. The Contents of This Manual	v
2. The Contents of the FLUENT Manuals	v
3. Typographical Conventions	vi
4. Mathematical Conventions	vii
5. Technical Support	viii
1. Fuel Cell and Electrolysis Model Theory	1
1.1. Introduction	1
1.1.1. Introduction to PEMFC	2
1.1.2. Introduction to SOFC	3
1.1.3. Introduction to Electrolysis	3
1.2. Electrochemistry Modeling	4
1.3. Current and Mass Conservation	7
1.4. Heat Source	8
1.5. Liquid Water Formation, Transport, and its Effects (PEMFC Only)	9
1.6. Properties	10
1.7. Transient Simulations	12
1.8. Specific Leakage Current (Cross-Over Current)	13
2. Using the Fuel Cell and Electrolysis Model	15
2.1. Introduction	15
2.2. Geometry Definition for the Fuel Cell and Electrolysis Model	15
2.3. Installing the Fuel Cell and Electrolysis Model	16
2.4. Loading the Fuel Cell and Electrolysis Module	16
2.5. Setting Up the Fuel Cell and Electrolysis Module	16
2.6. Modeling Fuel Cells and Electrolysis	17
2.6.1. Specifying Model Options	18
2.6.2. Specifying Model Parameters	20
2.6.3. Specifying Anode Properties	22
2.6.3.1. Specifying Current Collector Properties for the Anode	22
2.6.3.2. Specifying Flow Channel Properties for the Anode	23
2.6.3.3. Specifying Porous Electrode Properties for the Anode	24
2.6.3.4. Specifying Catalyst Layer Properties for the Anode	25
2.6.3.5. Specifying Cell Zone Conditions for the Anode	25
2.6.4. Specifying Electrolyte/Membrane Properties	26
2.6.4.1. Specifying Cell Zone Conditions for the Membrane	27
2.6.5. Specifying Cathode Properties	27
2.6.5.1. Specifying Current Collector Properties for the Cathode	27
2.6.5.2. Specifying Flow Channel Properties for the Cathode	28
2.6.5.3. Specifying Porous Electrode Properties for the Cathode	29
2.6.5.4. Specifying Catalyst Layer Properties for the Cathode	30
2.6.5.5. Specifying Cell Zone Conditions for the Cathode	30
2.6.6. Setting Advanced Properties	31
2.6.6.1. Setting Contact Resistivities for the Fuel Cell and Electrolysis Model	31
2.6.6.2. Setting Coolant Channel Properties for the Fuel Cell and Electrolysis Model	32
2.6.6.3. Managing Stacks for the Fuel Cell and Electrolysis Model	33
2.6.7. Reporting on the Solution	35
2.7. Modeling Current Collectors	35
2.8. Fuel Cell and Electrolysis Model Boundary Conditions	37
2.9. Solution Guidelines for the Fuel Cell and Electrolysis Model	37
2.10. Postprocessing the Fuel Cell and Electrolysis Model	38

2.11. User-Accessible Functions	39
2.11.1. Compiling the Customized Fuel Cell and Electrolysis Source Code	42
2.11.1.1. Compiling the Customized Source Code Under Linux	42
2.11.1.2. Compiling the Customized Source Code Under Windows	43
2.12. Using the Fuel Cell and Electrolysis Text User Interface	43
2.12.1. IV-Curve Calculations Using the Text Interface	45
3. SOFC Fuel Cell With Unresolved Electrolyte Model Theory	49
3.1. Introduction	49
3.2. The SOFC With Unresolved Electrolyte Modeling Strategy	50
3.3. Modeling Fluid Flow, Heat Transfer, and Mass Transfer	51
3.4. Modeling Current Transport and the Potential Field	52
3.4.1. Cell Potential	52
3.4.2. Activation Overpotential	53
3.4.3. Treatment of the Energy Equation at the Electrolyte Interface	55
3.4.4. Treatment of the Energy Equation in the Conducting Regions	57
3.5. Modeling Reactions	57
3.5.1. Modeling Electrochemical Reactions	57
3.5.2. Modeling CO Electrochemistry	58
4. Using the Solid Oxide Fuel Cell With Unresolved Electrolyte Model	61
4.1. Installing the Solid Oxide Fuel Cell With Unresolved Electrolyte Model	61
4.2. Loading the Solid Oxide Fuel Cell With Unresolved Electrolyte Module	61
4.3. Solid Oxide Fuel Cell With Unresolved Electrolyte Module Set Up Procedure	62
4.4. Setting the Parameters for the SOFC With Unresolved Electrolyte Model	67
4.5. Setting Up the Electrochemistry Parameters	69
4.6. Setting Up the Electrode-Electrolyte Interfaces	71
4.6.1. Setting Up the Anode Electrode-Electrolyte Interface	71
4.6.2. Setting Up the Cathode Electrode-Electrolyte Interface	72
4.6.3. Setting Up the Tortuosity Parameters	73
4.7. Setting Up the Electric Field Model Parameters	74
4.8. User-Accessible Functions for the Solid Oxide Fuel Cell With Unresolved Electrolyte Model	75
4.8.1. Compiling the Customized Solid Oxide Fuel Cell With Unresolved Electrolyte Source Code	76
4.8.1.1. Compiling the Customized Source Code Under Linux	76
4.8.1.2. Compiling the Customized Source Code Under Windows	77
4.9. Using the Solid Oxide Fuel Cell With Unresolved Electrolyte Text User Interface	78
Bibliography	79
Index	81

Using This Manual

This preface is divided into the following sections:

1. [The Contents of This Manual](#)
2. [The Contents of the FLUENT Manuals](#)
3. [Typographical Conventions](#)
4. [Mathematical Conventions](#)
5. [Technical Support](#)

1. The Contents of This Manual

The ANSYS FLUENT Fuel Cell Modules Manual provides information about the background and the usage of two separate add-on fuel cell modules for ANSYS FLUENT. For each type of fuel cell add-on module, you will find background information pertaining to the models, a theoretical discussion of the models used in ANSYS FLUENT, and a description of using the models for your CFD simulations. The available ANSYS FLUENT add-on fuel cell modules are:

- Fuel Cell and Electrolysis Model - allows you to model polymer electrolyte membrane fuel cells (PEMFC), solid oxide fuel cells (SOFC), and electrolysis with ANSYS FLUENT. This model is sometimes referred to as the Resolved Electrolyte model. For more information, see [Fuel Cell and Electrolysis Model Theory \(p. 1\)](#) and [Using the Fuel Cell and Electrolysis Model \(p. 15\)](#).
- SOFC With Unresolved Electrolyte Model - allows you to model solid oxide fuel cells (SOFC). For more information, see [SOFC Fuel Cell With Unresolved Electrolyte Model Theory \(p. 49\)](#) and [Using the Solid Oxide Fuel Cell With Unresolved Electrolyte Model \(p. 61\)](#).

2. The Contents of the FLUENT Manuals

The manuals listed below form the FLUENT product documentation set. They include descriptions of the procedures, commands, and theoretical details needed to use FLUENT products.

[FLUENT Getting Started Guide](#) contains general information about getting started with using FLUENT.

[FLUENT User's Guide](#) contains detailed information about using FLUENT, including information about the user interface, reading and writing files, defining boundary conditions, setting up physical models, calculating a solution, and analyzing your results.

[FLUENT in Workbench User's Guide](#) contains information about getting started with and using FLUENT within the Workbench environment.

[FLUENT Theory Guide](#) contains reference information for how the physical models are implemented in FLUENT.

[FLUENT UDF Manual](#) contains information about writing and using user-defined functions (UDFs).

[FLUENT Tutorial Guide](#) contains a number of example problems with detailed instructions, commentary, and postprocessing of results.

[FLUENT Text Command List](#) contains a brief description of each of the commands in FLUENT's text interface.

[FLUENT Adjoint Solver Module Manual](#) contains information about the background and usage of FLUENT's Adjoint Solver Module that allows you to obtain detailed sensitivity data for the performance of a fluid system.

[FLUENT Battery Module Manual](#) contains information about the background and usage of FLUENT's Battery Module that allows you to analyze the behavior of electric batteries.

[FLUENT Continuous Fiber Module Manual](#) contains information about the background and usage of FLUENT's Continuous Fiber Module that allows you to analyze the behavior of fiber flow, fiber properties, and coupling between fibers and the surrounding fluid due to the strong interaction that exists between the fibers and the surrounding gas.

[FLUENT Fuel Cell Modules Manual](#) contains information about the background and the usage of two separate add-on fuel cell models for FLUENT that allow you to model polymer electrolyte membrane fuel cells (PEMFC), solid oxide fuel cells (SOFC), and electrolysis with FLUENT.

[FLUENT Magnetohydrodynamics \(MHD\) Module Manual](#) contains information about the background and usage of FLUENT's Magnetohydrodynamics (MHD) Module that allows you to analyze the behavior of electrically conducting fluid flow under the influence of constant (DC) or oscillating (AC) electromagnetic fields.

[FLUENT Migration Manual](#) contains information about transitioning from the previous release of FLUENT, including details about new features, solution changes, and text command list changes.

[FLUENT Population Balance Module Manual](#) contains information about the background and usage of FLUENT's Population Balance Module that allows you to analyze multiphase flows involving size distributions where particle population (as well as momentum, mass, and energy) require a balance equation.

[Running FLUENT Under LSF](#) contains information about the using FLUENT with Platform Computing's LSF software, a distributed computing resource management tool.

[Running FLUENT Under PBS Professional](#) contains information about the using FLUENT with Altair PBS Professional, an open workload management tool for local and distributed environments.

[Running FLUENT Under SGE](#) contains information about the using FLUENT with Sun Grid Engine (SGE) software, a distributed computing resource management tool.

3. Typographical Conventions

Several typographical conventions are used in this manual's text to facilitate your learning process.

- Different type styles are used to indicate graphical user interface menu items and text interface menu items (for example, **Iso-Surface** dialog box, `surface/iso-surface` command).
- The text interface type style is also used when illustrating exactly what appears on the screen or exactly what you need to type into a field in a dialog box. The information displayed on the screen is enclosed in a large box to distinguish it from the narrative text, and user inputs are often enclosed in smaller boxes.
- A mini flow chart is used to guide you through the navigation pane, which leads you to a specific task page or dialog box. For example,

 **Models** →  **Multiphase** → **Edit...**

indicates that **Models** is selected in the navigation pane, which then opens the corresponding task page. In the **Models** task page, **Multiphase** is selected from the list. Clicking the **Edit...** button opens the **Multiphase** dialog box.

Also, a mini flow chart is used to indicate the menu selections that lead you to a specific command or dialog box. For example,

Define → **Injections...**

indicates that the **Injections...** menu item can be selected from the **Define** pull-down menu, and

`display` → `mesh`

indicates that the `mesh` command is available in the `display` text menu.

In this manual, mini flow charts usually precede a description of a dialog box or command, or a screen illustration showing how to use the dialog box or command. They allow you to look up information about a command or dialog box and quickly determine how to access it without having to search the preceding material.

- The menu selections that will lead you to a particular dialog box or task page are also indicated (usually within a paragraph) using a "/". For example, **Define/Materials...** tells you to choose the **Materials...** menu item from the **Define** pull-down menu.

4. Mathematical Conventions

- Where possible, vector quantities are displayed with a raised arrow (e.g., \vec{a} , \vec{A}). Boldfaced characters are reserved for vectors and matrices as they apply to linear algebra (e.g., the identity matrix, I).
- The operator ∇ , referred to as grad, nabla, or del, represents the partial derivative of a quantity with respect to all directions in the chosen coordinate system. In Cartesian coordinates, ∇ is defined to be

$$\frac{\partial}{\partial x} \vec{i} + \frac{\partial}{\partial y} \vec{j} + \frac{\partial}{\partial z} \vec{k} \quad (1)$$

∇ appears in several ways:

- The gradient of a scalar quantity is the vector whose components are the partial derivatives; for example,

$$\nabla p = \frac{\partial p}{\partial x} \vec{i} + \frac{\partial p}{\partial y} \vec{j} + \frac{\partial p}{\partial z} \vec{k} \quad (2)$$

- The gradient of a vector quantity is a second-order tensor; for example, in Cartesian coordinates,

$$\nabla (\vec{v}) = \left(\frac{\partial}{\partial x} \vec{i} + \frac{\partial}{\partial y} \vec{j} + \frac{\partial}{\partial z} \vec{k} \right) (v_x \vec{i} + v_y \vec{j} + v_z \vec{k}) \quad (3)$$

This tensor is usually written as

$$\begin{pmatrix} \frac{\partial v_x}{\partial x} & \frac{\partial v_x}{\partial y} & \frac{\partial v_x}{\partial z} \\ \frac{\partial v_y}{\partial x} & \frac{\partial v_y}{\partial y} & \frac{\partial v_y}{\partial z} \\ \frac{\partial v_z}{\partial x} & \frac{\partial v_z}{\partial y} & \frac{\partial v_z}{\partial z} \end{pmatrix} \quad (4)$$

- The divergence of a vector quantity, which is the inner product between ∇ and a vector; for example,

$$\nabla \cdot \vec{v} = \frac{\partial v_x}{\partial x} + \frac{\partial v_y}{\partial y} + \frac{\partial v_z}{\partial z} \quad (5)$$

- The operator $\nabla \cdot \nabla$, which is usually written as ∇^2 and is known as the Laplacian; for example,

$$\nabla^2 T = \frac{\partial^2 T}{\partial x^2} + \frac{\partial^2 T}{\partial y^2} + \frac{\partial^2 T}{\partial z^2} \quad (6)$$

$\nabla^2 T$ is different from the expression $(\nabla T)^2$, which is defined as

$$(\nabla T)^2 = \left(\frac{\partial T}{\partial x}\right)^2 + \left(\frac{\partial T}{\partial y}\right)^2 + \left(\frac{\partial T}{\partial z}\right)^2 \quad (7)$$

- An exception to the use of ∇ is found in the discussion of Reynolds stresses in "Modeling Turbulence" in the [User's Guide](#), where convention dictates the use of Cartesian tensor notation. In this chapter, you will also find that some velocity vector components are written as u , v , and w instead of the conventional v with directional subscripts.

5. Technical Support

If you encounter difficulties while using ANSYS FLUENT, please first refer to the section(s) of the manual containing information on the commands you are trying to use or the type of problem you are trying to solve. The product documentation is available from the online help, or from the ANSYS Customer Portal (www.ansys.com/customerportal).

If you encounter an error, please write down the exact error message that appeared and note as much information as you can about what you were doing in ANSYS FLUENT.

Technical Support for ANSYS, Inc. products is provided either by ANSYS, Inc. directly or by one of our certified ANSYS Support Providers. Please check with the ANSYS Support Coordinator (ASC) at your company to determine who provides support for your company, or go to www.ansys.com and select **About ANSYS> Contacts and Locations**. The direct URL is: <http://www1.ansys.com/customer/public/sup->

[portlist.asp](#). Follow the on-screen instructions to obtain your support provider contact information. You will need your customer number. If you don't know your customer number, contact the ASC at your company.

If your support is provided by ANSYS, Inc. directly, Technical Support can be accessed quickly and efficiently from the ANSYS Customer Portal, which is available from the ANSYS Website (www.ansys.com) under **Support > Technical Support** where the Customer Portal is located. The direct URL is: <http://www.ansys.com/customerportal>.

One of the many useful features of the Customer Portal is the Knowledge Resources Search, which can be found on the Home page of the Customer Portal.

Systems and installation Knowledge Resources are easily accessible via the Customer Portal by using the following keywords in the search box: *Systems/Installation*. These Knowledge Resources provide solutions and guidance on how to resolve installation and licensing issues quickly.

NORTH AMERICA

All ANSYS, Inc. Products

Web: Go to the ANSYS Customer Portal (<http://www.ansys.com/customerportal>) and select the appropriate option.

Toll-Free Telephone: 1.800.711.7199

Fax: 1.724.514.5096

Support for University customers is provided only through the ANSYS Customer Portal.

GERMANY

ANSYS Mechanical Products

Telephone: +49 (0) 8092 7005-55

Email: support@cadfem.de

All ANSYS Products

Web: Go to the ANSYS Customer Portal (<http://www.ansys.com/customerportal>) and select the appropriate option.

National Toll-Free Telephone:

German language: 0800 181 8499

English language: 0800 181 1565

International Telephone:

German language: +49 6151 3644 300

English language: +49 6151 3644 400

Email: support-germany@ansys.com

UNITED KINGDOM

All ANSYS, Inc. Products

Web: Go to the ANSYS Customer Portal (<http://www.ansys.com/customerportal>) and select the appropriate option.

Telephone: +44 (0) 870 142 0300

Fax: +44 (0) 870 142 0302

Email: support-uk@ansys.com

Support for University customers is provided only through the ANSYS Customer Portal.

JAPAN

CFX , ICEM CFD and Mechanical Products

Telephone: +81-3-5324-8333

Fax: +81-3-5324-7308

Email: *CFX:* japan-cfx-support@ansys.com; *Mechanical:* japan-ansys-support@ansys.com

FLUENT Products

Telephone: +81-3-5324-7305

Email: *FLUENT:* japan-fluent-support@ansys.com; *POLYFLOW:* japan-polyflow-support@ansys.com; *FfC:* japan-ffc-support@ansys.com; *FloWizard:* japan-flowizard-support@ansys.com

Icepak

Telephone: +81-3-5324-7444

Email: japan-icepak-support@ansys.com

Licensing and Installation

Email: japan-license-support@ansys.com

INDIA

ANSYS Products (including FLUENT, CFX, ICEM-CFD)

Web: Go to the ANSYS Customer Portal (<http://www.ansys.com/customerportal>) and select the appropriate option.

Telephone: +91 1 800 233 3475 (toll free) or +91 1 800 209 3475 (toll free)

Fax: +91 80 2529 1271

Email: *FEA products:* feasup-india@ansys.com; *CFD products:* cfdsup-india@ansys.com; *Installation:* installation-india@ansys.com

FRANCE

All ANSYS, Inc. Products

Web: Go to the ANSYS Customer Portal (<http://www.ansys.com/customerportal>) and select the appropriate option.

Toll-Free Telephone: +33 (0) 800 919 225

Email: support-france@ansys.com

Support for University customers is provided only through the ANSYS Customer Portal.

BELGIUM

All ANSYS Products

Web: Go to the ANSYS Customer Portal (<http://www.ansys.com/customerportal>) and select the appropriate option.

Telephone: +32 (0) 10 45 28 61

Email: support-belgium@ansys.com

Support for University customers is provided only through the ANSYS Customer Portal.

SWEDEN

All ANSYS Products

Web: Go to the ANSYS Customer Portal (<http://www.ansys.com/customerportal>) and select the appropriate option.

Telephone: +44 (0) 870 142 0300

Email: support-sweden@ansys.com

Support for University customers is provided only through the ANSYS Customer Portal.

SPAIN and PORTUGAL

All ANSYS Products

Web: Go to the ANSYS Customer Portal (<http://www.ansys.com/customerportal>) and select the appropriate option.

Telephone: +33 1 30 60 15 63

Email: support-spain@ansys.com

Support for University customers is provided only through the ANSYS Customer Portal.

ITALY

All ANSYS Products

Web: Go to the ANSYS Customer Portal (<http://www.ansys.com/customerportal>) and select the appropriate option.

Telephone: +39 02 89013378

Email: support-italy@ansys.com

Support for University customers is provided only through the ANSYS Customer Portal.

Chapter 1: Fuel Cell and Electrolysis Model Theory

This chapter presents the theoretical background for the Fuel Cell and Electrolysis modeling capabilities in ANSYS FLUENT.

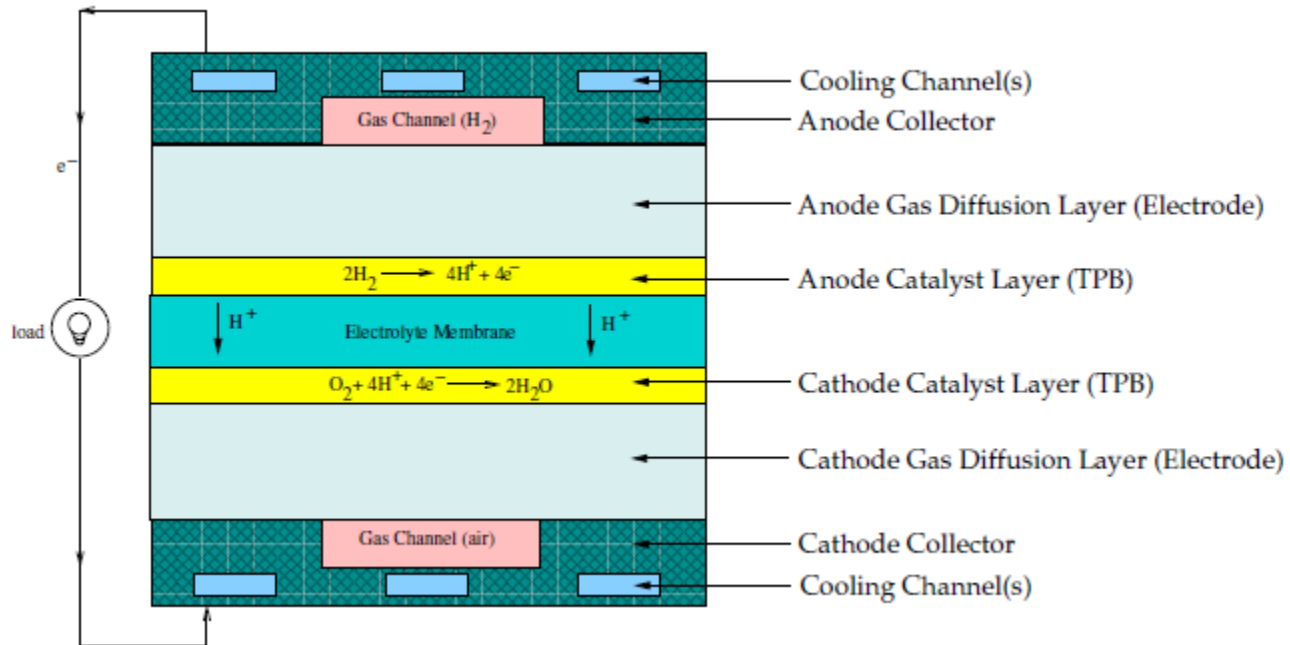
- 1.1. Introduction
- 1.2. Electrochemistry Modeling
- 1.3. Current and Mass Conservation
- 1.4. Heat Source
- 1.5. Liquid Water Formation, Transport, and its Effects (PEMFC Only)
- 1.6. Properties
- 1.7. Transient Simulations
- 1.8. Specific Leakage Current (Cross-Over Current)

1.1. Introduction

The Fuel Cell and Electrolysis module (sometimes referred to as the Resolved Electrolyte module) is provided as an add-on module with the standard ANSYS FLUENT licensed software. A special license is required to use this module.

A fuel cell is an energy conversion device that converts the chemical energy of fuel into electrical energy. With the Fuel Cell and Electrolysis Model, both the triple-phase boundary (TPB), also known as the catalyst layer, and the ionic conducting electrolyte (also known as the membrane in PEMFC terminology) are included in the computational domain. The Fuel Cell and Electrolysis module allows you to model PEMFC, SOFC, and high-temperature electrolysis.

To determine the physical domains that are included in the Fuel Cell and Electrolysis module, a schematic of a polymer electrolyte membrane fuel cell (PEMFC) is shown in [Figure 1.1 \(p. 2\)](#).

Figure 1.1 Schematic of a PEM Fuel Cell

Hydrogen flows into the fuel cell on the anode side. It diffuses through the porous gas diffusion layers and comes in contact with the catalyst layer. Here it forms hydrogen ions and electrons. The hydrogen ions diffuse through the polymer electrolyte membrane at the center, the electrons flow through the gas diffusion layer to the current collectors and into the electric load attached. Electrons enter the cathode side through the current collectors and the gas diffusion layer. At the catalyst layer on the cathode side, the electrons, the hydrogen ions, and the oxygen combine to form water.

In the Fuel Cell and Electrolysis Model in ANSYS FLUENT, two electric potential fields are solved. One potential is solved in the electrolyte and the TPB catalyst layer. The other is solved in the TPB catalyst layer, the porous electrode, and the current collectors. The rate of electrochemical reactions are computed in the TPB layers at both the anode and the cathode. Based on the cell voltage that you prescribe, the current density value is computed. Alternatively, a cell voltage can be computed based on a prescribed average current density.

For more information, please see the following sections:

- 1.1.1. Introduction to PEMFC
- 1.1.2. Introduction to SOFC
- 1.1.3. Introduction to Electrolysis

1.1.1. Introduction to PEMFC

Over the last decade, the proton exchange membrane fuel cell (PEMFC) has emerged as a favored technology for auto transportation and power generation because it is compact, clean, runs at low temperature ($< 100^\circ\text{C}$), permits an adjustable power output, and can be started relatively rapidly. Hydrogen is supplied at the anode and air is supplied at the cathode. The following electrochemical reactions take place in the anode and cathode triple phase boundary (TPB) layers, respectively,

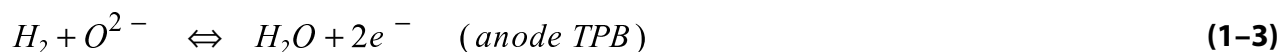


Electrons produced in the anode travel through an external circuit to the cathode, while protons (H^+) travel through the membrane from the anode TPB to the cathode TPB, thus forming an electrical circuit.

As more and more water is generated at the cathode, due to both osmotic drag and electrochemical reactions, water vapor pressure exceeds saturation pressure forming liquid water. The formation and transport of liquid water in the cathode is an important feature that can strongly influence cell performance of a PEMFC.

1.1.2. Introduction to SOFC

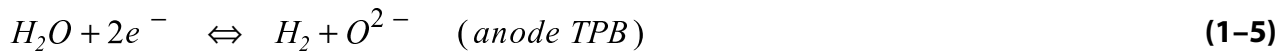
Unlike the low-temperature PEMFC, solid oxide fuel cells operate at very high temperatures ($\sim 1000^\circ\text{C}$). Hydrogen is supplied at the anode, either directly or through internal reforming of another hydrocarbon fuel, and air is supplied at the cathode. The following electrochemical reactions take place in the anode and cathode triple phase boundary layers, respectively,



In the cathode TPB, oxygen is reduced to oxygen ions which are then conducted through the ceramic electrolyte to the anode TPB where they react with hydrogen to form water and release electrons. The electrons travel through an external circuit to a load and then back to the cathode to close the circuit.

1.1.3. Introduction to Electrolysis

There has been an increasingly strong need for the large-scale production of hydrogen as a secondary energy carrier for the non-electrical market (such as PEMFC and SOFC applications). One of the cleaner and more efficient methods of producing hydrogen is to use high-temperature electrolysis to split water molecules. Therefore, electrolysis is essentially a reversed fuel cell process. Power is supplied to an electrolyzer to convert water vapor into hydrogen and oxygen. Water vapor is fed through the anode electrodes to the active electrolyte region. Once power is supplied to the anode electrode, the following electrochemical reactions take place:



In electrolysis, the activation overpotentials have the opposite sign of what is used in fuel cells. This means that the cell voltage is higher than the open circuit voltage, since power is added to overcome the activation overpotentials. The ionic conductivity in the electrolyte is typically a function of temperature, such as in the case of SOFC. And it is pointed out here that, for an electrolyzer, high thermodynamic efficiency can be achieved only at a high operating temperature ($> 500^\circ \text{C}$). Because of this, the flow field is in vapor phase only and is handled as such within ANSYS FLUENT.

1.2. Electrochemistry Modeling

At the center of the electrochemistry is the computation of the rates of the anodic and cathodic reactions. The electrochemistry model adopted in ANSYS FLUENT is the one that has been used by other groups ([3] (p. 79), [4] (p. 79), and [9] (p. 79)).

The driving force behind these reactions is the surface overpotential: the difference between the phase potential of the solid and the phase potential of the electrolyte/membrane. Therefore, two potential equations are solved for in the Fuel Cell and Electrolysis Model: one potential equation (*Equation 1-7* (p. 4)) accounts for the electron transport e^- through the solid conductive materials (i.e., the current collectors and solid grids of the porous media); the other potential equation (*Equation 1-8* (p. 4)) represents the protonic (i.e., ionic) transport of H^+ or O^{2-} . The two potential equations are as follows:

$$\nabla \cdot (\sigma_{sol} \nabla \phi_{sol}) + R_{sol} = 0 \quad (1-7)$$

$$\nabla \cdot (\sigma_{mem} \nabla \phi_{mem}) + R_{mem} = 0 \quad (1-8)$$

where

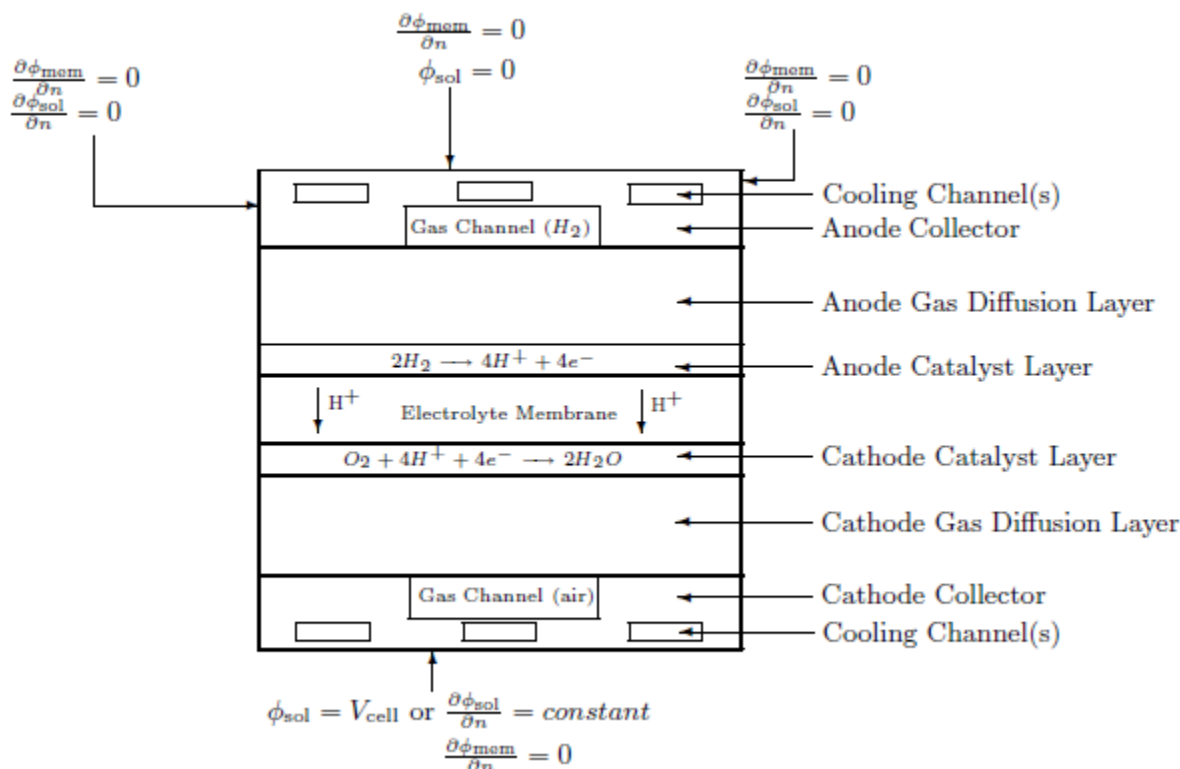
σ = electrical conductivity (1/ohm-m)

ϕ = electric potential (volts)

R = volumetric transfer current (A/m^3)

The following figure illustrates the boundary conditions that are used to solve for ϕ_{sol} and ϕ_{mem} .

Figure 1.2 Boundary Conditions for the Electric Potential (Solid and Membrane) and — PEMFC Used as an Example



There are two types of external boundaries: those that have an electrical current passing through them, and those that do not.

As no ionic current leaves the fuel cell through any external boundary, there is a zero flux boundary condition for the membrane phase potential, ϕ_{mem} , on all outside boundaries.

For the solid phase potential, ϕ_{sol} , there are external boundaries on the anode and the cathode side that are in contact with the external electric circuit and only through these boundaries passes the electrical current generated in the fuel cell. On all other external boundaries there is a zero flux boundary condition for ϕ_{sol} .

On the external contact boundaries, we recommend to prescribe fixed values for ϕ_{sol} (potentiostatic boundary conditions). If the anode side is set to zero, the (positive) value prescribed on the cathode side is the cell voltage. Specifying a constant flux (say on the cathode side) means to specify galvanostatic boundary conditions.

The transfer currents, or the source terms in [Equation 1-7](#) (p. 4) and [Equation 1-8](#) (p. 4), are non-zero only inside the catalyst layers and are computed as:

- For the solid phase, $R_{sol} = -R_{an} (< 0)$ on the anode side and $R_{sol} = +R_{cat} (> 0)$ on the cathode side.
- For the membrane phase, $R_{mem} = +R_{an} (> 0)$ on the anode side and $R_{mem} = -R_{cat} (< 0)$ on the cathode side.

The source terms in *Equation 1-7* (p. 4) and *Equation 1-8* (p. 4) (A/m^3), have the following general definitions:

$$R_{an} = \left(\zeta_{an} j_{an}^{ref} \right) \left(\frac{[A]}{[A]_{ref}} \right)^{\gamma_{an}} \left(e^{\alpha_{an} F \eta_{an} / RT} - e^{-\alpha_{cat} F \eta_{an} / RT} \right) \quad (1-9)$$

$$R_{cat} = \left(\zeta_{cat} j_{cat}^{ref} \right) \left(\frac{[C]}{[C]_{ref}} \right)^{\gamma_{cat}} \left(-e^{+\alpha_{an} F \eta_{cat} / RT} + e^{-\alpha_{cat} F \eta_{cat} / RT} \right) \quad (1-10)$$

where

j^{ref} = reference exchange current density per active surface area (A/m^2)

ζ = specific active surface area (1/m)

$[]$, $[]_{ref}$ = local species concentration, reference value ($kmol/m^3$)

γ = concentration dependence

α = transfer coefficient (dimensionless)

F = Faraday constant (9.65×10^7 C/kmol)

The above equation is the general formulation of the Butler-Volmer function. A simplification to this is the Tafel formulation that reads,

$$R_{an} = \left(\zeta_{an} j_{an}^{ref} \right) \left(\frac{[A]}{[A]_{ref}} \right)^{\gamma_{an}} \left(e^{\alpha_{an} F \eta_{an} / RT} \right) \quad (1-11)$$

$$R_{cat} = \left(\zeta_{cat} j_{cat}^{ref} \right) \left(\frac{[C]}{[C]_{ref}} \right)^{\gamma_{cat}} \left(e^{-\alpha_{cat} F \eta_{cat} / RT} \right) \quad (1-12)$$

By default, the Butler-Volmer function is used in the ANSYS FLUENT Fuel Cell and Electrolysis Model to compute the transfer currents inside the catalyst layers.

In *Equation 1-9* (p. 6) through *Equation 1-12* (p. 6), $[A]$ and $[C]$ represent the molar concentration of the species upon which the anode and cathode reaction rates depend, respectively. For PEMFC and SOFC, A represents H_2 and C represents O_2 . For Electrolysis, A represents H_2O and C is 1.0 (which indicates that the cathode reaction does not depend on any species concentration).

The driving force for the kinetics is the local surface overpotential, η , also known as the *activation loss*. It is generally the difference between the solid and membrane potentials, ϕ_{sol} and ϕ_{mem} .

The gain in electrical potential from crossing from the anode to the cathode side can then be taken into account by subtracting the open-circuit voltage V_{oc} on the cathode side.

$$\eta_{an} = \phi_{sol} - \phi_{mem} \quad (1-13)$$

$$\eta_{cat} = \phi_{sol} - \phi_{mem} - V_{oc} \quad (1-14)$$

From [Equation 1-7 \(p. 4\)](#) through [Equation 1-14 \(p. 7\)](#), the two potential fields can be obtained.

1.3. Current and Mass Conservation

Species volumetric source terms (kg/m^3 -s) in the triple-phase boundaries due to electrochemical reactions for the PEMFC, SOFC, and Electrolysis, respectively, are:

For PEMFC:

$$S_{H_2} = -\frac{M_{w,H_2}}{2F}R_{an} < 0 \quad (1-15)$$

$$S_{O_2} = -\frac{M_{w,O_2}}{4F}R_{cat} < 0 \quad (1-16)$$

$$S_{H_2O} = \frac{M_{w,H_2O}}{2F}R_{cat} > 0 \quad (1-17)$$

For SOFC:

$$S_{H_2} = -\frac{M_{w,H_2}}{2F}R_{an} < 0 \quad (1-18)$$

$$S_{O_2} = -\frac{M_{w,O_2}}{4F}R_{cat} < 0 \quad (1-19)$$

$$S_{H_20} = \frac{M_{w,H_20}}{2F}R_{an} > 0 \quad (1-20)$$

For Electrolysis:

$$S_{H_2} = \frac{M_{w,H_2}}{2F}R_{an} > 0 \quad (1-21)$$

$$S_{O_2} = \frac{M_{w,O_2}}{4F}R_{cat} > 0 \quad (1-22)$$

$$S_{H_20} = -\frac{M_{w,H_20}}{2F}R_{an} < 0 \quad (1-23)$$

Since the total electrical current produced in the cathode and the anode TPBs, respectively, is the same, we have the following equation for current conservation:

$$\int_{anode} R_{an}dV = \int_{cathode} R_{cat}dV \quad (1-24)$$

1.4. Heat Source

Additional volumetric sources to the thermal energy equation are present because not all chemical energy released in the electrochemical reactions can be converted to electrical work due to irreversibilities of the processes. The total source that goes to the thermal energy equation (i.e., enthalpy) is:

$$S_h = h_{react} - R_{an,cat}\eta_{an,cat} + I^2R_{ohm} + h_L \quad (1-25)$$

where h_{react} is the net enthalpy change due to the electrochemical reactions, $R_{an,cat}\eta_{an,cat}$ is the product of the transfer current and the overpotential in the anode or the cathode TPB, R_{ohm} is the

ohmic resistivity of the conducting media, and h_L is the enthalpy change due to condensation/vaporization of water.

1.5. Liquid Water Formation, Transport, and its Effects (PEMFC Only)

Since PEM fuel cells operate under relatively low temperature ($< 100^\circ\text{C}$), the water vapor may condense to liquid water, especially at high current densities. While the existence of the liquid water keeps the membrane hydrated, it also blocks the gas diffusion passage, reduces the diffusion rate and the effective reacting surface area and hence the cell performance. To model the formation and transport of liquid water, ANSYS FLUENT uses a saturation model based on [7] (p. 79), [5] (p. 79). In this approach, the liquid water formation and transport is governed by the following conservation equation for the volume fraction of liquid water, s , or the water saturation,

$$\frac{\partial (\varepsilon \rho_l s)}{\partial t} + \nabla \cdot (\rho_l \bar{V}_l s) = r_w \quad (1-26)$$

where the subscript l stands for liquid water, and r_w is the condensation rate that is modeled as,

$$r_w = c_r \max \left(\left[(1-s) \frac{P_{wv} - P_{sat}}{RT} M_{w,H_2O} \right], \left[-s \rho_l \right] \right) \quad (1-27)$$

where $-r_w$ is added to the water vapor equation, as well as the pressure correction (mass source). This term is not applied inside the membrane. The condensation rate constant is hardwired to $c_r = 100s^{-1}$. It is assumed that the liquid velocity, \bar{V}_l , is equivalent to the gas velocity inside the gas channel (i.e., a fine mist). Inside the highly-resistant porous zones, the use of the capillary diffusion term allows us to replace the convective term in Equation 1-26 (p. 9):

$$\frac{\partial (\varepsilon \rho_l s)}{\partial t} + \nabla \cdot \left[\rho_l \frac{K s^3}{\mu_l} \frac{dp_c}{ds} \nabla s \right] = r_w \quad (1-28)$$

Depending on the wetting phase, the capillary pressure is computed as a function of s (the Leverett function),

$$p_c = \begin{cases} \frac{\sigma \cos \theta_c}{\left(\frac{K}{\varepsilon}\right)^{0.5}} (1.417(1-s) - 2.12(1-s)^2 + 1.263(1-s)^3) & \theta_c < 90^\circ \\ \frac{\sigma \cos \theta_c}{\left(\frac{K}{\varepsilon}\right)^{0.5}} (1.417s - 2.12s^2 + 1.263s^3) & \theta_c > 90^\circ \end{cases} \quad (1-29)$$

where ε is the porosity, σ is the surface tension (N/m), θ_c is the contact angle and K the absolute permeability.

Equation 1–26 (p. 9) models various physical processes such as condensation, vaporization, capillary diffusion, and surface tension.

The clogging of the porous media and the flooding of the reaction surface are modeled by multiplying the porosity and the active surface area by $(1 - s)$, respectively.

1.6. Properties

- Gas Phase Species Diffusivity

Gas phase species diffusivities can be computed either by using the dilute approximation method or by using the full multicomponent method. With the dilute approximation method, we have

$$D_i = \varepsilon^{1.5} (1 - s)^{r_s} D_i^0 \left(\frac{P_0}{P} \right)^{\gamma_p} \left(\frac{T}{T_0} \right)^{\gamma_t} \quad (1-30)$$

where D_i^0 is the mass diffusivity of species i at reference temperature and pressure (P_0, T_0) [9] (p. 79). These reference values and the exponents (γ_p, γ_t) as well as the exponent of pore blockage (r_s) are defined in the Fuel Cell and Electrolysis user defined functions (UDF) as,

$$\begin{aligned} p_0 &= 101325 \text{ N/m}^2 \\ T_0 &= 300 \text{ K} \\ \gamma_p &= 1.0 \\ \gamma_t &= 1.5 \\ r_s &= 2.5 \end{aligned}$$

In addition to **Equation 1–30 (p. 10)**, the ANSYS FLUENT Fuel Cell and Electrolysis Model also contains a method to compute the gas phase species diffusion (a full multicomponent diffusion method with corrections to account for the porous media tortuosity):

$$D_{eff}^{ij} = \varepsilon^{1.5} D^{ij} \quad (1-31)$$

where D_{eff}^{ij} is the effective gas species diffusivity, ε is the porosity of the porous medium, and D^{ij} is the gas species mass diffusivity computed by the full multicomponent diffusion method (as described in **Full Multicomponent Diffusion** in the separate ANSYS FLUENT **User's Guide**). Please note that $\varepsilon^{1.5}$ in **Equation 1–31 (p. 10)** is used to model the effect of tortuosity. While this is implemented as the default method in the Fuel Cell and Electrolysis Model, you can overwrite it with your own correction methods by using the user-modifiable routines that are provided.

Properties such as electrolyte phase electrical conductivity, water diffusivity, and the osmotic drag coefficient are evaluated as functions of the water content, using various correlations as suggested by [8] (p. 79). To capture the relevant physics of the problem, various properties of the membrane are incorporated into the model as default options. You can, however, directly incorporate your own formulations and data for these properties by editing the functions defined in the provided

source code file called `pem_user.c` and compiling the code yourself. For more information, see [User-Accessible Functions](#) (p. 39).

- Electrolyte Phase (Ionic) Conductivity

For SOFC and high-temperature Electrolysis, the ionic conductivity in the electrolyte is modeled as a function of temperature, and, by default, is defined as:

$$\sigma_{ionic} = \frac{100}{0.3685 + 0.002838e^{(10300/T)}} \quad (1-32)$$

This is valid for temperatures ranging from 1073 K to 1373 K. You can implement your own models in the user-customizable UDF function `Electrolyte_Conductivity` in `pem_user.c`.

For PEMFC, the electrolyte (also called the membrane) phase conductivity is modeled as:

$$\sigma_{mem} = \beta (0.514\lambda - 0.326) \omega e^{1268 \left(\frac{1}{303} - \frac{1}{T} \right)} \quad (1-33)$$

where λ is the water content. Two model constants, β and ω are introduced in ANSYS FLUENT for generality. [Equation 1-33](#) (p. 11) becomes the original correlation from [8] (p. 79) when $\beta = \omega = 1$.

- Osmotic Drag Coefficient (PEMFC)

$$n_d = 2.5 \frac{\lambda}{22} \quad (1-34)$$

- Back Diffusion Flux (PEMFC)

$$J_w^{diff} = - \frac{\rho_m}{M_m} M_{h_2O} D_l \nabla \lambda \quad (1-35)$$

where ρ_m and M_m are the density and the equivalent weight of the dry membrane, respectively.

- Membrane Water Diffusivity (PEMFC)

$$D_l = f(\lambda) e^{2416 \left(\frac{1}{303} - \frac{1}{T} \right)} \quad (1-36)$$

- Water Content (PEMFC)

The water content, λ , that appears in the preceding property computations are obtained using Springer et al's correlation [8] (p. 79),

$$\begin{aligned}\lambda &= 0.043 + 17.18a - 39.85a^2 + 36a^3 \quad (a < 1) \\ \lambda &= 14 + 1.4(a - 1) \quad (a > 1)\end{aligned}\tag{1-37}$$

here a is the water activity that is defined as,

$$a = \frac{P_{wv}}{P_{sat}} + 2s\tag{1-38}$$

- Water Vapor Pressure

The water vapor pressure is computed based upon the vapor molar fraction and the local pressure,

$$P_{wv} = x_{H_2O}P\tag{1-39}$$

- Saturation Pressure

The saturation pressure is calculated, in terms of *atm*, as,

$$\begin{aligned}\log_{10}P_{sat} &= -2.1794 + 0.02953(T - 273.17) \\ &\quad - 9.1837 \times 10^{-5}(T - 273.17)^2 \\ &\quad + 1.4454 \times 10^{-7}(T - 273.17)^3\end{aligned}\tag{1-40}$$

It is noted here that in [8] (p. 79), water activity is defined on the basis of total water or super-saturated water vapor. With phase change being invoked in the present two-phase model, $2s$ is added to the original formulation as suggested by [2] (p. 79).

1.7. Transient Simulations

Dynamics response to changes in operating conditions as a function of time can be modeled using the Fuel Cell and Electrolysis module. For example, a change in the cell voltage or current density, or inlet mass flow rates at the anode and/or the cathode. The procedure for setting up and solving transient Fuel Cell and Electrolysis problems are the same as that used for a normal ANSYS FLUENT transient problem as discussed in the ANSYS FLUENT [User's Guide](#).

Assuming that the time scales associated with the electric fields are much smaller than those associated with the flow and thermal fields, the steady-state equations are retained for the two electric potentials, (i.e., [Equation 1-7](#) (p. 4) and [Equation 1-8](#) (p. 4)). Transient terms in all other equations such as momentum transport, energy transport, species transport, liquid water transport, and membrane water content equations are activated.

1.8. Specific Leakage Current (Cross-Over Current)

The specific leakage current, I_{leak} (A / m^3), models the effect of species cross-over from one electrode to another across the electrolyte. In addition to the source terms expressed by [Equation 1-15](#) (p. 7) through [Equation 1-23](#) (p. 8):

For PEMFC and SOFC:

$$S_{H_2} = -\frac{M_{w,H_2}}{2F}I_{leak} \quad (1-41)$$

$$S_{O_2} = -\frac{M_{w,O_2}}{4F}I_{leak} \quad (1-42)$$

$$S_{H_2O} = \frac{M_{w,H_2O}}{2F}I_{leak} > 0 \quad (1-43)$$

For Electrolysis:

$$S_{H_2} = \frac{M_{w,H_2}}{2F}I_{leak} \quad (1-44)$$

$$S_{O_2} = \frac{M_{w,O_2}}{4F}I_{leak} \quad (1-45)$$

$$S_{H_2O} = -\frac{M_{w,H_2O}}{2F}I_{leak} \quad (1-46)$$

Chapter 2: Using the Fuel Cell and Electrolysis Model

The procedure for setting up and solving fuel cell problems using the Fuel Cell and Electrolysis Model is described in detail in this chapter. Please refer to the following sections for more information:

- 2.1. Introduction
- 2.2. Geometry Definition for the Fuel Cell and Electrolysis Model
- 2.3. Installing the Fuel Cell and Electrolysis Model
- 2.4. Loading the Fuel Cell and Electrolysis Module
- 2.5. Setting Up the Fuel Cell and Electrolysis Module
- 2.6. Modeling Fuel Cells and Electrolysis
- 2.7. Modeling Current Collectors
- 2.8. Fuel Cell and Electrolysis Model Boundary Conditions
- 2.9. Solution Guidelines for the Fuel Cell and Electrolysis Model
- 2.10. Postprocessing the Fuel Cell and Electrolysis Model
- 2.11. User-Accessible Functions
- 2.12. Using the Fuel Cell and Electrolysis Text User Interface

2.1. Introduction

The ANSYS FLUENT Fuel Cell and Electrolysis Model comprises several user-defined functions (UDFs) and a graphical user interface. The potential fields are solved as user-defined scalars. The liquid water saturation, s , and the water content, λ , are also solved as user-defined scalars. The electrochemical reactions occurring on the catalyst are modeled through various source terms while other model parameters are handled through the user interface. The Fuel Cell and Electrolysis Model can be used in parallel ANSYS FLUENT as well.

2.2. Geometry Definition for the Fuel Cell and Electrolysis Model

Due to the fact that there are a number of different physical zones associated with the fuel cell, the following regions must be present in the fuel cell mesh:

- Anode flow channel
- Anode gas diffusion layer
- Anode catalyst layer
- Membrane layer
- Cathode catalyst layer
- Cathode gas diffusion layer
- Cathode flow channel

The following zones have to be identified, if present in the fuel cell mesh:

- Anode current collector
- Cathode current collector
- Coolant channel

2.3. Installing the Fuel Cell and Electrolysis Model

The Fuel Cell and Electrolysis Model is provided as an add-on module with the standard ANSYS FLUENT licensed software. A special license is required to use the Fuel Cell and Electrolysis Model. The module is installed with the standard installation of ANSYS FLUENT in a directory called `addons/fuelcells` in your installation area. The Fuel Cell and Electrolysis Model consists of a UDF library and a pre-compiled scheme library, which needs to be loaded and activated before calculations can be performed.

2.4. Loading the Fuel Cell and Electrolysis Module

The Fuel Cell and Electrolysis module is loaded into ANSYS FLUENT through the text user interface (TUI). The module can only be loaded after a valid ANSYS FLUENT case file has been set or read. The text command to load the add-on module is

```
define → models → add-on-module
```

A list of ANSYS FLUENT add-on modules is displayed:

```
> /define/models/addon-module
FLUENT  Addon Modules:
  0. None
  1. MHD Model
  2. Fiber Model
  3. Fuel Cell and Electrolysis Model
  4. SOFC Model with Unresolved Electrolyte
  5. Population Balance Model
  6. Adjoint Solver
  7. Battery Module
Enter Module Number: [0] 3
```

Select the Fuel Cell and Electrolysis Model by entering the module number 3. During the loading process, a scheme library (containing the graphical and text user interface) and a UDF library (containing a set of user defined functions) are loaded into ANSYS FLUENT.

2.5. Setting Up the Fuel Cell and Electrolysis Module

The Fuel Cell and Electrolysis Model can be used to model polymer electrolyte membrane fuel cells (PEMFC), solid oxide fuel cells (SOFC), and the process of high-temperature electrolysis. The following describes an overview of the procedure required in order to use the Fuel Cell and Electrolysis Model in ANSYS FLUENT.

1. Start ANSYS FLUENT.
2. Read the case file.
3. Scale the grid, if necessary.
4. Use the **Fuel Cell and Electrolysis Models** dialog box to define the fuel cell model parameters.
5. Define material properties.
6. Set the operating conditions.
7. Set the boundary conditions.
8. Start the calculations.
9. Save the case and data files.
10. Process your results.

Important

The **Fuel Cell and Electrolysis Models** dialog box greatly simplifies the input of parameters and boundary conditions, but it does not replace the boundary conditions interface. Therefore, it is a good policy to start the setup with the **Fuel Cell and Electrolysis Models** dialog box and do the finishing steps for boundary conditions afterwards.

Important

Note that the majority of this chapter describes how to set up the ANSYS FLUENT Fuel Cell and Electrolysis Model using the graphical user interface. You can also perform various tasks using the text user interface. For more information, see [Using the Fuel Cell and Electrolysis Text User Interface](#) (p. 43).

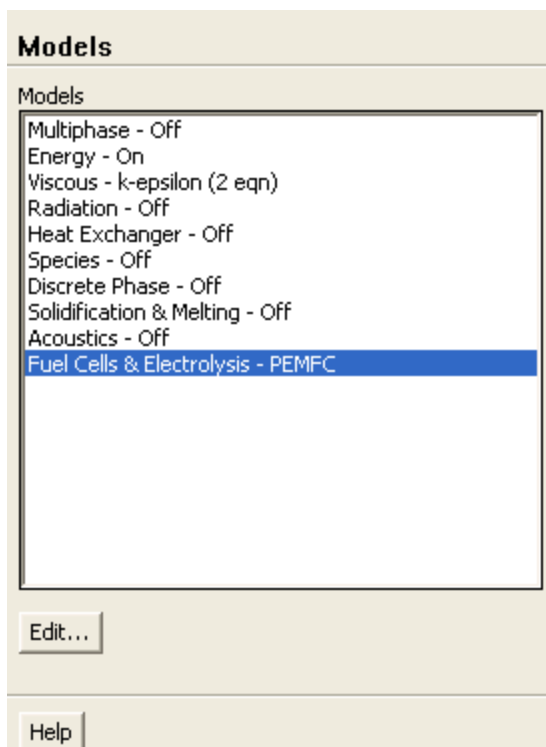
2.6. Modeling Fuel Cells and Electrolysis

Once the module has been loaded, in order to set fuel cell model parameters and assign properties to the relevant regions in your fuel cell, you need to access the fuel cell graphical user interface (the **Fuel Cell and Electrolysis Models** dialog box).

To open the **Fuel Cell and Electrolysis Models** dialog box, select **Models** under **Problem Setup** in the navigation pane to display the **Models** task page. In the **Models** task page, select the **Fuel Cells and Electrolysis** option in the **Models** list and click the **Edit...** button.

 **Models** → **Fuel Cells and Electrolysis** → **Edit...**

Figure 2.1 Opening the Fuel Cell and Electrolysis Models Dialog Box in the Models Task Page



By default, the PEMFC model is already enabled, however, you can also choose the SOFC or the Electrolysis models.

Using the **Fuel Cell and Electrolysis Models** dialog box, you can identify the relevant zones for the current collectors, flow channels, gas diffusion layers, catalyst layers, and the membrane/electrolyte. You can specify the following inputs using the **Fuel Cell and Electrolysis Models** dialog box. Optional inputs are indicated as such.

1. Enable the appropriate type of fuel cell model, either **PEMFC**, **SOFC**, or **Electrolysis**.
2. Enable either the single-phase or the multi-phase fuel cell model (if **PEMFC** is selected).
3. Set the appropriate options for the fuel cell model (optional).
4. Set the various parameters for the fuel cell model.
5. Select the appropriate zones and specify the properties on the anode side.
6. Select the appropriate zones and specify the properties of the membrane/electrolyte.
7. Select the appropriate zones and specify the properties on the cathode side.
8. Provide input for advanced features such as contact resistivities, coolant channel properties, or stack management settings (optional).
9. Set solution controls such as under-relaxation factors (optional).
10. Provide input to assist reporting (optional).

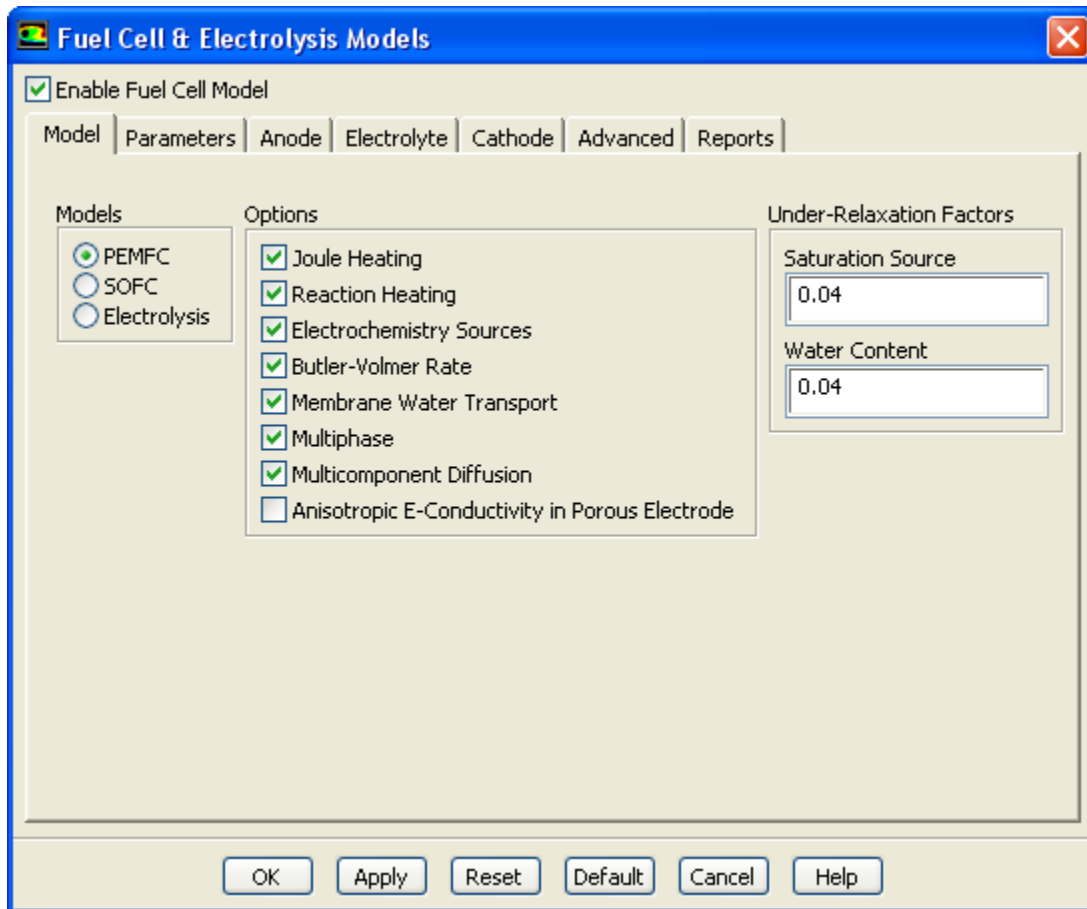
Please refer to the following sections for more information:

- [2.6.1. Specifying Model Options](#)
- [2.6.2. Specifying Model Parameters](#)
- [2.6.3. Specifying Anode Properties](#)
- [2.6.4. Specifying Electrolyte/Membrane Properties](#)
- [2.6.5. Specifying Cathode Properties](#)
- [2.6.6. Setting Advanced Properties](#)
- [2.6.7. Reporting on the Solution](#)

2.6.1. Specifying Model Options

The **Model** tab of the **Fuel Cell and Electrolysis Models** dialog box allows you to turn on or off various options when solving a fuel cell problem. To model polymer electrolyte membrane fuel cells, enable the **PEMFC** option in the **Model** tab. Likewise, to model solid oxide fuel cells, enable the **SOFC** option in the **Model** tab. Finally, to model electrolysis, enable the **Electrolysis** option in the **Model** tab.

Figure 2.2 The Model Options in the Fuel Cell and Electrolysis Models Dialog Box - PEMFC Enabled



Several fuel cell model options are available in the **Model** tab of the **Fuel Cell and Electrolysis Models** dialog box including:

- The **Joule Heating** option takes into account ohmic heating. This option includes the I^2R term in the energy source term from [Equation 1–25 \(p. 8\)](#) in the calculations.
- The **Reaction Heating** option takes into account the heat generated by the electrochemical reactions, which includes the $h_{reaction}$ term, and the product of transfer current and the over-potentials in the energy source term from [Equation 1–25 \(p. 8\)](#) in the calculations.
- The **Electrochemistry Sources** option allows the Fuel Cell and Electrolysis Model to take electrochemistry effects into account. If you are only interested in the basic flow field throughout the fuel cell, you can turn off the **Electrochemistry Sources** option in order to suppress most effects of the Fuel Cell and Electrolysis Model. To turn off all effects of the Fuel Cell and Electrolysis Model, you should also turn off the **Membrane Water Transport** and **Multiphase** options.
- The **Butler-Volmer Rate** option (the default) is used to compute the transfer currents inside the catalyst layers. If this option is turned off, the Tafel approximation ([Equation 1–12 \(p. 6\)](#)) is used.
- The **Membrane Water Transport** option takes into account the transport of water across the membrane. This option is only available for the **PEMFC** model.
- The **Multiphase** option takes into account multiphase calculations. Use this option if you are solving for approximate liquid transport in the gas diffusion layer of the fuel cell. (PEMFC only)

- The **Multicomponent Diffusion** option is used to compute the gas species mass diffusivity using the full multicomponent diffusion method as described in [Equation 1–31 \(p. 10\)](#), as opposed to the default option that uses [Equation 1–30 \(p. 10\)](#).
- The **Anisotropic E-Conductivity in Porous Electrode** option is used to model the typically non-isotropic electrical conductivity. It is applicable only for porous electrodes (gas diffusion layers).

Due to the fibrous structure of the porous material that is used for the electrodes (or gas diffusion layer), the electrical conductivity is typically non-isotropic, with the cross-plane components being orders of magnitude smaller than the in-plane components. This can be modeled using the **Anisotropic E-Conductivity in Porous Electrode** setting. When this option is enabled, the **Electrical Conductivity** for the solid material used in the electrolyte is no longer used. Instead, you need to specify, for this solid material, the electrical conductivity by choosing one of the three non-isotropic options for the UDS diffusivity (UDS-0). The three options are: **anisotropic**; **orthotropic**; and **cyl-orthotropic**. For more information about these UDS Diffusivity options, refer to the ANSYS FLUENT [User's Guide](#).

For example, to use this feature, perform the following steps:

- Select the **Anisotropic E-Conductivity in Porous Electrode** option in the **Model** tab of the **Fuel Cell and Electrolysis Models** dialog box.
- In the **Materials** task page, select `defined-per-uds` for **UDS Diffusivity** for the solid material that is to be used for the porous electrode.
- Select one of the three options for UDS-0: **anisotropic**; **orthotropic**; or **cyl-orthotropic** and set the appropriate values.

Important

Note that, in this case, the **Electrical Conductivity** for *this* solid material is ignored.

- For PEMFC problems, you can use the **Under-Relaxation Factors** fields to influence the solution process.

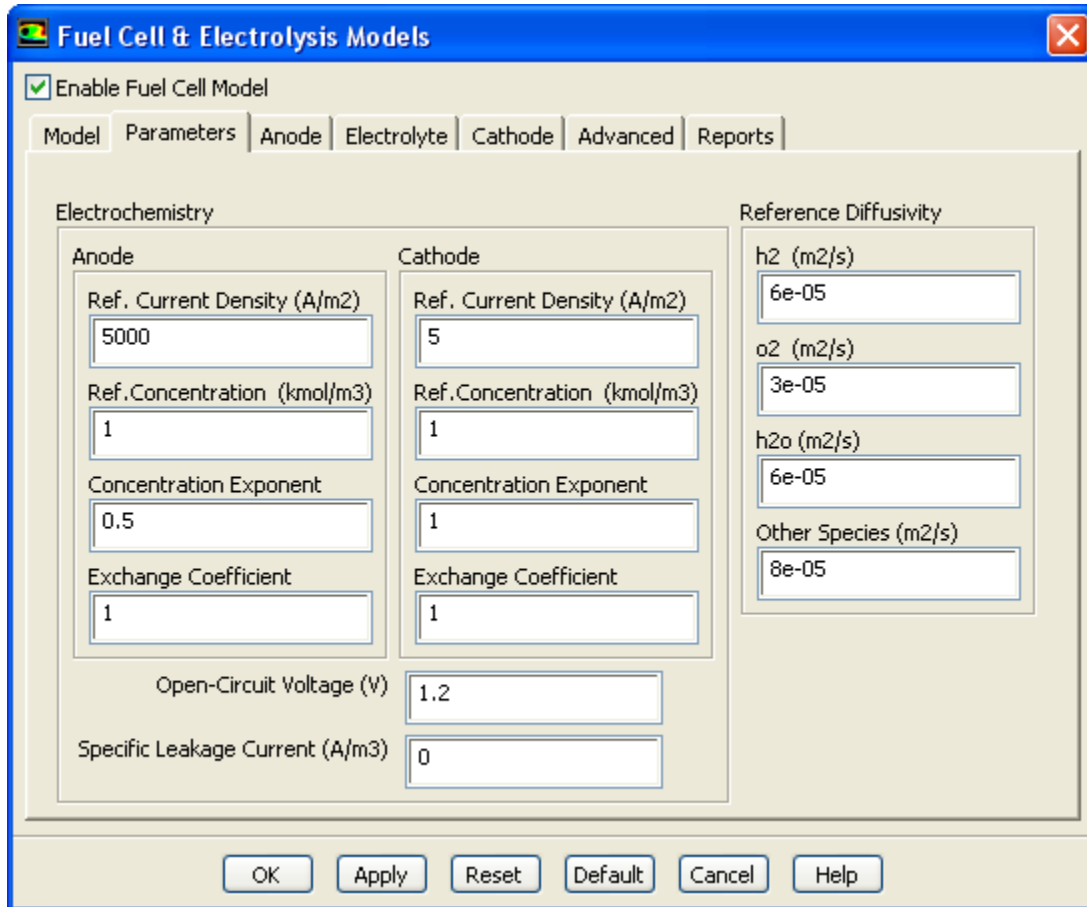
The saturation source term r_w in [Equation 1–27 \(p. 9\)](#) usually requires under-relaxation. You can change the default value for the under-relaxation factor by changing the value for **Saturation Source**.

The water content, λ , in [Equation 1–37 \(p. 12\)](#) also may need under-relaxation. You can change the default value for the under-relaxation factor by changing the value for **Water Content**.

Nearly all options are turned on by default. You may wish to override the default values, depending on the problem you wish to model. For instance, if you are not concerned with the heat generated due to chemical reaction, then you may want to turn off the **Reaction Heating** option.

2.6.2. Specifying Model Parameters

You can use the **Parameters** tab of the **Fuel Cell and Electrolysis Models** dialog box to specify the electrochemistry parameters for the Fuel Cell and Electrolysis Model, reference diffusivities for the reactants, among other model parameters.

Figure 2.3 The Parameters Tab of the Fuel Cell and Electrolysis Models Dialog Box

There are various parameters under **Electrochemistry** in the **Fuel Cell and Electrolysis Models** dialog box. For both the anode and the cathode, you can also set the following parameters or leave the default values.

- The **Ref. Current Density** corresponds to j_{an}^{ref} and j_{cat}^{ref} , the reference exchange current density from [Equation 1–9 \(p. 6\)](#) and [Equation 1–10 \(p. 6\)](#).
- The **Ref. Concentration** corresponds to the reference concentration ($[H_2]_{ref}$ and $[O_2]_{ref}$) with units of $1 \text{ kmol}/\text{m}^3$ (see [Equation 1–9 \(p. 6\)](#) and [Equation 1–10 \(p. 6\)](#)).
- The **Concentration Exponent** corresponds to γ , the concentration dependence from [Equation 1–9 \(p. 6\)](#).
- The **Exchange Coefficient** corresponds to α , the transfer coefficient from [Equation 1–9 \(p. 6\)](#).
- The **Open-Circuit Voltage** corresponds to V_{oc} in [Equation 1–14 \(p. 7\)](#).
- The **Specific Leakage Current** corresponds to I_{leak} in [Equation 1–41 \(p. 13\)](#) through [Equation 1–46 \(p. 13\)](#).

This is the total amount of transfer current (A/m^3) due to fuel oxidant cross-over (leakage through the electrolyte). When this happens, the fuel cell generates less current especially for cases with low values of fuel or air utilization. In addition to the constant value you can specify in the **Electrochemistry** tab, you can also specify the specific leakage current through the user-defined function `Leakge_Current()`. For more information, see [User-Accessible Functions \(p. 39\)](#).

Moreover, the following parameters can also be set here:

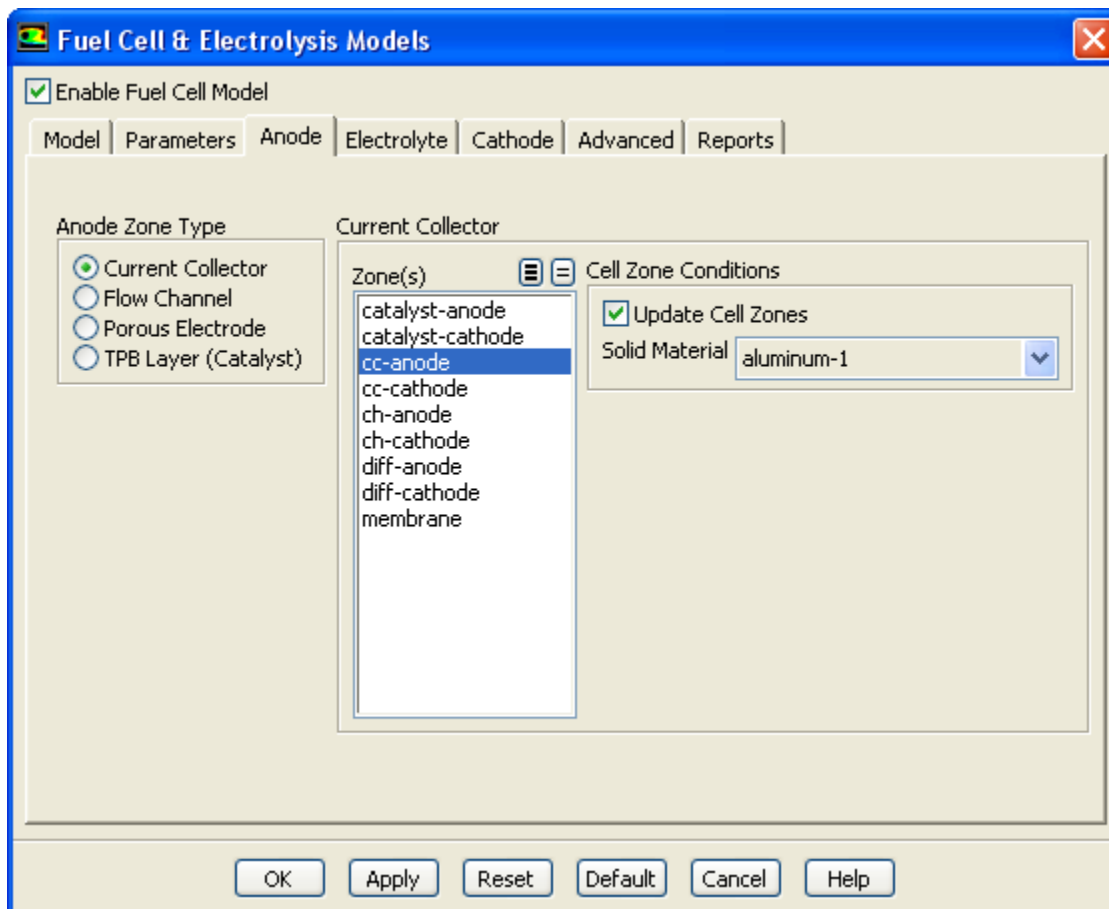
- The **Reference Diffusivities** correspond to D_i^0 from *Equation 1–30* (p. 10), the species mass diffusivity. These are not be required if the **Multicomponent Diffusion** option is enabled in the **Model** tab.
- The **Saturation Exponent for Pore Blockage** corresponds to r_3 from *Equation 1–30* (p. 10) for multiphase PEMFC calculations.

2.6.3. Specifying Anode Properties

You can use the **Anode** tab of the **Fuel Cell and Electrolysis Models** dialog box to specify zones and properties of the current collector, the flow channel, the diffusion layer, and the catalyst layer for the anode portion of the fuel cell.

2.6.3.1. Specifying Current Collector Properties for the Anode

Figure 2.4 The Anode Tab of the Fuel Cell and Electrolysis Models Dialog Box With Current Collector Selected

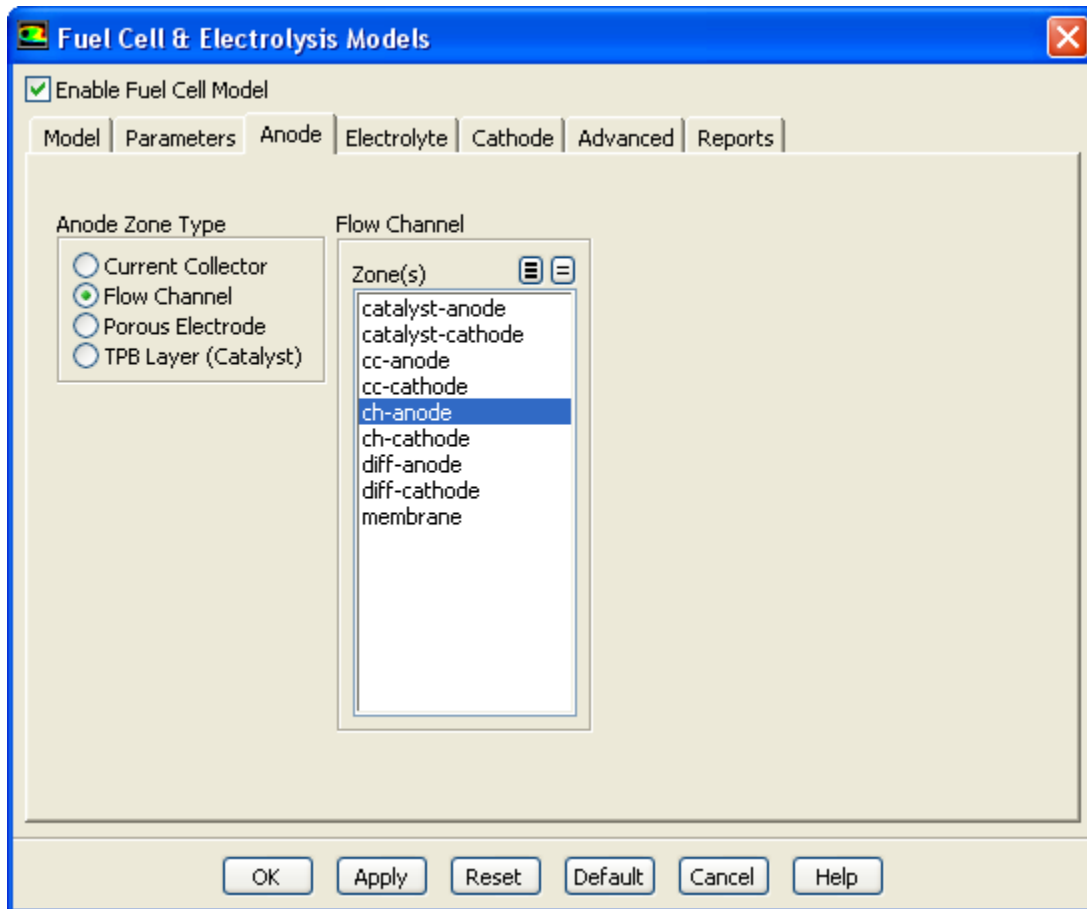


1. Select the **Anode** tab of the **Fuel Cell and Electrolysis Models** dialog box.
2. Select **Current Collector** under **Anode Zone Type**.
3. Select a corresponding zone from the **Zone(s)** list. If you are modeling a fuel cell stack, then you must pick *all* zones of a particular type as a group.
4. Select a **Solid Material** from the corresponding drop-down list. Solid materials can be customized using the **Create/Edit Materials** dialog box. Note that for the **Electrical Conductivity**, you can only

choose a constant value in the **Create/Edit Materials** dialog box. The solid electrical conductivity value is the diffusivity of the solid phase potential in the solid zones.

2.6.3.2. Specifying Flow Channel Properties for the Anode

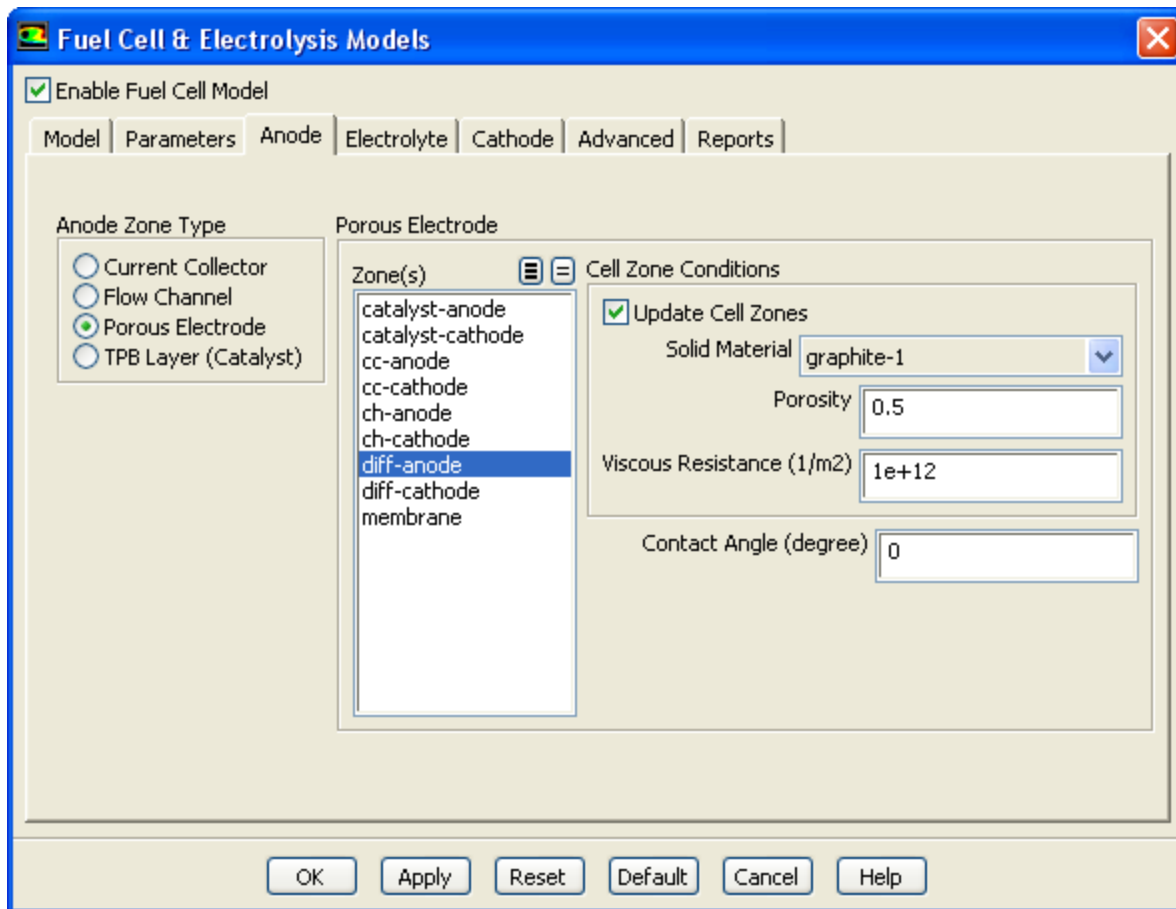
Figure 2.5 The Anode Tab of the Fuel Cell and Electrolysis Models Dialog Box With Flow Channel Selected



1. Select the **Anode** tab of the **Fuel Cell and Electrolysis Models** dialog box.
2. Select **Flow Channel** under **Anode Zone Type**.
3. Select a corresponding zone from the **Zone(s)** list.

2.6.3.3. Specifying Porous Electrode Properties for the Anode

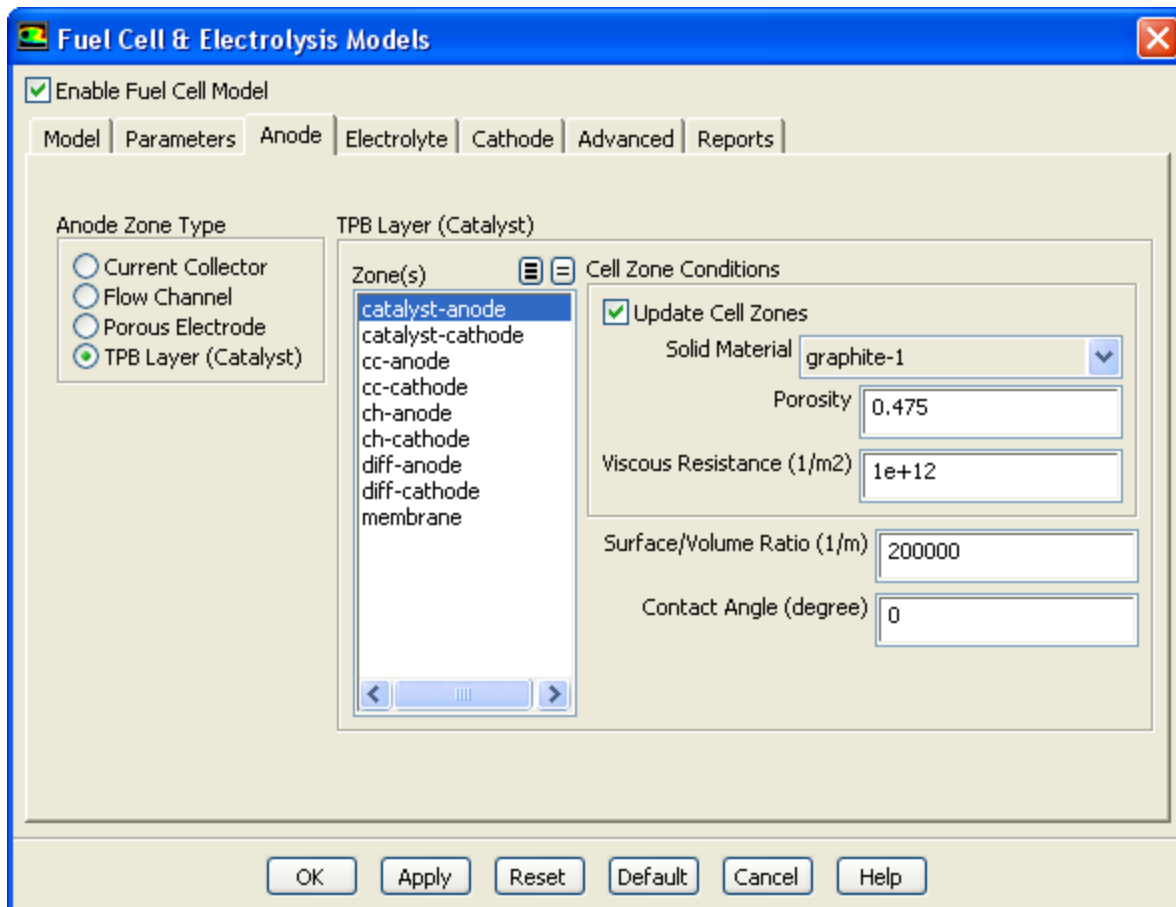
Figure 2.6 The Anode Tab of the Fuel Cell and Electrolysis Models Dialog Box With Porous Electrode Selected



1. Select the **Anode** tab of the **Fuel Cell and Electrolysis Models** dialog box.
2. Select **Porous Electrode** under **Anode Zone Type**.
3. Select a corresponding zone from the **Zone(s)** list. If you are modeling a fuel cell stack, then you must pick *all* zones of a particular type as a group.
4. Select a **Solid Material** from the corresponding drop-down list. Solid materials can be customized using the **Create/Edit Materials** dialog box. Note that for the **Electrical Conductivity**, you can only choose a constant value in the **Create/Edit Materials** dialog box. The solid electrical conductivity value is the diffusivity of the solid phase potential in the solid zones.
5. Specify a value for the **Porosity**.
6. Specify a value for the **Viscous Resistance**.
7. Specify a value for the **Contact Angle** for multiphase fuel cell calculations (θ_c in *Equation 1-29* (p. 9)).

2.6.3.4. Specifying Catalyst Layer Properties for the Anode

Figure 2.7 The Anode Tab of the Fuel Cell and Electrolysis Models Dialog Box With TPB Layer - Catalyst Selected



1. Select the **Anode** tab of the **Fuel Cell and Electrolysis Models** dialog box.
2. Select **TPB Layer (Catalyst)** under **Anode Zone Type**.
3. Select a corresponding zone from the **Zone(s)** list. If you are modeling a fuel cell stack, then you must pick *all* zones of a particular type as a group.
4. Select a **Solid Material** from the corresponding drop-down list. Solid materials can be customized using the **Create/Edit Materials** dialog box. Note that for the **Electrical Conductivity**, you can only choose a constant value in the **Create/Edit Materials** dialog box. The solid electrical conductivity value is the diffusivity of the solid phase potential in the solid zones.
5. Specify a value for the **Porosity**.
6. Specify a value for the **Viscous Resistance**.
7. Specify a value for the **Surface/Volume Ratio** (the specific active surface area in [Equation 1–9](#) (p. 6)).
8. Specify a value for the **Contact Angle** for multiphase fuel cell calculations (θ_c in [Equation 1–29](#) (p. 9)).

2.6.3.5. Specifying Cell Zone Conditions for the Anode

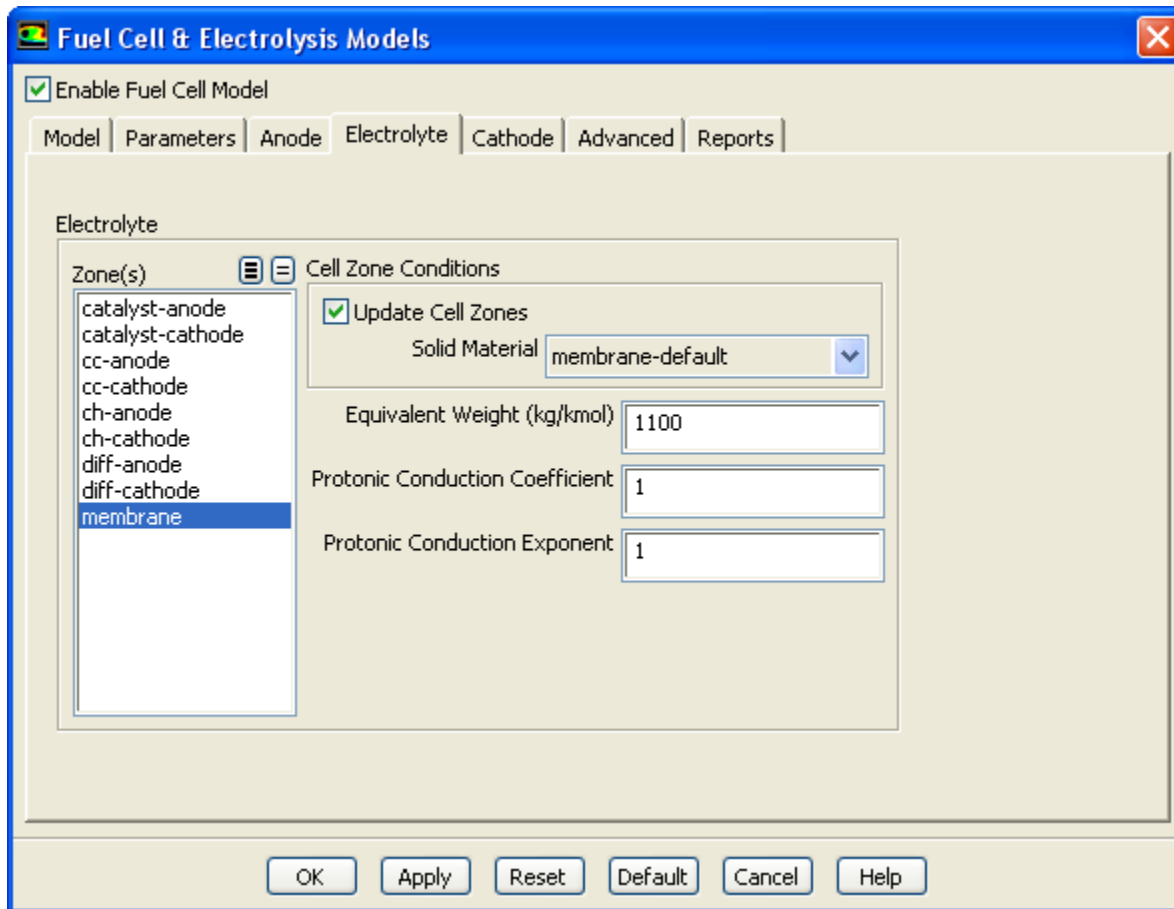
For each case of the anode's current collector, diffusion layer, and catalyst layer, you assign a solid material and/or set the porosity and the viscous resistance. These settings represent setting a cell zone

condition. With the **Update Cell Zones** option turned on (the default setting), this cell zone condition is applied to all selected zones in the **Zone(s)** list. If you want to set the cell zone conditions for each zone individually (using the **Cell Zone Conditions** task page), you should turn off the **Update Cell Zones** option.

2.6.4. Specifying Electrolyte/Membrane Properties

You can use the **Electrolyte** tab of the **Fuel Cell and Electrolysis Models** dialog box to specify zones and properties of the electrolyte/membrane portion of the fuel cell.

Figure 2.8 The Electrolyte Tab of the Fuel Cell and Electrolysis Models Dialog Box



1. Select a corresponding zone from the **Zone(s)** list. If you are modeling a fuel cell stack, then you must pick *all* membrane zones as a group.
2. Select a **Solid Material** from the corresponding drop-down list. Solid materials can be customized using the **Create/Edit Materials** dialog box.
3. Specify a value for the **Equivalent Weight** (M_m in *Equation 1–35* (p. 11)).
4. Specify a value for the **Protonic Conduction Coefficient** (β in *Equation 1–33* (p. 11)). This is used to calculate the membrane phase electric conductivity.
5. Specify a value for the **Protonic Conduction Exponent** (ω in *Equation 1–33* (p. 11)).

Note that the Fuel Cell and Electrolysis Model allows you to model the electrolyte/membrane as either a fluid zone or as a solid zone. For PEMFC, the Fuel Cell and Electrolysis Model still allows for water and the ionic current to pass through the electrolyte/membrane.

2.6.4.1. Specifying Cell Zone Conditions for the Membrane

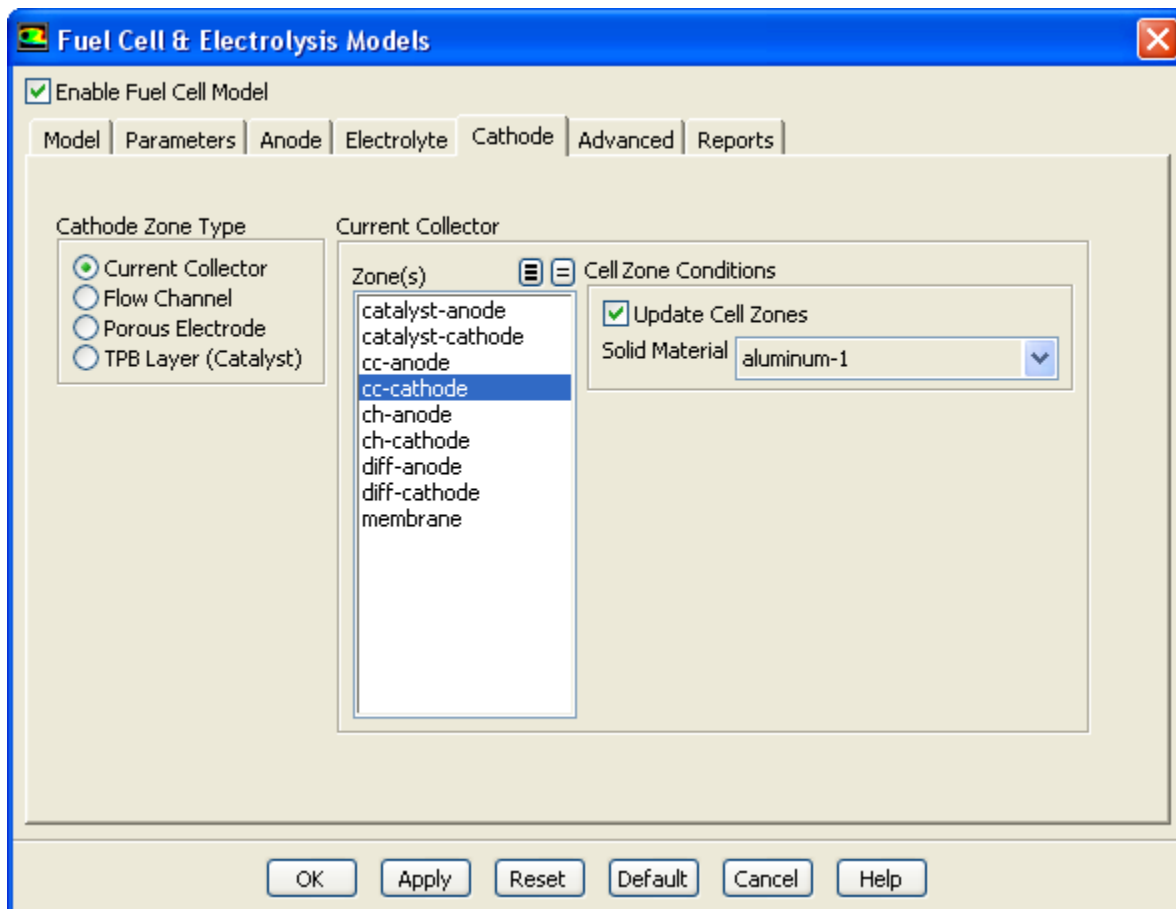
When you assign a solid material to the membrane, you are setting a cell zone condition. With the **Update Cell Zones** option turned on (the default setting), this cell zone condition is applied to all selected zones in the **Zone(s)** list. If you want to set the cell zone conditions for each zone individually (using the **Cell Zone Conditions** task page), you should turn off the **Update Cell Zones** option.

2.6.5. Specifying Cathode Properties

You can use the **Cathode** tab of the **Fuel Cell and Electrolysis Models** dialog box to specify zones and properties of the current collector, the flow channel, the diffusion layer, and the catalyst layer for the cathode portion of the fuel cell.

2.6.5.1. Specifying Current Collector Properties for the Cathode

Figure 2.9 The Cathode Tab of the Fuel Cell and Electrolysis Models Dialog Box With Current Collector Selected

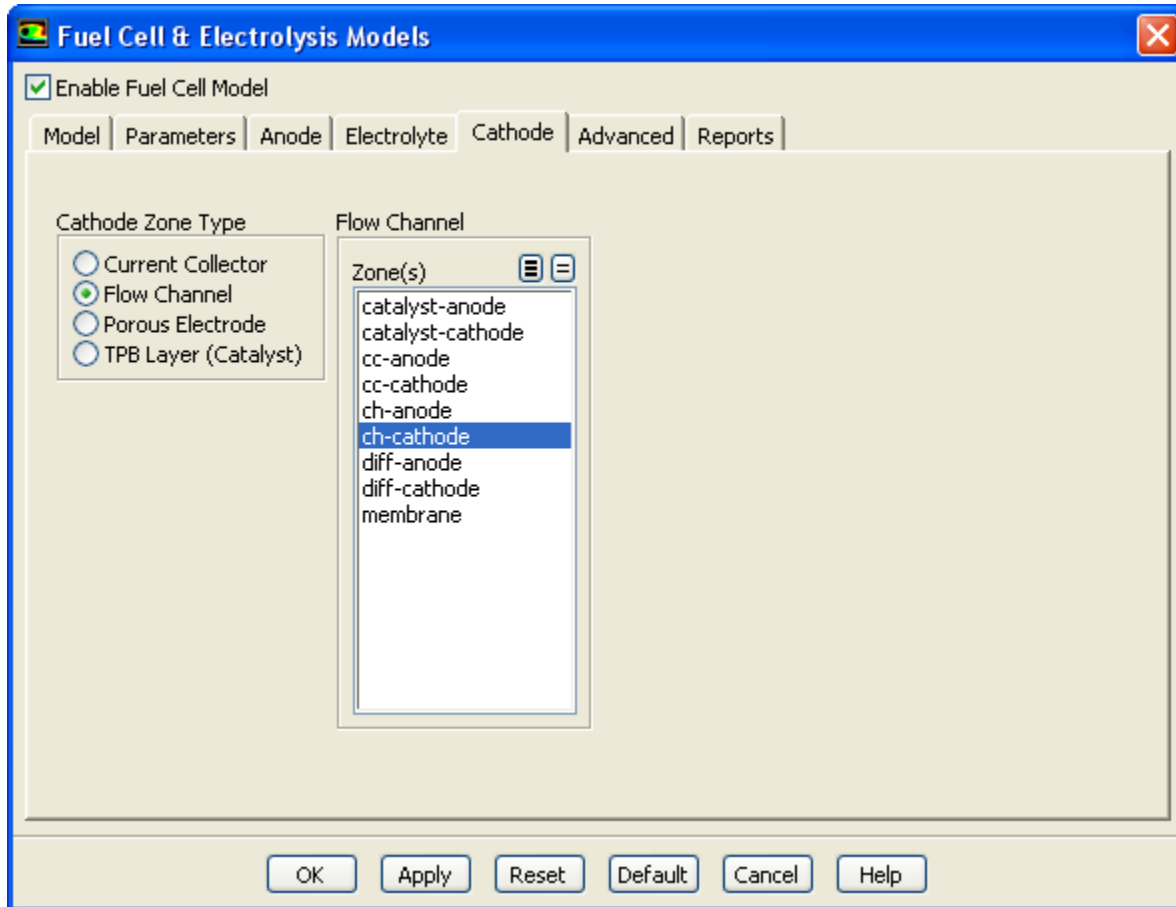


1. Select the **Cathode** tab of the **Fuel Cell and Electrolysis Models** dialog box.
2. Select **Current Collector** under **Cathode Zone Type**.
3. Select a corresponding zone from the **Zone(s)** list. If you are modeling a fuel cell stack, then you must pick *all* zones of a particular type as a group.
4. Select a **Solid Material** from the corresponding drop-down list. Solid materials can be customized using the **Create/Edit Materials** dialog box. Note that for the **Electrical Conductivity**, you can only

choose a constant value in the **Create/Edit Materials** dialog box. The solid electrical conductivity value is the diffusivity of the solid phase potential in the solid zones.

2.6.5.2. Specifying Flow Channel Properties for the Cathode

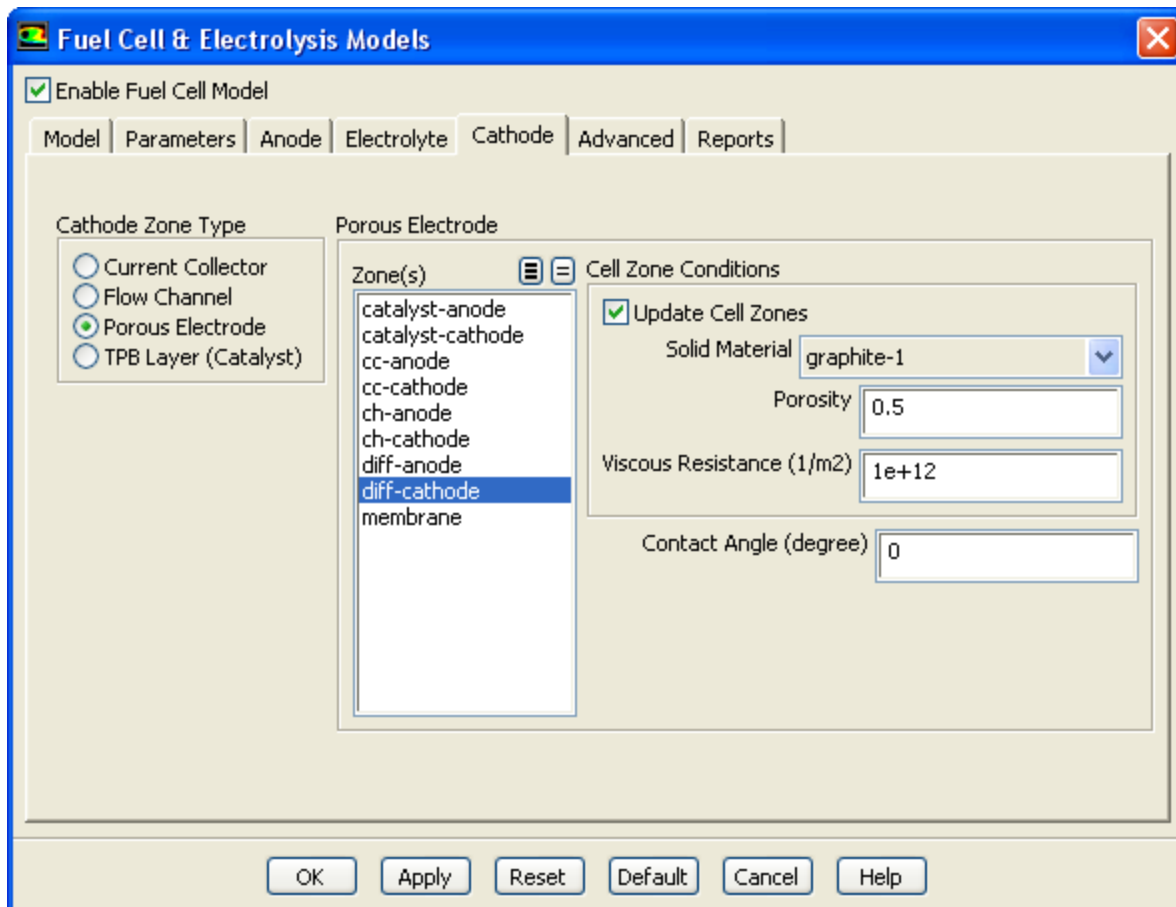
Figure 2.10 The Cathode Tab of the Fuel Cell and Electrolysis Models Dialog Box With Flow Channel Selected



1. Select the **Cathode** tab of the **Fuel Cell and Electrolysis Models** dialog box.
2. Select **Flow Channel** under **Cathode Zone Type**.
3. Select a corresponding zone from the **Zone(s)** list. If you are modeling a fuel cell stack, then you must pick *all* zones of a particular type as a group.

2.6.5.3. Specifying Porous Electrode Properties for the Cathode

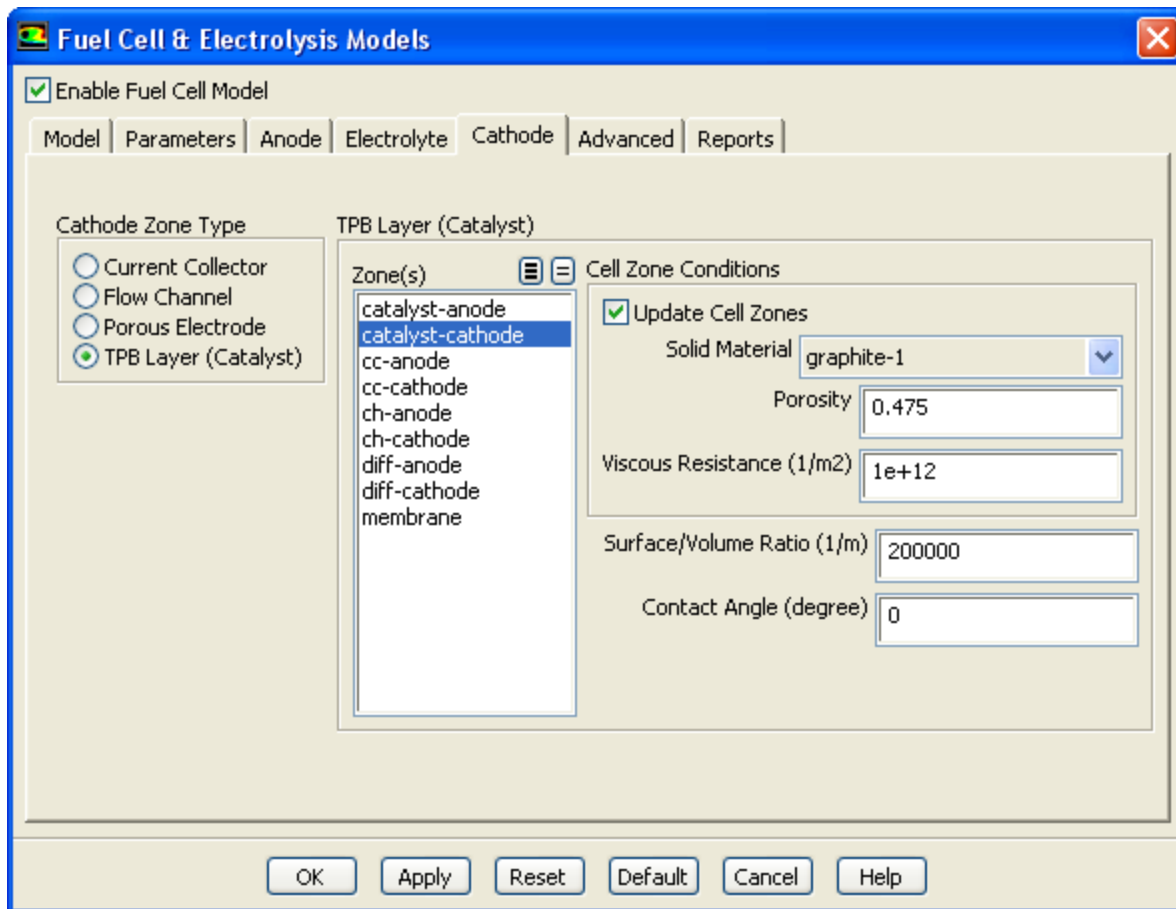
Figure 2.11 The Cathode Tab of the Fuel Cell and Electrolysis Models Dialog Box With Porous Electrode Selected



1. Select the **Cathode** tab of the **Fuel Cell and Electrolysis Models** dialog box.
2. Select **Porous Electrode** under **Cathode Zone Type**.
3. Select a corresponding zone from the **Zone(s)** list. If you are modeling a fuel cell stack, then you must pick *all* zones of a particular type as a group.
4. Select a **Solid Material** from the corresponding drop-down list. Solid materials can be customized using the **Create/Edit Materials** dialog box. Note that for the **Electrical Conductivity**, you can only choose a constant value in the **Create/Edit Materials** dialog box. The solid electrical conductivity value is the diffusivity of the solid phase potential in the solid zones.
5. Specify a value for the **Porosity**.
6. Specify a value for the **Viscous Resistance**.
7. Specify a value for the **Contact Angle** for multiphase fuel cell calculations (θ_c in *Equation 1-29* (p. 9)).

2.6.5.4. Specifying Catalyst Layer Properties for the Cathode

Figure 2.12 The Cathode Tab of the Fuel Cell and Electrolysis Models Dialog Box With TPB Layer - Catalyst Selected



1. Select the **Cathode** tab of the **Fuel Cell and Electrolysis Models** dialog box.
2. Select **TPB Layer(Catalyst)** under **Cathode Zone Type**.
3. Select a corresponding zone from the **Zone(s)** list. If you are modeling a fuel cell stack, then you must pick *all* zones of a particular type as a group.
4. Select a **Solid Material** from the corresponding drop-down list. Solid materials can be customized using the **Create/Edit Materials** dialog box. Note that for the **Electrical Conductivity**, you can only choose a constant value in the **Create/Edit Materials** dialog box. The solid electrical conductivity value is the diffusivity of the solid phase potential in the solid zones.
5. Specify a value for the **Porosity**.
6. Specify a value for the **Viscous Resistance**.
7. Specify a value for the **Surface/Volume Ratio** (the specific active surface area in *Equation 1–10* (p. 6)).
8. Specify a value for the **Contact Angle** for multiphase fuel cell calculations (θ_c in *Equation 1–29* (p. 9)).

2.6.5.5. Specifying Cell Zone Conditions for the Cathode

For each case of the cathode's current collector, diffusion layer, and catalyst layer, you assign a solid material and/or set the porosity and the viscous resistance. These settings represent setting a cell zone

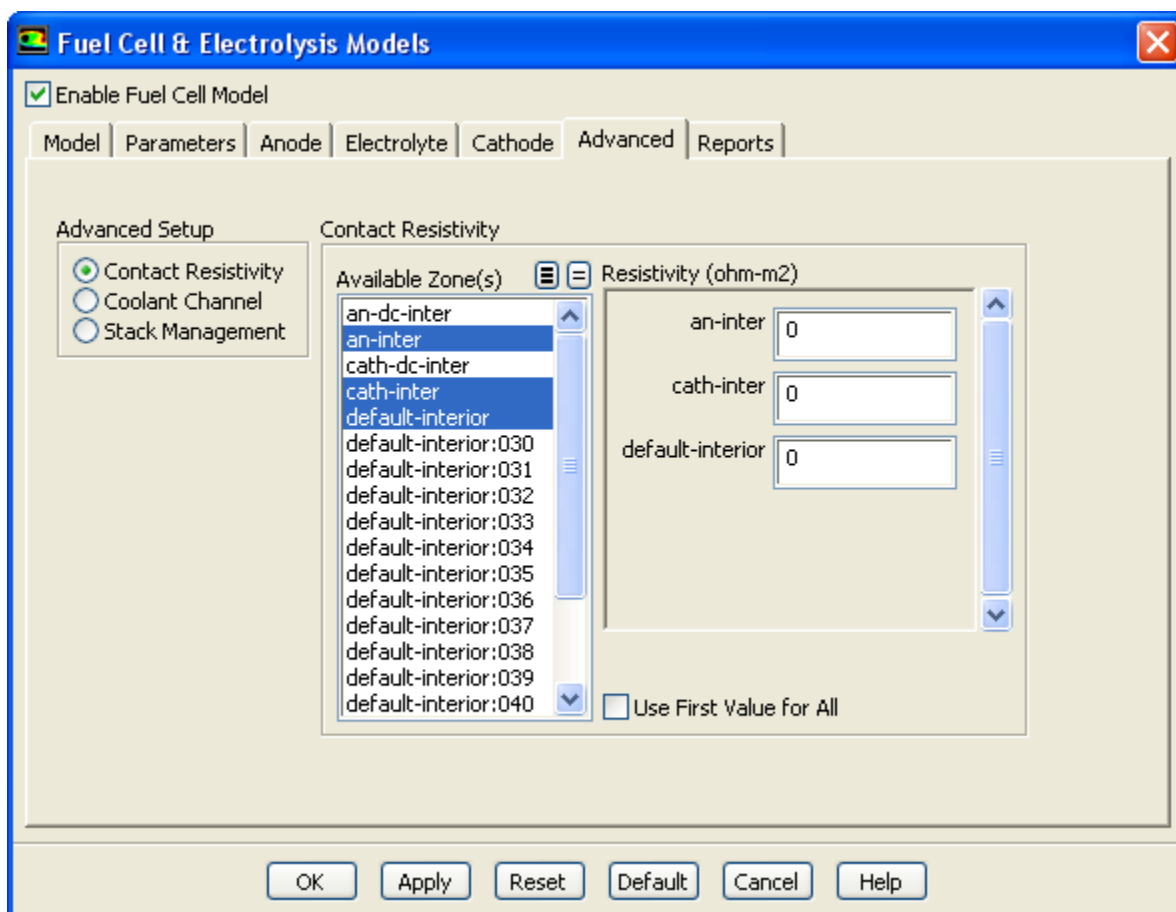
condition. With the **Update Cell Zones** option turned on (the default setting), this cell zone condition is applied to all selected zones in the **Zone(s)** list. If you want to set the cell zone conditions for each zone individually (using the **Cell Zone Conditions** task page), you should turn off the **Update Cell Zones** option.

2.6.6. Setting Advanced Properties

You can use the **Advanced** tab of the **Fuel Cell and Electrolysis Models** dialog box to specify the contact resistivity for any material interface in the geometry, set parameters for coolant channels, and define fuel stack units for managing stacks of fuel cells.

2.6.6.1. Setting Contact Resistivities for the Fuel Cell and Electrolysis Model

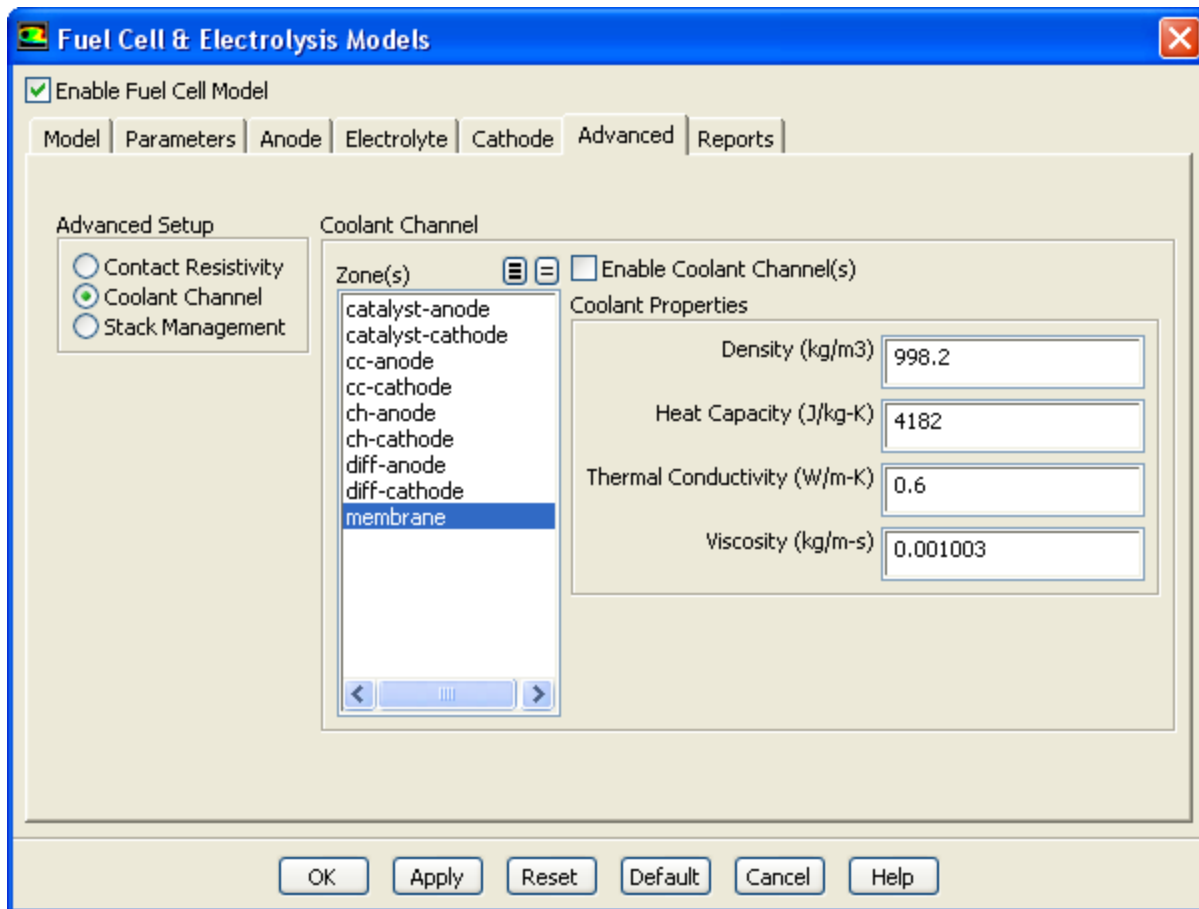
Figure 2.13 The Advanced Tab of the Fuel Cell and Electrolysis Models Dialog Box for Contact Resistivities



1. Select the **Advanced** tab of the **Fuel Cell and Electrolysis Models** dialog box.
2. Select **Contact Resistivity** under **Advanced Setup**.
3. Select any number of corresponding interfaces from the **Available Zone(s)** list. These zones are face zones over which a jump in electrical potential is caused by imperfect conduction.
4. Specify a value for the **Resistivity** for each specified zone.
5. To simplify the input, you can choose to use the resistivity value of the first selected zone for all others as well by turning on the **Use First Value for All** option.

2.6.6.2. Setting Coolant Channel Properties for the Fuel Cell and Electrolysis Model

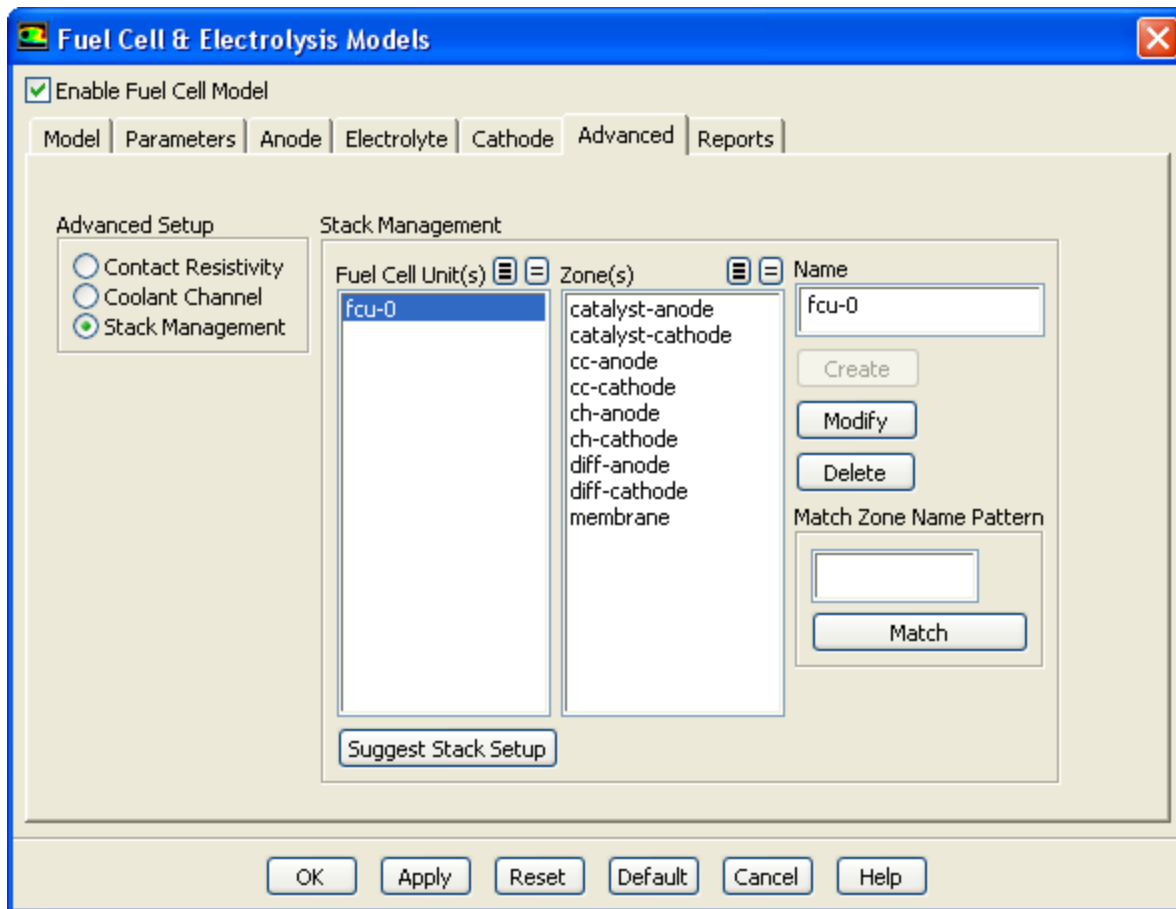
Figure 2.14 The Advanced Tab of the Fuel Cell and Electrolysis Models Dialog Box for the Coolant Channel



1. Select the **Advanced** tab of the **Fuel Cell and Electrolysis Models** dialog box.
2. Select **Coolant Channel** under **Advanced Setup**.
3. Select any number of corresponding zones from the **Zone(s)** list.
4. Specify a value for the **Density**.
5. Specify a value for the **Heat Capacity**.
6. Specify a value for the **Thermal Conductivity**.
7. Specify a value for the **Viscosity**.
8. To enable the coolant channel, turn on the **Enable Coolant Channel(s)** option. Amongst other settings, this will change the mixture to include the coolant species, which is otherwise absent.

2.6.6.3. Managing Stacks for the Fuel Cell and Electrolysis Model

Figure 2.15 The Advanced Tab of the Fuel Cell and Electrolysis Models Dialog Box for Stack Management



The ANSYS FLUENT Fuel Cell and Electrolysis Model allows you to model fuel cell stacks as well as individual fuel cells. In the **Advanced** tab of the **Fuel Cell and Electrolysis Models** dialog box, you can define *fuel cell units* for each fuel cell in a stack. A fuel cell unit consists of all zones of a single fuel cell in the stack.

Important

If you are only modeling a single fuel cell, then you do not need to set anything for **Stack Management** in the **Advanced** tab of the **Fuel Cell and Electrolysis Models** dialog box.

1. Select the **Advanced** tab of the **Fuel Cell and Electrolysis Models** dialog box.
2. Select **Stack Management** under **Advanced Setup**.
3. Since a fuel cell unit consists of all zones of a single fuel cell in the stack, select the corresponding zones from the **Zone(s)** list.
4. Create a new fuel cell unit by clicking the **Create** button. The new fuel cell is listed under **Fuel Cell Unit(s)** with a default name.

5. Edit a pre-existing fuel cell unit by selecting it in the **Fuel Cell Unit(s)** list. The zones in this fuel cell unit are automatically selected in the **Zone(s)** list. You can then modify the zones that comprise the fuel cell unit and/or change its name in the **Name** field and click **Modify** to save the new settings.
6. Remove a pre-existing fuel cell unit by selecting it in the **Fuel Cell Unit(s)** list and clicking the **Delete** button.
7. If your model contains many zone names, you can use the **Match Zone Name Pattern** field to specify a pattern to look for in the names of zones. Type the pattern in the text field and click **Match** to select (or deselect) the zones in the **Zones** list with names that match the specified pattern. You can match additional characters using * and ?. For example, if you specify `wall*`, all surfaces whose names begin with **wall** (e.g., **wall-1**, **wall-top**) will be selected automatically. If they are all selected already, they will be deselected. If you specify `wall?`, all surfaces whose names consist of **wall** followed by a single character will be selected (or deselected, if they are all selected already).

For example, in a stack there are many fuel cells, say 10 - 100, each having at least 9 zones (current collector, gas channel, diffusion layer, and catalyst layer for both anode and cathode and a membrane). Additionally, there may be coolant channels, and it may be that for mesh construction reasons each of these physical zones is made up of more than one mesh zone. Even for small stacks, you can easily end up having hundreds of cell zones in an ANSYS FLUENT mesh. Therefore, you may want to consider numbering the fuel cells in a stack and to use the assigned fuel cell number in the names of the mesh zones. When you set up your stacked fuel cell case, you would use the **Match Zone Name Pattern** field to pick all the zones belonging to a single fuel cell in the stack, rather than scrolling through the potentially very long list and selecting them manually.

8. You can have ANSYS FLUENT attempt to automatically determine the zones that constitute a single fuel cell in a stack using the **Suggest Stack Setup** button. Manually performing this task is often time-consuming and error-prone. Using the **Suggest Stack Setup** button can save you from having to manually enter this information yourself for potentially hundreds of zones.

When using the **Suggest Stack Setup** button, ANSYS FLUENT needs to correctly identify electrically conducting parts and their connectivity (anode, electrolyte, cathode, coolant channels, and external contacts) using zone information generally required by the fuel cell model anyway. ANSYS FLUENT requires zone information to have been specified in *all* of the following tabs in the **Fuel Cell and Electrolysis Models** dialog box:

- In the **Anode** tab, specify zone information for the current collector, the porous electrode, and the TPB catalyst layer.
- In the **Electrolyte** tab, specify zone information for the electrolyte/membrane.
- In the **Cathode** tab, specify zone information for the current collector, the porous electrode, and the TPB catalyst layer.
- In the **Advanced** tab, specify zone information for the coolant channel.
- In the **Reports** tab, specify the external contact interface(s).
- In the **Advanced** tab, click the **Suggest Stack Setup** button.

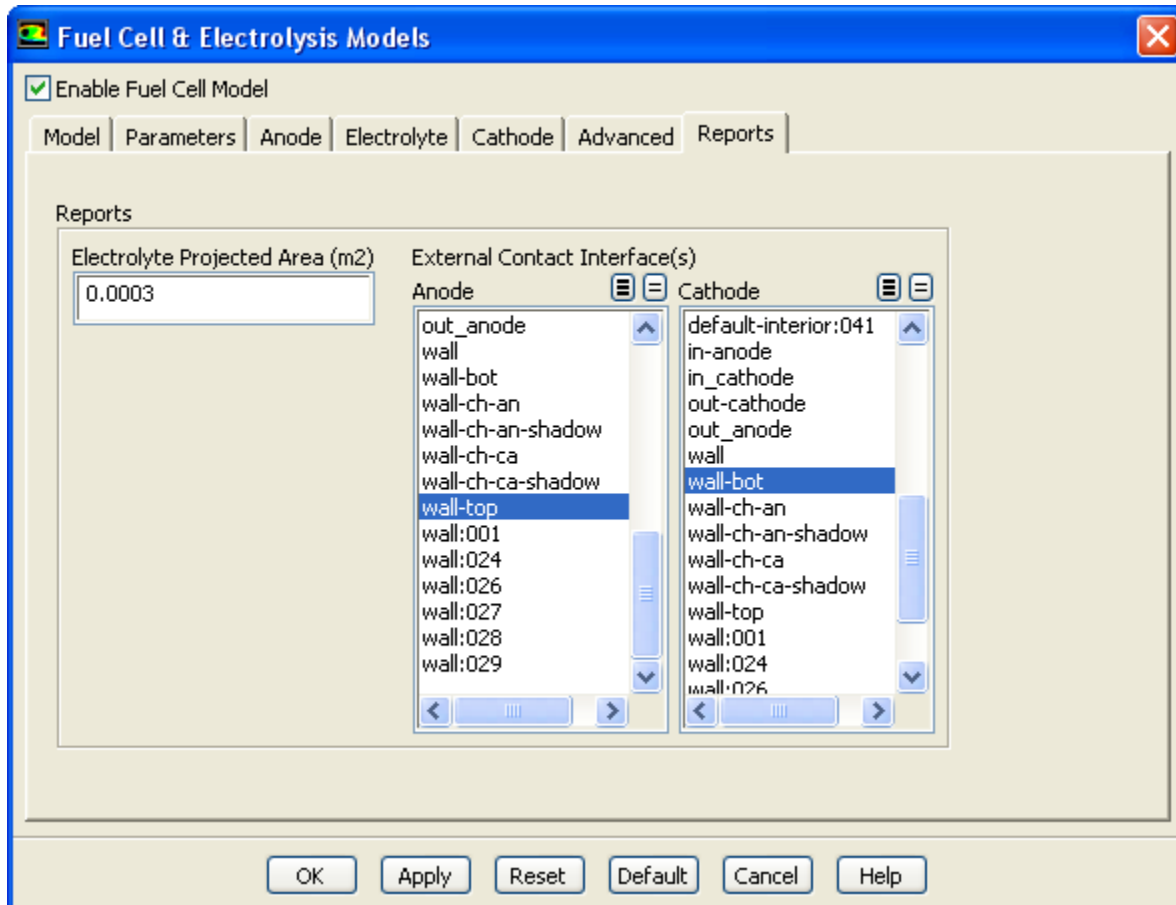
Important

If your fuel cell model inputs are incorrect (or incomplete), ANSYS FLUENT cannot completely be sure that the resulting setup is correct. Also, even if your inputs are correct, an unconventional fuel cell design may cause ANSYS FLUENT to suggest an inaccurate stack setup. So, it is your responsibility to verify the stack setup prior to clicking either the **OK** or the **Apply** buttons.

2.6.7. Reporting on the Solution

You can use the **Reports** tab of the **Fuel Cell and Electrolysis Models** dialog box to set up parameters that will be useful in reporting data relevant to the fuel cell.

Figure 2.16 The Reports Tab of the Fuel Cell and Electrolysis Models Dialog Box



The **Electrolyte Projected Area** field requires the projected area of the Membrane Electrolyte Assembly (MEA) and is only used to calculate the average current density. The assembly consists of the membrane and the catalyst layers above and below the membrane. The value of the projected area can be computed from the **Projected Surface Areas** dialog box.

Reports → **Projected Areas** → **Set Up...**

The **External Contact Interface(s)** fields requires the face zones that act as external contact surfaces for the anode and the cathode.

These inputs are used to report cell voltage. For potentiostatic boundary conditions, this is the difference between the provided values, but for galvanostatic boundary conditions, the cell voltage is part of the solution.

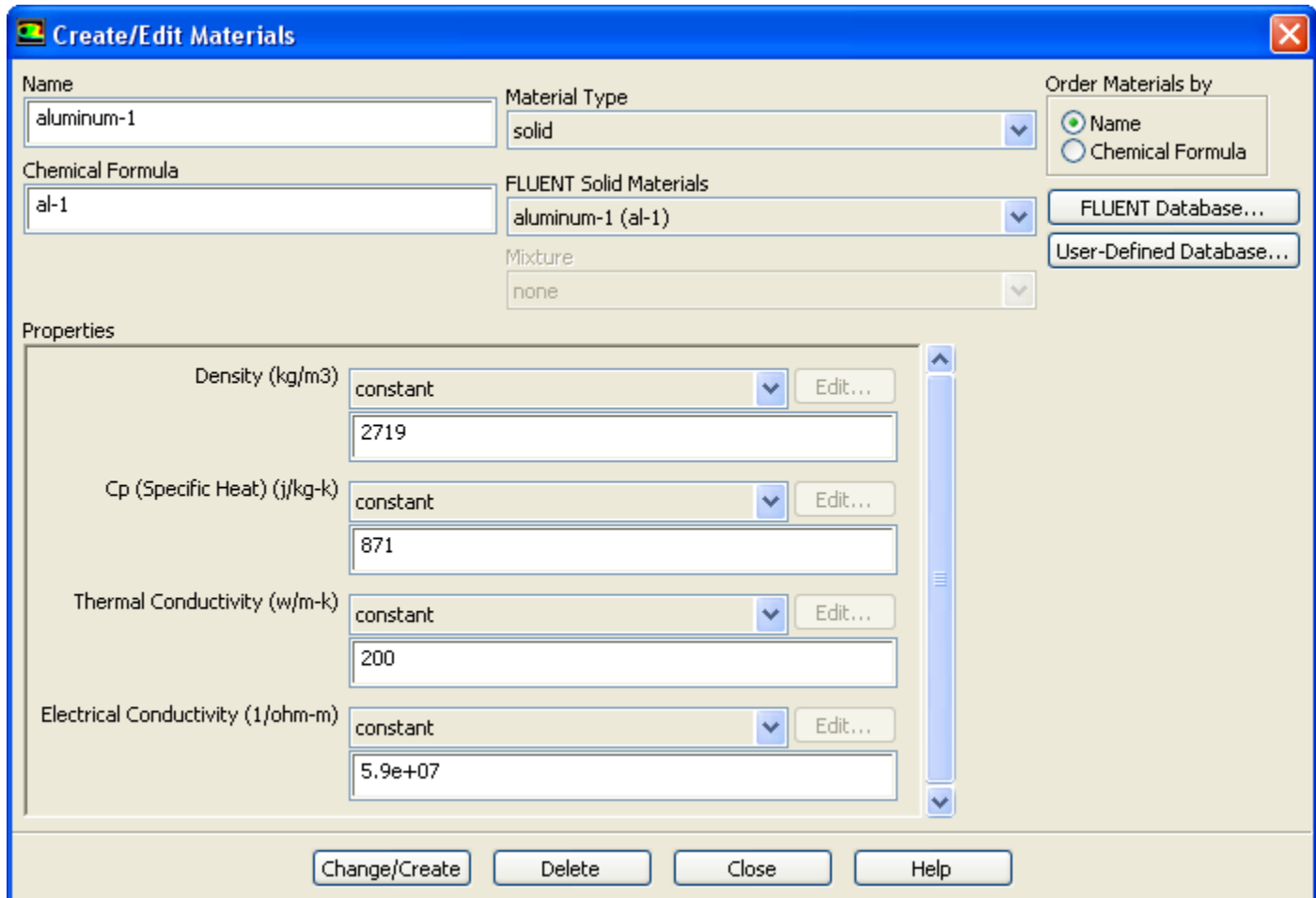
2.7. Modeling Current Collectors

In previous versions of ANSYS FLUENT, user-defined scalar (UDS) equations could only be solved in fluid zones. This restriction is now removed. As a result, the Fuel Cell and Electrolysis module allows

you to model current collectors as solid, as well as fluid zones. One advantage of using solids as the current collector is that the convergence of the species equations are not hindered by the potentially skewed mesh inside the current collectors.

If fluid zones are used to model solid current collectors, ANSYS FLUENT automatically sets velocities to zero and cuts off species transport into these zones. If solid zones are used, however, you need to activate the solution of the electric potential (UDS-0) in these solid zones (see the separate ANSYS FLUENT User's Guide for details). The value of the **Electric Conductivity** for the solid material needs to be assigned in the **Create/Edit Materials** dialog box.

Figure 2.17 The Electric Conductivity Field in the Create/Edit Materials Dialog Box



Important

Note that the **UDS Diffusivity** should be set to user-defined (`cond::fuelcells`). Do not use the `defined-per-uds` option.

For more information on the user-defined scalar diffusivity, see the separate ANSYS FLUENT [User's Guide](#).

Note that the Fuel Cell and Electrolysis Model allows you to model current collectors either as porous media zones (if you want to allow for mass and momentum transport within the collectors) or as solid zones.

2.8. Fuel Cell and Electrolysis Model Boundary Conditions

The following boundary conditions need to be defined for the Fuel Cell and Electrolysis simulation based on your problem specification:

- Anode Inlet
 - Mass flow rate
 - Temperature
 - Direction specification method
 - Mass fractions (e.g., **h2**, and **h2o**).
 - The coolant *must* be set to zero if coolant channels are enabled.
 - UDS-2 (Water Saturation) must be set to 0
- Cathode Inlet
 - Mass flow rate
 - Temperature
 - Direction specification method
 - Mass fractions (e.g. **o2**, **h2o**, and **n2**).
 - The coolant *must* be set to zero if coolant channels are enabled.
 - UDS-2 (Water Saturation) must be set to 0
- Coolant Inlet (if any)
 - Mass flow rate
 - Temperature
 - Direction specification method
 - Coolant mass fraction set to 1
 - UDS-2 (Water Saturation) must be set to 0
- Pressure Outlets (all)

Realistic backflow conditions.
- Terminal Anode
 - Temperature (or flux if known)
 - UDS-0 (electric potential) set to ground voltage
- Terminal Cathode
 - Temperature (or flux if known)
 - UDS-0 (electric potential) is set to the voltage of the cathode (if solving to constant voltage), or the UDS-0 (electric potential) flux is set to the current density in A/m^2 (SI units) (if solving for constant current). Note that the sign of the UDS-0 flux on the cathode side is negative.

2.9. Solution Guidelines for the Fuel Cell and Electrolysis Model

For potentiostatic boundary conditions, after initialization, steady state solutions are calculated easily for cell voltages close to the open-circuit voltage. The same can be said for galvanostatic boundary

conditions and low electric current. By lowering the cell voltage or by raising the average electric current, you can calculate subsequent stationary solutions.

In the event of convergence problems, it is recommended to change the multigrid cycle to **F-cycle** with **BCGSTAB** (bi-conjugate gradient stabilized method) selected as the stabilization method for the species and the two potential equations. For the species and the user-defined scalar equations, it may be necessary to reduce the termination (criteria) of the multigrid-cycles to 1×10^{-3} . For stack simulations, the termination criterion may be reduced to $\times 10^{-7}$ for the two potential equations.

Also, it may be useful to turn off **Joule Heating** and **Reaction Heating** in the **Fuel Cell and Electrolysis Models** dialog box (in the **Model** tab) for the first few (approximately 5-10) iterations after initialization. This allows the two electric potentials to adjust from their initial values to more physical values, avoiding the possibility of extreme electrochemical reactions and electric currents that would in turn adversely impact the solution.

2.10. Postprocessing the Fuel Cell and Electrolysis Model

You can perform post-processing using standard ANSYS FLUENT quantities and by using user-defined scalars and user-defined memory allocations. By default, the ANSYS FLUENT Fuel Cell and Electrolysis Model defines several user-defined scalars and user-defined memory allocations, described in [Table 2.1: User-Defined Scalar Allocations](#) (p. 38) and [Table 2.2: User-Defined Memory Allocations](#) (p. 38).

Table 2.1 User-Defined Scalar Allocations

UDS 0	Electric Potential (solid phase potential) (Volts)
UDS 1	Protonic Potential (membrane phase potential) (Volts)
UDS 2	Water Saturation (liquid saturation)
UDS 3	Water Content

Table 2.2 User-Defined Memory Allocations

UDM 0	X Current Flux Density (A / m^2)
UDM 1	Y Current Flux Density (A / m^2)
UDM 2	Z Current Flux Density (A / m^2)
UDM 3	Current Flux Density Magnitude (A / m^2)
UDM 4	Ohmic Heat Source (W / m^3)
UDM 5	Reaction Heat Source (W / m^3)
UDM 6	Overpotential (Volts)
UDM 7	Phase Change Source ($kg / m^3 - s$)
UDM 8	Osmotic Drag Coefficient
UDM 9	Liquid Water Activity
UDM 10	Membrane Water Content
UDM 11	Protonic Conductivity (1/ohm-m)

UDM 12	Back Diffusion Mass Source ($kg/m^3 - s$)
UDM 13	Transfer Current (A/m^3)
UDM 14	Osmotic Drag Source ($kg/m^3 - s$)

You can obtain this list by opening the **Execute On Demand** dialog box and pulling down the **Function** drop-down list.

Define → **User-Defined** → **Execute On Demand...**

and access the execute-on-demand function called `list_pemfc_udf`.

Alternatively, you can view the listing that appears when you first load your Fuel Cell and Electrolysis case, or you can type `list_pemfc_udf` in the text user interface and the listing will appear in the console window.

Important

When you load older Fuel Cell and Electrolysis cases into ANSYS FLUENT, and you are monitoring a UDS using volume or surface monitors, make sure you re-visit the corresponding monitors dialog box (e.g., the **Volume Monitor** or the **Surface Monitor** dialog box) to make sure that the correct UDS name is used for the appropriate monitor.

2.11. User-Accessible Functions

As noted in *Properties* (p. 10), you can directly incorporate your own formulations and data for the properties of the fuel cell membrane using the `pem_user.c` source code file.

The following listing represents a description of the contents of the `pem_user.c` source code file:

- `real Get_P_sat(real T)`: Returns the value of the water vapor saturation pressure as a function of temperature (*Equation 1–40* (p. 12)).
- `real Water_Activity(real P, real T, cell_t c, Thread *t)`: Returns the value of water activity (*Equation 1–38* (p. 12)).
- `real Water_Content(real act)`: Returns the value of the membrane water content at the membrane catalyst interface (*Equation 1–37* (p. 12)).
- `real Osmotic_Drag_Coefficient(real P, real T, cell_t c, Thread *t)`: Returns the value of the osmotic drag coefficient (*Equation 1–34* (p. 11)).
- `real Membrane_Conductivity(real lam, cell_t c, Thread *t)`: Returns the value of the membrane's protonic conductivity (*Equation 1–33* (p. 11)).
- `real Electrolyte_Conductivity(cell_t c, Thread *t)`: Returns the value of the ionic conductivity in the electrolyte (*Equation 1–32* (p. 11)). (SOFC and Electrolysis only)
- `real Water_Content_Diffusivity(real lam, real T, real mem_mol_density, cell_tc, Thread*t)`: Returns the value of the water content diffusivity in the membrane (*Equation 1–36* (p. 11)).
- `real Gas_Diffusivity(cell_tc, Thread *t, int j_spe)`: Returns the value of the gaseous species diffusivities in the channels, gas diffusion layers and catalysts (*Equation 1–30* (p. 10)).

- `real MCD_Gas_Diffusivity(cell_t c, Thread *t, int i)`: Returns the tortuosity-corrected value of the gas species diffusion coefficients computed with the multicomponent diffusion option (*Equation 1-31* (p. 10)).
- `real Saturation_Diffusivity(real sat, real cos_theta, real porosity, cell_t c, Thread *t)`: Returns the value of diffusivity of the liquid saturation. It comprises the term $\rho_l \frac{Ks3}{\mu_l} \frac{dp_c}{ds}$ from *Equation 1-26* (p. 9).
- `real Anode_AV_Ratio(cell_t c, Thread *t)`: Returns the value of the specific active surface area (ζ_{an} in *Equation 1-9* (p. 6)) for the anode catalyst.
- `real Cathode_AV_Ratio(cell_t c, Thread *t)`: Returns the value of the specific active surface area (ζ_{cat} in *Equation 1-10* (p. 6)) for the cathode catalyst.
- `real Anode_J_TransCoef(cell_t c, Thread *t)`: Returns the value of the anode reaction reference current density $\frac{\zeta_{an}^{ref}}{([H_2]_{ref})^{\gamma_{an}}}$ used in *Equation 1-9* (p. 6).
- `real Cathode_J_TransCoef(cell_t c, Thread *t)`: Returns the value of the cathode reaction reference current density $\frac{\zeta_{cat}^{ref}}{([O_2]_{ref})^{\gamma_{cat}}}$ used in *Equation 1-10* (p. 6).
- `real Open_Cell_Voltage(cell_t c, Thread *t)`: Returns the value of the open-circuit voltage V_{oc} used in *Equation 1-14* (p. 7).
- `real Leakage_Current(cell_t c, Thread *t)`: Returns the value of the specific leakage current (I_{leak} in *Equation 1-41* (p. 13) through *Equation 1-46* (p. 13)).
- `void Set_UDS_Names(char uds[n_uds_required][STRING_SIZE])`: Used to rename user defined scalars (UDSs). Note that the units of the user defined scalars cannot be changed.

```
void Set_UDS_Names(char uds[n_uds_required][STRING_SIZE])
{
    strncpy(uds[0], "Electric Potential", STRING_SIZE-1);
    strncpy(uds[1], "Protonic Potential", STRING_SIZE-1);
    strncpy(uds[2], "Water Saturation", STRING_SIZE-1);
    strncpy(uds[3], "Water Content", STRING_SIZE-1);
}
```

If you want to change the names of UDSs, change the second argument of the `strncpy` functions, recompile and link the module as with any modification to `pem_user.c`. Note that `STRING_SIZE` is fixed in `pem.h` and should not be changed.

Important

When you load older Fuel Cell and Electrolysis cases into ANSYS FLUENT, and you are monitoring a UDS using volume or surface monitors, make sure you re-visit the corresponding monitors dialog box (e.g., the **Volume Monitor** or the **Surface Monitor** dialog box) to make sure that the correct UDS name is used for the appropriate monitor.

- `void Set_UDM_Names(char udm[n_udm_required][STRING_SIZE])`: Used to rename user defined memory (UDMs). Note that the units of user defined memory cannot be changed.

```

void Set_UDM_Names(char udm[n_udm_required][STRING_SIZE])
{
    strncpy(udm[ 0], "X Current Flux Density",  STRING_SIZE-1);
    strncpy(udm[ 1], "Y Current Flux Density",  STRING_SIZE-1);
    strncpy(udm[ 2], "Z Current Flux Density",  STRING_SIZE-1);
    strncpy(udm[ 3], "Current Flux Density Magnitude", STRING_SIZE-1);
    strncpy(udm[ 4], "Ohmic Heat Source", STRING_SIZE-1);
    strncpy(udm[ 5], "Reaction Heat Source",  STRING_SIZE-1);
    strncpy(udm[ 6], "Overpotential", STRING_SIZE-1);
    strncpy(udm[ 7], "Phase Change Source (PEM)", STRING_SIZE-1);
    strncpy(udm[ 8], "Osmotic Drag Coefficient (PEM)", STRING_SIZE-1);
    strncpy(udm[ 9], "Liquid Water Activity (PEM)", STRING_SIZE-1);
    strncpy(udm[10], "Membrane Water Content (PEM)", STRING_SIZE-1);
    strncpy(udm[11], "Protonic Conductivity",  STRING_SIZE-1);
    strncpy(udm[12], "Back Diffusion Source (PEM)", STRING_SIZE-1);
    strncpy(udm[13], "Transfer Current",  STRING_SIZE-1);
    strncpy(udm[14], "Osmotic Drag Source (PEM)", STRING_SIZE-1);
}

```

If you want to change the names of UDMs, change the second argument of the `strncpy` functions, recompile and link the module as with any modification to `pem_user.c`. Note that `STRING_SIZE` is fixed in `pem.h` and should not be changed.

Important

When you load older Fuel Cell and Electrolysis cases into ANSYS FLUENT, and you are monitoring a UDM using volume or surface monitors, make sure you re-visit the corresponding monitors dialog box (e.g., the **Volume Monitor** or the **Surface Monitor** dialog box) to make sure that the correct UDM name is used for the appropriate monitor.

- `real electric_contact_resistance(face_t f, Thread *t, int ns)`: Returns the value for the electrical contact resistance.
- `real Transfer_Current(real i_ref, real gamma, int species_i, real alpha_a, real alpha_c, real *dRade, real *dRcde, Thread *t, cell_t c)`: Computes the transfer current (A/m^3), corresponding to R_{an} in [Equation 1–9 \(p. 6\)](#) and R_{cat} in [Equation 1–10 \(p. 6\)](#).

Inputs for this function include:

<code>i_ref</code>	effective transfer current coefficient, computed by <code>Cathode_J_TransCoef(c, t)</code> or <code>Anode_J_TransCoef(c, t)</code>
<code>gamma</code>	cathode or anode concentration exponent
<code>species_i</code>	species index used in fuel cells (e.g. <code>i_o2</code> , <code>i_h2</code> , <code>i_h2o</code>)
<code>alpha_a</code>	product of anode exchange coefficient and $\frac{F}{RT}$
<code>alpha_c</code>	product of cathode exchange coefficient and $\frac{F}{RT}$
<code>t</code>	current thread
<code>c</code>	current cell

Outputs for this function include:

<code>source</code>	anode or cathode volumetric transfer current (R_{an} in Equation 1–9 or R_{cat} in Equation 1–10)
<code>*dRade</code>	partial derivative of R_{an} with respect to activation loss

*dR_{cat} partial derivative of R_{cat} with respect to activation loss

For more information, please see the following sections:

[2.11.1. Compiling the Customized Fuel Cell and Electrolysis Source Code](#)

2.11.1. Compiling the Customized Fuel Cell and Electrolysis Source Code

This section includes instructions on how to compile a customized Fuel Cell and Electrolysis user-defined module. Note that you can also refer to the file `INSTRUCTIONS-CLIENT` that comes with your distribution (see `addons/fuelcells`).

Important

It is assumed that you have a basic familiarity with compiling user-defined functions (UDFs). For an introduction on how to compile UDFs, please refer to the separate UDF manual.

You will first want to use a local copy of the `fuelcells` directory in the `addons` directory before you recompile the Fuel Cell and Electrolysis module.

2.11.1.1. Compiling the Customized Source Code Under Linux

1. Make a local copy of the `fuelcells` directory. Do not create a symbolic link.

Important

The custom version of the library must be named according to the convention used by ANSYS FLUENT: e.g., `fuelcells`.

2. Change directories to the `fuelcells/src` directory.
3. Make changes to the `pem_user.c` file.
4. Edit the `makefile` located in the `src/` directory and make sure that the `FLUENT_INC` variable correctly refers to the current ANSYS FLUENT installation directory. Be careful not to leave any trailing spaces when you make your changes.
5. Define the `FLUENT_ADDONS` environment variable to correspond to your customized version of the Fuel Cell and Electrolysis module.
6. Change directories to the `fuelcells/` directory.
7. Issue the following `make` command:

```
make FLUENT_INC=[ansys_inc/v140/fluvent] FLUENT_ARCH=[arch]
-f Makefile-client
```

where `your_arch` is `lnx86` on LINUX, or `ultra` on the Sun operating system, etc.

The following example demonstrates the steps required to set up and run a customized version of the Fuel Cell and Electrolysis module that is located in a folder call `home/sample`:

- Make a directory (e.g., `mkdir -p /home/sample`).
- Copy the default `addon` library to this location.

```
cp -RH [ansys_inc/v140/fluent]/fluent14.0.0/addons/fuelcells
/home/sample/fuelcells
```

- Using a text editor, make the appropriate changes to the `pem_user.c` file located in `/home/sample/fuelcells/src/pem_user.c`
- Edit the `makefile` located in the `src/` directory and make sure that the `FLUENT_INC` variable correctly refers to the current ANSYS FLUENT installation directory. Be careful not to leave any trailing spaces when you make your changes.

- Build the library.

```
cd /home/sample/fuelcells
make FLUENT_INC=[ansys_inc/v140/fluent] FLUENT_ARCH=[arch]
-f Makefile-client
```

- Set the `FLUENT_ADDONS` environment variable (using CSH, other shells will differ).

```
setenv FLUENT_ADDONS /home/sample
```

- Start ANSYS FLUENT and load the customized module using the text interface command.

2.11.1.2. Compiling the Customized Source Code Under Windows

1. Open **Visual Studio.NET** at the DOS prompt.
2. Make sure that the `$FLUENT_INC` environment variable is correctly set to the current ANSYS FLUENT installation directory (e.g., `ANSYS Inc\v140\fluent`).
3. Make a local copy of the `fuelcells` folder. Do not create a shortcut.
4. Enter the `fuelcells\src` folder.
5. Make changes to the `pem_user.c` file.
6. Define the `FLUENT_ADDONS` environment variable to correspond to your customized version of the Fuel Cell and Electrolysis module.
7. Return to the `fuelcells` folder.
8. Issue the following command in the command window:

```
nmake /f makefile_master-client.nt
```

2.12. Using the Fuel Cell and Electrolysis Text User Interface

All of the features for the Fuel Cell and Electrolysis Model (sometimes referred to as the Resolved Electrolyte model) that are available through the graphical user interface are also available through text user interface (TUI) commands. The TUI allows text commands to be typed directly in the ANSYS FLUENT console window where additional information can be extracted and processed for more advanced analysis.

Once the fuel cell module is loaded (see [Loading the Fuel Cell and Electrolysis Module \(p. 16\)](#)), you can access the text user interface through the Console Window under `resolved-MEA-fc`. A listing of the various text commands is as follows:

```
resolved-MEA-fuelcells/
```

```
Fuel cell model menu
```

```
enable-fc-model?
```

```
Enable/disable fuel cell model.
```

select-model

Select model.

model-options

Model options.

model-parameters

Model parameters.

anode-setup/

Anode setup.

list-zones-briefly

List zone names and IDs.

current-collector

Set current collector.

flow-channel

Set flow channel.

porous-electrode

Set porous electrode.

catalyst-layer

Set catalyst layer.

cathode-setup/

Cathode setup.

list-zones-briefly

List zone names and IDs.

current-collector

Set current collector.

flow-channel

Set flow channel.

porous-electrode

Set porous electrode.

catalyst-layer

Set catalyst layer.

electrolyte-setup/

Electrolyte setup.

electrolyte-layer

Set electrolyte layer.

list-zones-briefly

List zone names and IDs.

advanced-setup/

Advanced setup.

list-zones-briefly

List zone names and IDs.

contact-resistivity

Set contact resistivity.

coolant-channel

Set coolant channel.

stack-management/

Stack setup.

list-fc-units

List fuel cell units.

list-zones-briefly

List zone names and IDs.

create-fc-unit

Create fuel cell unit.

modify-fc-unit

Modify fuel cell unit.

delete-fc-unit

Delete fuel cell unit.

set-stack-current-density

Set the current density on the anode or cathode and modify the current solution to assist convergence. Note: Input here is in units of A/cm². This is only available if the case contains valid data (e.g., after initialization, iterating, or reading in data). For more information, see [IV-Curve Calculations Using the Text Interface \(p. 45\)](#).

set-stack-voltage

Set the voltage difference in Volts between the anode and the cathode and modify the current solution to assist convergence. This is only available if the case contains valid data (e.g., after initialization, iterating, or reading in data). For more information, see [IV-Curve Calculations Using the Text Interface \(p. 45\)](#).

reset-setup

Reset the stack setup in case mistakes are made.

submit-setup

Submit the stack setup and makes the stack setup take effect.

suggest-setup

Suggest the stack setup, invoking the automatic stack setup.

controls

Set model control parameters.

reports

Set electrolyte project area and external contacts.

set-default

Set default.

2.12.1. IV-Curve Calculations Using the Text Interface

For valid case and data files, there are two text commands available to assist in the IV-curve calculation. These commands are `set-stack-voltage` (aliased as `ssv`) and `set-stack-current-density` (aliased as `ssc`), available from the Fuel Cell and Electrolysis text command menu: `/define/models/resolved-MEA-fc/advanced-setup/stack-management/`.

For fuel cells, you either prescribe the voltage and obtain the total current delivered by the fuel cell as a result, or you specify the total current (via flux boundary conditions multiplied by the area) and obtain

the voltage as part of the solution. The details of this IV-relation are specific for each single fuel cell and depend on mass and heat transport, electrochemistry and inlet conditions, outlet conditions, operating conditions, and any other parameter or material property involved in the calculation. The IV-curve is important for applications, because its product is the power delivered by the system.

As described earlier in this manual, you would start a new simulation from fairly static conditions, i.e., high voltage/low current (which implies low species transport and low heat generation). After convergence, you typically may be interested in solutions for new electric boundary conditions, i.e., either for a new cell/stack voltage or current.

In such cases, simply going to the **Boundary Conditions** task page and changing the value of the electric potential (`uds=0`) boundary condition, typically allows only small changes, most notably for stacks. Otherwise the solution will not converge. This is where the `set-stack-voltage` and `set-stack-current-density` commands are important.

In addition to changing the boundary conditions (either to a prescribed voltage or current density), these commands process the current data in order to estimate the solution for the new boundary conditions. Because these commands modify the data, you are prompted to save your data, if you have not already done so.

Before going into details of the commands, here are some general remarks about electric potential boundary conditions.

For fixed voltage boundary conditions, both external contacts have a fixed value for the electric potential (`uds=0`). The anode value will typically be zero, but it does not have to be. The cathode value will be larger than the anode value and the difference ($V_{cathode} - V_{anode}$) is the positive cell/stack voltage.

For a fixed current boundary condition, one external contact has to have a fixed value and the other flux boundary conditions. As described earlier in the manual, typically, the anode will have a fixed (zero) value, and the cathode will be floating, however, you can also set the cathode to a fixed zero potential, yielding a floating negative anode potential.

The `set-stack-voltage` command sets the effective stack voltage, i.e., the difference ($V_{cathode} - V_{anode}$). For fixed voltage boundary conditions for the previous solution, boundary conditions on both boundaries are of type fixed value and then the cathode value will be changed accordingly. In the case of fixed current boundary conditions for the previous solution, the flux boundary condition will be changed to a fixed value boundary condition, and the value adjusted accordingly with respect to the other fixed value boundary condition.

The `set-stack-current-density` command sets the current density on one boundary to the desired value. Note that the input will be in $\frac{A}{cm^2}$, not $\frac{A}{m^2}$ as you would normally have to enter in the **Boundary Conditions** task page. The reason for this is that average current densities reported in the text command interface are also in $\frac{A}{cm^2}$, and this makes it easier to choose the conditions you would like to prescribe next. Also, flux boundary conditions entered in the **Boundary Conditions** dialog box would have to have a positive sign on the anode side, and a negative sign on the cathode side. The input for the text interface command is just a positive number, signs are automatically accounted for.

For fixed current boundary conditions for the previous solution, the `set-stack-current-density` command changes the respective flux boundary condition accordingly. In the case of fixed voltage boundary conditions for the previous solution, the cathode side is chosen to be changed from a fixed value to a flux boundary condition with the new flux.

The two commands may be mixed in an IV-curve calculation. For the type of boundary condition setups currently described in this manual, boundary condition changes will consistently happen on the cathode side. However, if anode flux boundary conditions had been chosen initially, switching to fixed voltage boundary conditions by `set-stack-voltage` command and then back to fixed current boundary conditions by the `set-stack-current-density` command will then have flux boundary conditions on the cathode side. In this case, using the `set-stack-current-density` command exclusively will preserve the anode flux boundary condition setting.

Chapter 3: SOFC Fuel Cell With Unresolved Electrolyte Model Theory

This chapter presents an overview of theory and equations for solid oxide fuel cell (SOFC, with unresolved electrolyte) modeling capabilities in ANSYS FLUENT.

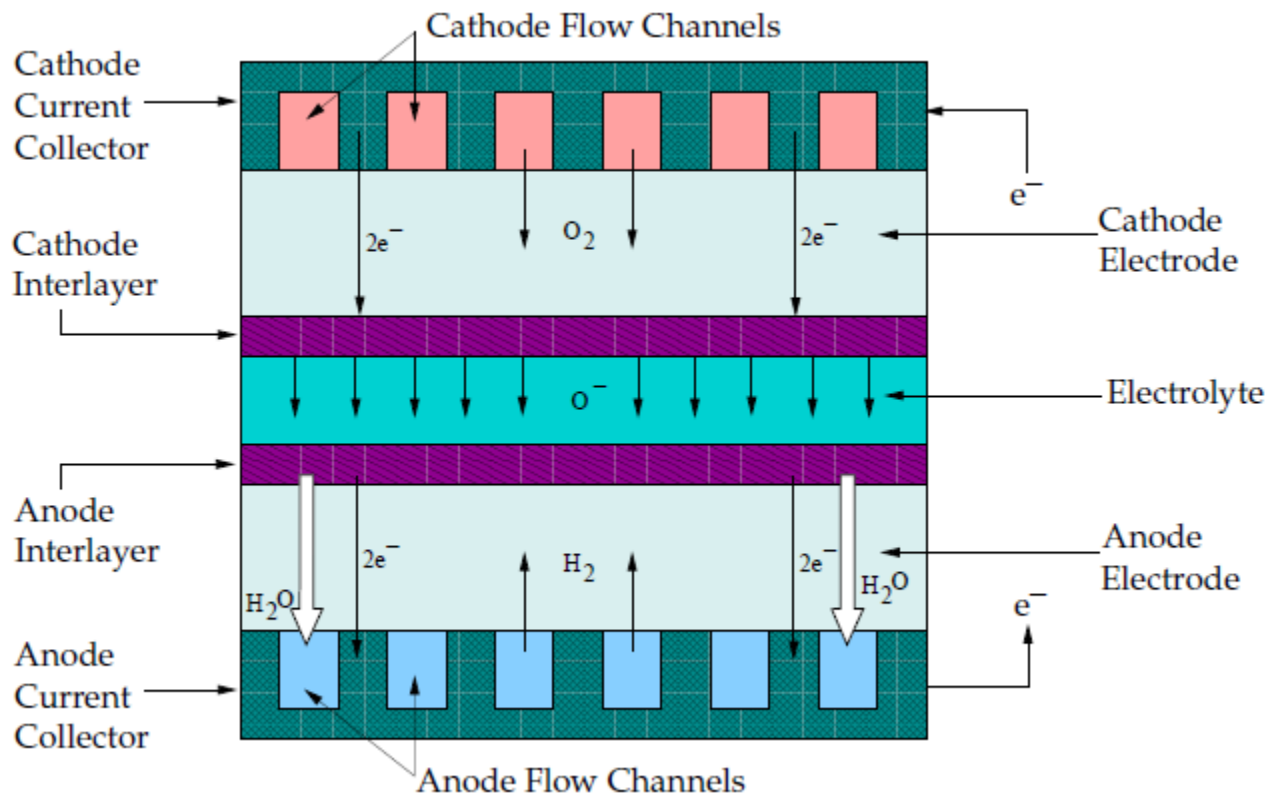
- 3.1. Introduction
- 3.2. The SOFC With Unresolved Electrolyte Modeling Strategy
- 3.3. Modeling Fluid Flow, Heat Transfer, and Mass Transfer
- 3.4. Modeling Current Transport and the Potential Field
- 3.5. Modeling Reactions

3.1. Introduction

The Solid Oxide Fuel Cell (SOFC) With Unresolved Electrolyte Model is provided as an add-on module with the standard ANSYS FLUENT licensed software. A special license is required to use the SOFC With Unresolved Electrolyte Model.

A fuel cell is an energy conversion device that converts the chemical energy of fuel into the electrical energy. A schematic of a solid oxide fuel cell (SOFC) is shown in [Figure 3.1](#) (p. 49).

Figure 3.1 Schematic of a Solid Oxide Fuel Cell



As noted in [1] (p. 79), a solid oxide fuel cell is typically composed of an anode, cathode, and an electrolyte. Multiple fuel cells can be connected together, or stacked, using electrical interconnects. The electrolyte material must be solid, i.e., non-porous, and exhibit a high ionic conductivity.

Note that the reason this modeling approach is referred to as the "SOFC Model with Unresolved Electrolyte" model is that the anode and the cathode "interlayers" and "electrolyte" (as shown in [Figure 3.1 \(p. 49\)](#)) are not actually included in the computational domain. They are modeled as a pair of wall and wall-shadow faces, named "electrolyte interfaces," with the species and energy sources and sinks due to the electrochemical reactions added to the adjacent computational cells.

All components of the fuel cell must have similar thermal expansion in order to minimize thermal stresses, which may cause cracking and de-lamination during thermal cycling. In addition, the components must be chemically stable in order to limit chemical interactions with other cell components.

A solid oxide fuel cell works by having electrically conducting porous ceramic electrodes attached on each side of an ionically conducting ceramic material. At the cathode/electrolyte/gas interface, also known as the triple phase boundary, oxygen is reduced to oxygen ions. The oxygen ions are conducted through the oxygen vacancies in the electrolyte to the anode side. At the anode/electrolyte/gas interface, oxygen ions combine to react with hydrogen at the anode electrode to form water and release electrons. The electrons travel through an external circuit to a load and back to the cathode electrode to close the circuit.

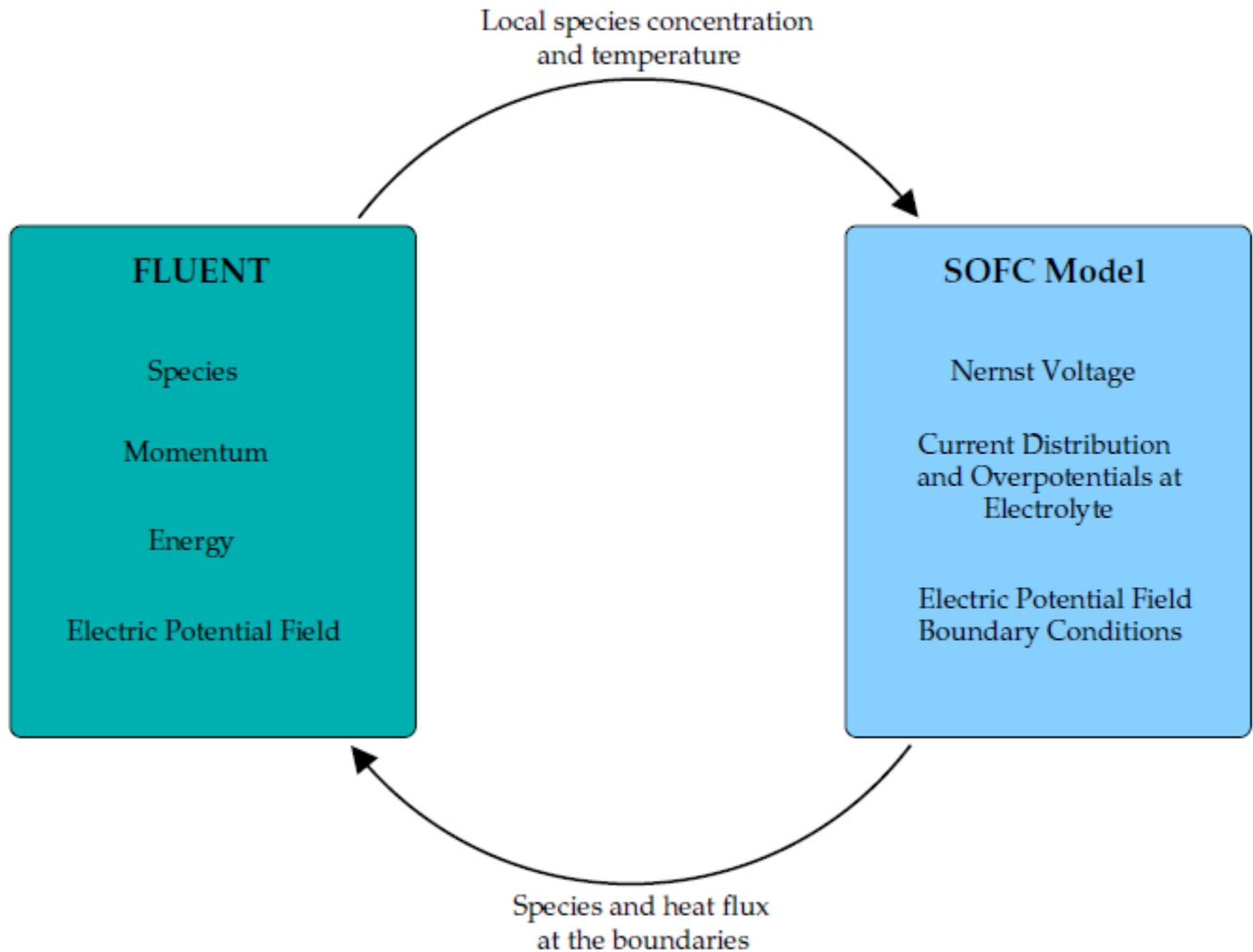
The ANSYS FLUENT SOFC With Unresolved Electrolyte Model provides the following features:

- Local electrochemical reactions coupling the electric field and the mass, species, and energy transport.
- Electric field solution in all porous and solid cell components, including ohmic heating in the bulk material.
- Ability to handle H_2 and combined CO/H_2 electrochemistry.
- Inclusion of tortuosity for porous regions
- Treatment of an arbitrary number of electrochemical cells arranged as a stack.
- Significant geometric flexibility for treating planar, tubular, and other nonstandard SOFC configuration.
- Use of non-conformal interface meshing (as long as these interfaces are not the electrolyte interfaces).
- Ability to model high-temperature electrolysis.

3.2. The SOFC With Unresolved Electrolyte Modeling Strategy

To model solid oxide fuel cells (SOFC) with unresolved electrolyte, you need to perform the following:

- Capture the fluid flow, heat transfer, and the mass transfer in the flow channels and in the porous anode and cathode electrodes.
- Model the transport of the current and the potential field in the porous electrodes and in the solid conducting regions.
- Model the electrochemical reactions that take place at the electrolyte/electrode/gaseous species interface.

Figure 3.2 How the SOFC With Unresolved Electrolyte Model Works in ANSYS FLUENT

3.3. Modeling Fluid Flow, Heat Transfer, and Mass Transfer

All aspects of fluid flow, heat transfer, and mass transfer in the flow channels and porous electrodes are handled by ANSYS FLUENT.

The default multicomponent diffusion model in ANSYS FLUENT is used to calculate the mass diffusion coefficient of species i in the mixture.

To account for the effect of porosity on the multicomponent mass diffusion coefficient

$$D_{ij,eff} = \frac{\varepsilon}{\tau} D_{ij} \quad (3-1)$$

where ε is the porosity and τ is the tortuosity (i.e., the average path length over the actual length).

3.4. Modeling Current Transport and the Potential Field

Solving for three-dimensional electrical conduction is directly analogous to the calculation of heat transfer. The potential field throughout the conductive regions is calculated based on the conservation of charge.

$$\nabla \cdot i = 0 \quad (3-2)$$

where

$$i = -\sigma \nabla \phi \quad (3-3)$$

and σ is the electrical conductivity and ϕ is the electrical potential. Therefore, the governing equation for the electric field is the Laplace equation:

$$\nabla \cdot (\sigma \nabla \phi) = 0 \quad (3-4)$$

The electric field potential calculation combines the following attributes:

- Ohmic losses in all the conducting materials, including the electrolyte, electrodes, and current collectors.
- Contact resistance at the appropriate interfaces.
- Ohmic heating through conduction materials as the result of ohmic losses, and the current density throughout the domain.

For more information, please see the following sections:

[3.4.1. Cell Potential](#)

[3.4.2. Activation Overpotential](#)

[3.4.3. Treatment of the Energy Equation at the Electrolyte Interface](#)

[3.4.4. Treatment of the Energy Equation in the Conducting Regions](#)

3.4.1. Cell Potential

The electrode reactions are assumed to take place in a single step. The charge transfer reaction acts as the rate limiting step for the electrode reactions.

Oxygen is electrochemically reduced at the triple phase boundary at the cathode electrode:



Oxygen is electrochemically re-oxidized at the triple phase boundary at the anode electrode:



In the absence of an electrical load, the oxygen activity on both sides of the electrolyte is fixed and given by their respective chemical potentials. Under equilibrium, the electromotive force, or reversible cell voltage, is given by the Nernst equation:

$$\phi_{ideal} = \frac{RT}{4F} \ln \frac{P_{O_2, cathode}}{P_{O_2, anode}} \quad (3-7)$$

If hydrogen is present at the anode electrode, then the cell reaction becomes:



At equilibrium, the cell voltage is given by the Nernst equation:

$$\phi_{ideal} = \phi^o + \frac{RT}{2F} \ln \frac{P_{H_2} P_{O_2}^{\frac{1}{2}}}{P_{H_2O}} \quad (3-9)$$

The cell potential measured at equilibrium (i.e., no load), is called the open circuit voltage. The open circuit voltage should be equivalent to the Nernst potential at no load, unless there is leakage across the electrolyte. When the external circuit is closed, then cell voltage drops due to polarization losses at the electrodes.

The electric field and the electrochemistry interact solely at the electrolyte interface. ANSYS FLUENT treats the electrolyte interface as an impermeable wall. The potential field must have a "jump" condition applied to the two sides of this wall to account for the effect of the electrochemistry. To closely couple the electrochemical behavior to the potential field calculation, you need to include all of the electrochemical effects into this jump condition. It encapsulates the voltage jump due to Nernst, the voltage reduction due to activation, the Ohmic losses due to the resistivity of the electrolyte, and a linearized for voltage reduction due to activation. This interface condition relates the potential on the anode side and the cathode side of the electrolyte and has the following form:

$$\phi_{cell} = \phi_{jump} - \eta_s \quad (3-10)$$

where

$$\phi_{jump} = \phi_{ideal} - \eta_{ele} - \eta_{act,a} - \eta_{act,c} \quad (3-11)$$

where η_{ele} represents the ohmic overpotential of the electrolyte, and $\eta_{act,a}$, $\eta_{act,c}$ represent the activation overpotential of the anode and the cathode. η_s represent ohmic losses in the solid conducting regions.

ϕ_{ideal} represents the Nernst potential.

3.4.2. Activation Overpotential

The general electrochemical reaction is, according to [6] (p. 79),

$$\sum_j^N a_j A_j \Leftrightarrow n e^- \quad (3-12)$$

where a_j is the stoichiometric coefficient of species j , A_j is the chemical species, and n is the number of electrons.

The reaction rate is:

$$r = \frac{i}{nF} = k_a e^{\frac{\alpha_a n F}{RT} \phi} \prod_i c_i^{p_i} - k_c e^{-\frac{\alpha_c n F}{RT} \phi} \prod_i c_i^{q_i} \quad (3-13)$$

where ϕ is the voltage, k_a and p_i are the rate constant and the reaction order for the anodic direction, k_c and q_i are the rate constant and the reaction order for the cathodic direction, α_a is the anodic transfer coefficient, α_c is the cathodic transfer coefficient, and n is the number of electrons that are released. At equilibrium, the forward and the backward reaction rates are the same, therefore:

$$\frac{i_0}{nF} = k_a e^{\frac{\alpha_a n F}{RT} \phi_0} \prod_i c_i^{p_i} = k_c e^{-\frac{\alpha_c n F}{RT} \phi_0} \prod_i c_i^{q_i} \quad (3-14)$$

where i_0 is the exchange current density.

The reaction rate (i.e., current) can be written in terms of the exchange current density i_0 to obtain the Butler-Volmer formulation [6] (p. 79):

$$i = i_0 \left[e^{\frac{\alpha_a n (\phi - \phi_0) F}{RT}} - e^{-\frac{\alpha_c n (\phi - \phi_0) F}{RT}} \right] \quad (3-15)$$

The activation overpotential is the energy lost due to the slowness of electrochemical reactions at the anode and the cathode electrodes.

$$\eta_{act} = \phi - \phi_0 \quad (3-16)$$

Using this relation, the Butler-Volmer equation can be written as:

$$i = i_{0eff} \left[e^{\frac{\alpha_a n \eta_{act} F}{RT}} - e^{-\frac{\alpha_c n \eta_{act} F}{RT}} \right] \quad (3-17)$$

where

$$i_{0eff} = i_{0,ref} \left(\frac{\chi_j}{\chi_{j,ref}} \right)^{\gamma_j} \quad (3-18)$$

with $i_{0,ref}$ being the exchange current density at the reference condition, (χ_j) is the mole fraction and γ_j is the concentration exponent for species j . More specifically, at the anode side, you have:

$$i_{0eff}^{anode} = i_{0,ref}^{anode} \left(\frac{\chi_{H_2}}{\chi_{H_2,ref}} \right)^{\gamma_{H_2}} \left(\frac{\chi_{H_2O}}{\chi_{H_2O,ref}} \right)^{\gamma_{H_2O}} \quad (3-19)$$

Likewise, at the cathode side, you have:

$$i_{0eff}^{cathode} = i_{0,ref}^{cathode} \left(\frac{\chi_{O_2}}{\chi_{O_2,ref}} \right)^{\gamma_{O_2}} \quad (3-20)$$

Given values for α_a and α_c , the full version of the Butler-Volmer equation can be solved using the Newton method, therefore finding the activation overpotential at the anode ($\eta_{act,a}$) and the cathode ($\eta_{act,c}$).

3.4.3. Treatment of the Energy Equation at the Electrolyte Interface

For an incompressible flow, the energy equation that ANSYS FLUENT solves for within each computational cell is given by the following:

$$\frac{\partial}{\partial t} (\rho E) + \nabla \cdot (\bar{v} (\rho E + p)) = \nabla \cdot \left(k_{eff} \nabla T - \sum_j h_j \bar{J}_j + (\bar{\tau}_{eff} \cdot \bar{v}) \right) + S_h \quad (3-21)$$

where S_h is the volumetric source or sink of energy and where

$$E = h - \frac{p}{\rho} + \frac{v^2}{2} \quad (3-22)$$

and

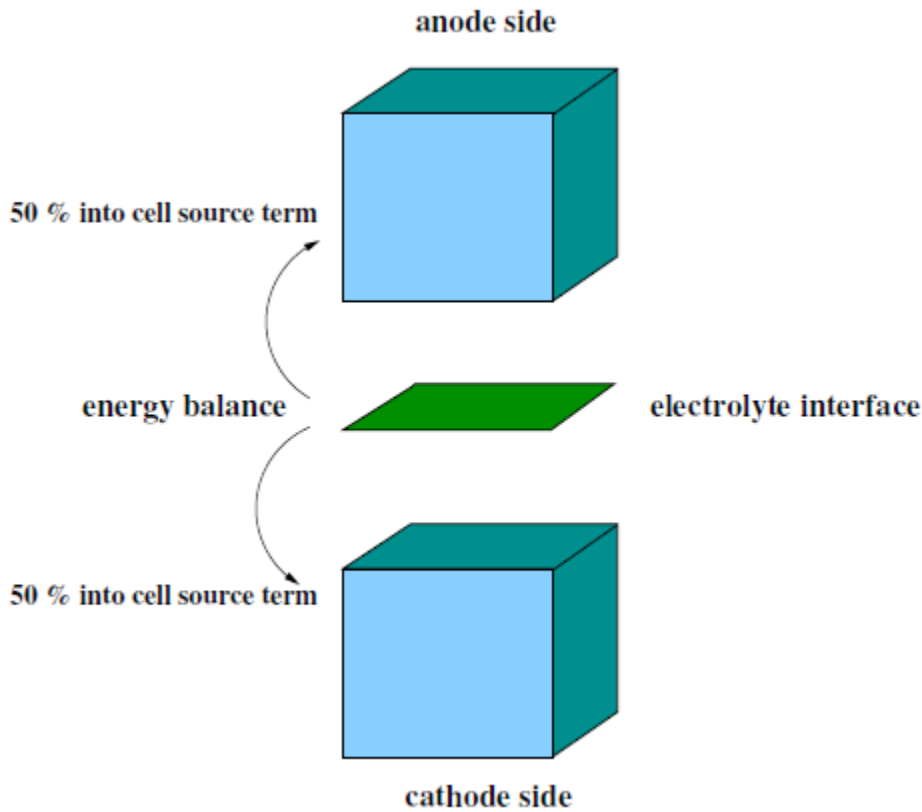
$$h = \sum_j Y_j h_j \quad (3-23)$$

In all electrically conducting zones (e.g., electrodes, current collectors, interconnects), ohmic heating, $i^2 * R_{ohmic}$, is added to the energy equation as a source term. In other words,

$$S_h = i^2 * R_{ohmic} \quad (3-24)$$

In addition, the energy equation needs treatment at the electrode-electrolyte interface to account for the heat generated or lost as the result of electrochemistry and the overpotentials (i.e., activation overpotential and ohmic loss through the electrolyte).

Figure 3.3 Energy Balance at the Electrolyte Interface



The total energy balance on the electrolyte interface is computed by enumerating the enthalpy flux of all species, including the heat of formation (sources of chemical energy entering the system), and then subtracting off the work done (leaving the system) which is simply the local voltage jump multiplied by the local current density. What remains is the waste heat due to irreversibilities. For hydrogen reaction, the balance would be

$$\dot{Q}'' = h''_{H_2} + h''_{O_2} - h''_{H_2O} - i \Delta V \quad (3-25)$$

where Q is the heat generation (W) and h is the total enthalpy of species (J/s) composed of the sensible enthalpy in addition to the enthalpy of formation.

The heat of formation is

$$h_{H2} = \dot{m}_{H2} \left[\int_{T_{ref}}^T C_p dT + h_0 \right] \quad (3-26)$$

The source term is then added in the cell energy equation by taking $S_h = \frac{Q}{Volume}$.

One half of this value is applied as a source term to the energy equation of the anode computational cell adjacent to the electrolyte and the other half is applied as a source term to the energy equation for the cathode cell adjacent to the electrolyte. The equal distribution of the heat generation/destruction is purely arbitrary. Note that by using the work term, the effect from all overpotentials are taken into account.

3.4.4. Treatment of the Energy Equation in the Conducting Regions

Ohmic polarization involves ionic losses through the electrolyte, electrical resistance in the conducting porous electrodes and solid collectors. The ohmic polarization also includes the electrical resistance at the interface of the current collectors and the electrodes or the electrodes and the membrane (i.e., the contact resistance).

$$\eta_{ohmic} = i \cdot R \quad (3-27)$$

3.5. Modeling Reactions

The ANSYS FLUENT SOFC With Unresolved Electrolyte Model can model both electrochemical reactions, as well as *CO* electrochemistry.

For more information, please see the following sections:

[3.5.1. Modeling Electrochemical Reactions](#)

[3.5.2. Modeling CO Electrochemistry](#)

3.5.1. Modeling Electrochemical Reactions

The rate of species production and destruction is:

$$S = -\frac{ai}{nF} \text{ (g - mole/m}^2\text{/sec)} \quad (3-28)$$

where S is the source or sink of the species (molar flux), a is the stoichiometric coefficient, i is the current density (A/m^2), n is the number of electrons per mole of fuel, and F is the Faraday constant.

Using the local current information, the ANSYS FLUENT SOFC With Unresolved Electrolyte Model applies species fluxes to the electrode boundaries. By convention [6] (p. 79), the current density is positive when it flows from the electrode into the electrolyte solution. The current densities are positive at the anodes and negative at the cathodes.

The reaction at the cathode electrode is:



or

$$S_{O_2} = -\frac{-\left(\frac{1}{2}\right)(-i)}{2F} = -\frac{i}{4F} \quad (3-30)$$

$$S_{O^{2-}} = -\frac{(-i)}{2F} = -\frac{i}{2F} \quad (3-31)$$

The reaction at the anode electrode is:



$$S_{H_2} = -\frac{i}{2F} \quad (3-33)$$

$$S_{O^{2-}} = -\frac{i}{2F} \quad (3-34)$$

$$S_{H_2O} = -\frac{(1)(i)}{2F} = \frac{i}{2F} \quad (3-35)$$

3.5.2. Modeling CO Electrochemistry

In practice, if carbon monoxide (CO) is present in the anode fuel stream, it may be oxidized to generate carbon dioxide (CO_2). This effect is modeled by introducing a H_2/CO split factor as follows:

$$\alpha = \frac{x_{H_2}}{x_{H_2} + x_{CO}} \quad (3-36)$$

where x_{H_2} and x_{CO} are species mole fractions of H_2 and CO , respectively.

The molar source terms for each anode-side species are:

$$S_{H_2} = -\frac{\alpha}{2F}i \quad (3-37)$$

$$S_{CO} = -\frac{1-\alpha}{2F}i \quad (3-38)$$

$$S_{CO_2} = \frac{1-\alpha}{2F}i \quad (3-39)$$

$$S_{H_2O} = \frac{\alpha}{2F}i \quad (3-40)$$

By default, [Equation 3-36 \(p. 58\)](#) is used to compute the H_2/CO split factor, however, you can define your own split factor in the user-defined function called `h2_co_split_func()` (see [User-Accessible Functions for the Solid Oxide Fuel Cell With Unresolved Electrolyte Model \(p. 75\)](#)).

Chapter 4: Using the Solid Oxide Fuel Cell With Unresolved Electrolyte Model

The procedure for setting up and solving solid oxide fuel cell (SOFC) problems (with unresolved electrolyte) is described in detail in this chapter. Please refer to the following sections for more information:

- 4.1. Installing the Solid Oxide Fuel Cell With Unresolved Electrolyte Model
- 4.2. Loading the Solid Oxide Fuel Cell With Unresolved Electrolyte Module
- 4.3. Solid Oxide Fuel Cell With Unresolved Electrolyte Module Set Up Procedure
- 4.4. Setting the Parameters for the SOFC With Unresolved Electrolyte Model
- 4.5. Setting Up the Electrochemistry Parameters
- 4.6. Setting Up the Electrode-Electrolyte Interfaces
- 4.7. Setting Up the Electric Field Model Parameters
- 4.8. User-Accessible Functions for the Solid Oxide Fuel Cell With Unresolved Electrolyte Model
- 4.9. Using the Solid Oxide Fuel Cell With Unresolved Electrolyte Text User Interface

4.1. Installing the Solid Oxide Fuel Cell With Unresolved Electrolyte Model

The Solid Oxide Fuel Cell (SOFC) With Unresolved Electrolyte Model is provided as an add-on module with the standard ANSYS FLUENT licensed software. A special license is required to use the SOFC With Unresolved Electrolyte Model. The module is installed with the standard installation of ANSYS FLUENT in a directory called `addons/sofc` in your installation area. The SOFC With Unresolved Electrolyte Model consists of a UDF library and a pre-compiled scheme library, which needs to be loaded and activated before calculations can be performed.

4.2. Loading the Solid Oxide Fuel Cell With Unresolved Electrolyte Module

The Solid Oxide Fuel Cell (SOFC) With Unresolved Electrolyte Model is loaded into ANSYS FLUENT through the text user interface (TUI). The module can only be loaded after a valid ANSYS FLUENT mesh or case file has been set or read. The text command to load the add-on module is

```
define → models → add-on-module
```

A list of ANSYS FLUENT add-on modules is displayed:

```
> /define/models/addon-module
FLUENT  Addon Modules:
  0. None
  1. MHD Model
  2. Fiber Model
  3. Fuel Cell and Electrolysis Model
  4. SOFC Model with Unresolved Electrolyte
  5. Population Balance Model
  6. Adjoint Solver
  7. Battery Module
Enter Module Number: [0] 4
```

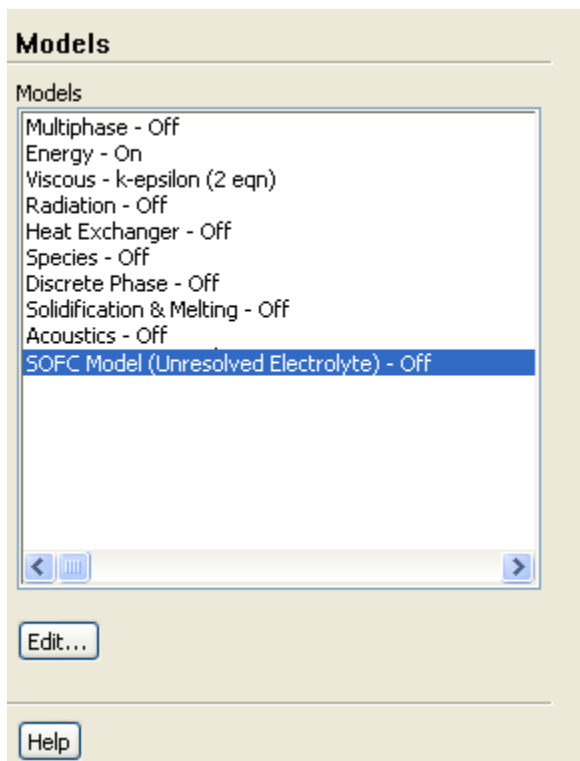
Select the SOFC With Unresolved Electrolyte Model by entering the module number 4. During the loading process, a scheme library (containing the graphical and text user interface) and a UDF library (containing a set of user defined functions) are loaded into ANSYS FLUENT.

Once the module has been loaded, to open the **SOFC Model** dialog box, select **Models** under **Problem Setup** in the navigation pane to display the **Models** task page. In the **Models** task page, select the **SOFC Model (Unresolved Electrolyte)** option in the **Models** list and click the **Edit...** button.

Models → SOFC Model (Unresolved Electrolyte) → Edit...

When the **SOFC Model** dialog box appears, select the **Enable SOFC Model** option.

Figure 4.1 Opening the SOFC Model Dialog Box in the Models Task Page



4.3. Solid Oxide Fuel Cell With Unresolved Electrolyte Module Set Up Procedure

The following describes an overview of the procedure required in order to use the SOFC With Unresolved Electrolyte Model in ANSYS FLUENT.

1. Start ANSYS FLUENT.

You must start ANSYS FLUENT in 3d double-precision mode. Note that the SOFC With Unresolved Electrolyte Model is only available in 3d.

2. Read the case file.

File → **Read** → **Case...**

3. Scale the mesh.

 **General** → **Scale...**

4. Define various model parameters for the simulation.
 - a. Check the solver settings.

 **General**

- i. In the **Solver** group box, enable **Pressure Based** under **Type**.
 - ii. Enable **Implicit** under **Formulation**.
 - iii. Enable **Steady** under **Time**.
 - iv. Enable **Absolute** under **Velocity Formulation**.
- b. Open the **Energy** dialog box and enable the **Energy** option (if it is not already enabled).

 **Models** →  **Energy** → **Edit...**

Enable the **Energy Equation** option, if it is not already enabled.

- c. Open the **Viscous Model** dialog box and enable the **Laminar** option.

 **Models** →  **Viscous** → **Edit...**

Enable the **Laminar** option, if it is not already enabled.

- d. Open the **Species Model** dialog box.

 **Models** →  **Species** → **Edit...**

- i. Enable the **Species Transport** option.
 - ii. Enable the **Volumetric** option under **Reactions**.
 - iii. Disable the **Inlet Diffusion** option under **Options**.
 - iv. Enable the **Diffusion Energy Source** option under **Options**.
 - v. Enable the **Full Multicomponent Diffusion** option under **Options**.
 - vi. Enable the **Thermal Diffusion** option under **Options**.
- e. Open the **SOFC Model** dialog box to set the parameters for the SOFC With Unresolved Electrolyte Model.

 **Models** →  **SOFC Model (Unresolved Electrolyte)** → **Edit...**

Enable the **SOFC Model** option.

Open the **Model Parameters** tab.

- i. Set the **Current Underrelaxation Factor** to a value of either 0 . 3 or 0 . 4.
 - ii. Set the **Electrical and Electrolyte Parameters** according to your problem specification.
- f. Set the electrochemistry parameters.

Open the **Electrochemistry** tab of the **SOFC Model** dialog box.

- i. Enter values for the **Constant Exchange Current Densities** for the anode and the cathode. These are the i_0 values in the Butler-Volmer equation for the anodic and cathodic reactions. By default, the **Anode Exchange Current Density** is set to 1000 Amps and the **Cathode Exchange Current Density** is set to 100 Amps.
- ii. Enter values for the **Mole Fraction Reference Values**.

These are the species concentrations at which the exchange current densities were taken. These are used to adjust the i_0 values of reactant species that are depleted. By default, the **H2 Reference Value** is set to 0.8 moles/moles, the **O2 Reference Value** is set to 0.21 moles/moles, and the **H2O Reference Value** is set to 0.2 moles/moles.
- iii. Enter values for the **Stoichiometric Exponents** by setting values for the **H2 Exponent**, the **H2O Exponent**, and the **O2 Exponent** (defaulted to 0.5). These are the stoichiometric factors in the electrochemical reaction equation. They are used as exponents as part of the i_0 scaling.
- iv. Enter values for the **Butler-Volmer Transfer Coefficients** by setting values for the **Anodic Transfer Coefficient** and the **Cathode Transfer Coefficient** for both the anode reaction and the cathode reaction (defaulted to 0.5). These are the alpha values in the Butler-Volmer equation. They represent the forward and backward rates of reaction at both the anode and cathode.
- v. Enter a values for the **Temperature Dependent Exchange Current Density** by turning on the **Enable Temperature Dependent I_0** option and setting values for **A** and **B**. These two coefficients allow the i_0 for the cathode to vary as a function of temperature.

- g. Set the anode interface components for the SOFC With Unresolved Electrolyte Model.

Open the **Electrode and Tortuosity** tab of the **SOFC Model** dialog box.

- i. Under **Anode Electrolyte**, specify a zone in the **Zone(s)** list.
- ii. Enable **Anode Interface** if it is applicable.

- h. Set the cathode interface components for the SOFC With Unresolved Electrolyte Model.

Open the **Electrode and Tortuosity** tab of the **SOFC Model** dialog box.

- i. Under **Cathode Electrolyte**, specify a zone in the **Zone(s)** list.
- ii. Enable **Cathode Interface** if it is applicable.

- i. Set the tortuosity parameters for the SOFC With Unresolved Electrolyte Model.

Open the **Electrode and Tortuosity** tab of the **SOFC Model** dialog box.

- i. Under **Tortuosity Zone**, specify a zone in the **Zone(s)** list.
- ii. Turn on **Enable Tortuosity** if it is applicable.

- j. Set the parameters for the electric field model.

Open the **Electric Field** tab of the **SOFC Model** dialog box.

- i. Specify up to 5 conductive regions.
- ii. Specify up to 3 contact surfaces.
- iii. Specify a voltage tap surface.
- iv. Specify a current tap surface.

5. Define material properties.

- a. Define a user-defined scalar.
 - i. Open the **User-Defined Scalars** dialog box.
Define → **User-Defined** → **Scalars...**
 - ii. Change the **Number of User-Defined Scalars** to 1.
 - iii. Indicate **none** as the **Flux Function** and click **OK** to close the **User-Defined Scalars** dialog box.
- b. Define 14 user-defined memory locations.
 - i. Open the **User-Defined Memory** dialog box.
Define → **User-Defined** → **Memory...**
 - ii. Change the **Number of User-Defined Memory Locations** to 14.
 - iii. Click **OK** to close the **User-Defined Memory** dialog box.
- c. Define user-defined function hooks.
 - i. Open the **User-Defined Function Hooks** dialog box.
Define → **User-Defined** → **Function Hooks...**
 - ii. Change the **Adjust** function to `adjust_function`.
 - iii. Click **OK** to close the **User-Defined Function Hooks** dialog box.
- d. Create or re-define new solid materials as appropriate for the anode, the cathode, and the electrolyte according to your problem specification.

Important


Note that although the ANSYS FLUENT SOFC With Unresolved Electrolyte Model does support the shell conduction model that you use to take into account the transversal conductive heat inside the electrolyte material, it does not currently support a similarly transversal conduction of electric current inside the electrolyte.

- e. Edit the mixture-template mixture material.

Materials → **Create/Edit...**

- i. In the **Create/Edit Materials** dialog box, click **FLUENT Database...** to open the **FLUENT Database Materials** dialog box.
- ii. In the **FLUENT Database Materials** dialog box, make a copy of the **h2** fluid material.
- iii. In the **Create/Edit Materials** dialog box, change the **Material Type** to `mixture` and click **Edit...** for the **Mixture Species**.
- iv. In the **Species** dialog box, arrange the materials under **Selected Species** in the following order: **h2o**, **o2**, **h2**, and **n2**.
- v. In the **Create/Edit Materials** dialog box, change the **Thermal Conductivity** and the **Viscosity** to **ideal-gas-mixing-law**.
- vi. Change the **Mass Diffusivity** to **user-defined** and select **diffusivity::sofc** as the corresponding user-defined function.

- vii. Change the **UDS Diffusivity** to **user-defined** and select **E_Conductivity::sofc** as the corresponding user-defined function.
 - viii. Retain the default values for the other parameters and click **Change/Create**.
6. Set the operating conditions.

 **Boundary Conditions** → **Operating Conditions...**

Retain the default values.

7. Set the boundary conditions.

 **Boundary Conditions**

Define the conditions at the anode, the cathode, the interface between the anode and current collector, the interface between the cathode and the current collector, the anode inlet, and the cathode inlet boundaries according to your problem specification.

Note that sources only need to be hooked to the mass, species, and energy equation in a single fluid zone in order for the ANSYS FLUENT SOFC With Unresolved Electrolyte Model to function properly (but can be hooked to all zones if you so choose). ANSYS FLUENT does not rely on cell or thread referencing in the sources in order to cover the solution domain.

8. Set the multigrid control parameters.

 **Solution Controls** → **Advanced...**

- a. In the **Multigrid** tab of the **Advanced Solution Controls** dialog box, set the cycle type for **h2**, **h2o**, and **o2** to **V-cycle**. For serial calculations, set the cycle type for **Energy** and **User-defined Scalar-0** to **W-cycle** or **F-cycle**. For parallel calculations, select the **F-cycle** option for both.
 - b. Set the **Max Cycles** to **50**.
 - c. Retain the default values for the rest of the parameters.
9. Define the convergence criteria.

 **Monitors** →  **Residuals** → **Edit...**

In the **Residual Monitors** dialog box, set the **Convergence Criterion** for all equations to $1e-08$.

10. Initialize the flow field.

 **Solution Initialization** → **Initialize**

Retain the default values for all parameters.

11. In the **Model Parameters** tab of the **SOFC Model** dialog box, activate the **Enable Surface Energy Source** option, the **Enable Species Sources** option, and the **Disable CO Electrochemistry** option.

If the electrolyte resistivity changes as a function of temperature, then turn on the **Enable Electrolyte Conductivity Submodel**

Table 4.1 User-Defined Memory Allocations

UDM-0	Interface Current Density (A/m^2)
-------	---------------------------------------

UDM-1	Nernst Potential (Volts)
UDM-2	Activation Overpotential (Volts)
UDM-3	Volumetric Ohmic Source (W / m^3)
UDM-4	x Component of the Current Density (A / m^2)
UDM-5	y Component of the Current Density (A / m^2)
UDM-6	z Component of the Current Density (A / m^2)
UDM-7	Electrolyte Voltage Jump
UDM-8	Electrolyte Resistivity (Ohm-m)
UDM-9	Effective Electric Resistance
UDM-10	Anode Activation
UDM-11	Cathode Activation
UDM-12	Electrochemical Source (W / m^3)
UDM-13	Magnitude of Current Density (A / m^2)

Note that UDM-7 and UDM-8 are the linearized values that ANSYS FLUENT uses to solve the potential field and electrochemical coupling. They are necessary to the calculations but do not contain any real physical meaning.

Note that UDM-10 and UDM-11 contain the disaggregated activation polarizations at the anode and cathode.

The current density in UDM-0 is a conservative flux of electricity (A / m^2). Performing an area integral over the electrolyte surface will sum to the total current (A). That value is computed at the electrolyte faces during the electric field solution. The values in UDM-4 through UDM-6 contain cell-centered values of the current density vector. These are not, and cannot be, conservative, so depending on the material conductivity and the geometric configuration, these can sometimes unavoidably produce values that do not match the UDM values. The same issues exist when interpolating velocity values to obtain the mass fluxes.

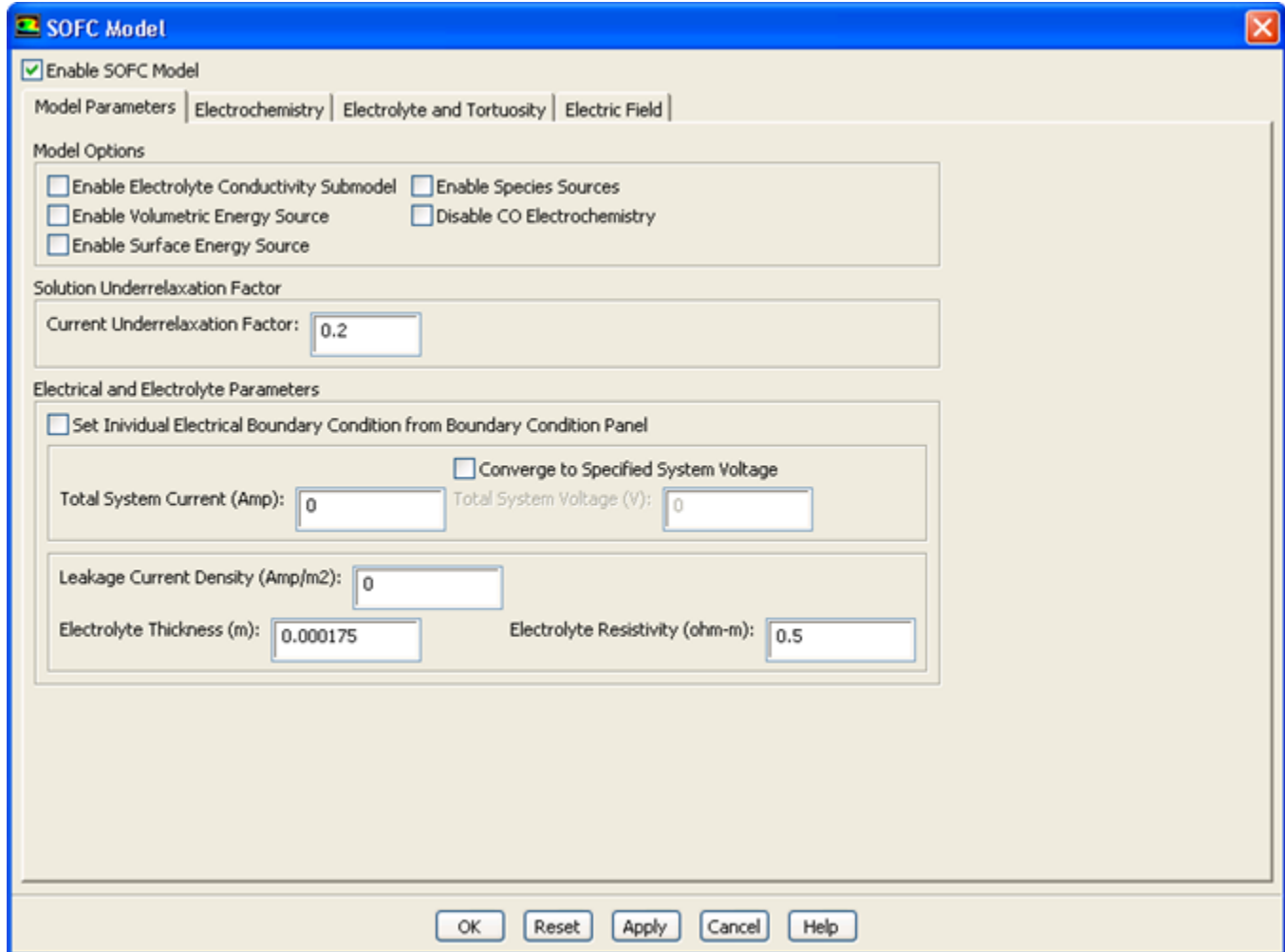
Note that, by default, the ANSYS FLUENT SOFC With Unresolved Electrolyte Model defines a single user-defined scalar:

Table 4.2 User-Defined Scalar Allocations

UDS-0	Electric Potential (Volts)
-------	----------------------------

4.4. Setting the Parameters for the SOFC With Unresolved Electrolyte Model

You can specify the general settings for the SOFC With Unresolved Electrolyte Model using the **Model Parameters** tab in the **SOFC Model** dialog box.

Figure 4.2 The Model Parameters Tab in the SOFC Model Dialog Box

From this tab, you can set the various parameters for the SOFC With Unresolved Electrolyte Model such as total system current, electrolyte thickness, electrolyte resistivity, etc.

The **Enable Electrolyte Conductivity Submodel** option allows the ionic conductivity (or resistivity) of the electrolyte to change as a function of temperature. At the moment, there is one correlation that provides ionic conductivity (or resistivity) of the electrolyte as function of temperature.

$$\text{resistivity} = \frac{0.3685 + 0.002838e^{\frac{10300}{T}}}{100} \quad (4-1)$$

Important

Note that this is valid only for temperatures ranging from 1073 K to 1373 K.

By turning off the **Enable Surface Energy Source** option, ANSYS FLUENT excludes the heat addition due to electrochemistry and all the reversible processes. This option should be turned on at all times.

The **Enable Volumetric Energy Source** option includes the ohmic heating throughout the electrically conducting zones. You should keep this option turned off (to avoid slowing the convergence rates)

until a certain rate of convergence for the potential field has been achieved, at which point, you should turn the option on manually. Note that this option is important so that the solution can account for the effects of the internal Ohmic heating.

The **Disable CO Electrochemistry** is enabled if there is carbon monoxide (CO) in the fuel line and if you do not want to include the CO in the electrochemistry.

Since the calculations are very sensitive to large current fluctuations early in the solution process, it is recommended to use 0.3 or 0.4 for the **Current Underrelaxation Factor** for a more effective solution.

The leakage current is the total amount of current due to the leakage of oxidizer to the fuel side (through the electrolyte) and the electric current across the electrolyte due to any short circuit. You can specify a value for the leakage current under **Leakage Current Density**.

If the leakage current density is temperature-dependent, you can specify your own temperature-dependent implementation of the leakage current density by performing the following steps:

1. Make the required changes to `Leakage_Current_Density (real T)` which is a real function in the user-modifiable source code, `constit.c`.
2. Compile `constit.c` and make your own `sofc` UDF library.

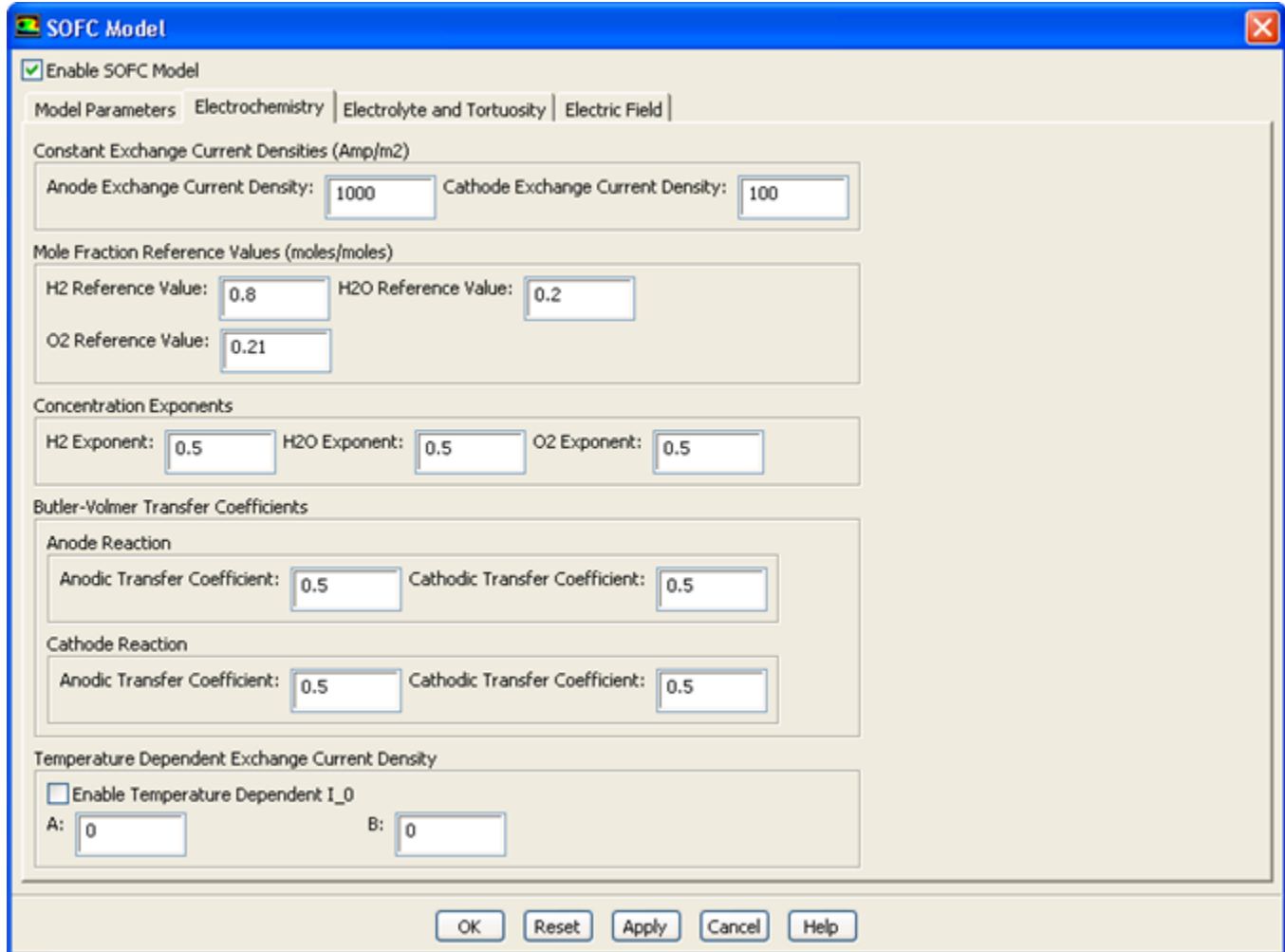
For more information, see [Compiling the Customized Solid Oxide Fuel Cell With Unresolved Electrolyte Source Code](#) (p. 76).

The **Converge to Specified System Voltage** is used when the you want to specify a system voltage instead of system current as an input.

The **Set Individual Electrical Boundary Condition from Boundary Conditions Task Page** option, when enabled, allows you to directly specify the current density or voltage for each individual current-collecting boundary using the ANSYS FLUENT **Boundary Conditions** task page. This allows you to apply multiple current types or multiple voltage types (either constant or variable) to your boundary conditions.

4.5. Setting Up the Electrochemistry Parameters

You can specify electrochemistry settings for the SOFC With Unresolved Electrolyte Model using the **Electrochemistry** tab in the **SOFC Model** dialog box.

Figure 4.3 The Electrochemistry Tab in the SOFC Model Dialog Box

In the **Electrochemistry** tab, you can set the anode and cathode exchange current density, the anode and cathode mole fraction reference values, the concentration exponents, the Butler-Volmer transfer coefficients, and the temperature-dependent exchange current density.

You can specify a value for the exchange current density at the anode (the default value is $1000 \text{ Amps} / \text{m}^2$) using the **Anode Exchange Current Density** field. Likewise, you can specify a value for the exchange current density at the cathode (the default value is $100 \text{ Amps} / \text{m}^2$) using the **Cathode Exchange Current Density** field.

You can also specify **Mole Fraction Reference Values** for the fuel cell reactants in the **Electrochemistry** tab. By default, the reference value for H_2 is 0.8, the reference value for O_2 is 0.21, and the reference value for H_2O is 0.2.

The Butler-Volmer transfer coefficients can be set in the **Electrochemistry** tab as well. These coefficients are the α_a and α_c from *Equation 3-15* (p. 54) for both the anode and the cathode reactions.

$$i = i_0 \left[e^{\frac{\alpha_a n (\phi - \phi_0) F}{RT}} - e^{-\frac{\alpha_c n (\phi - \phi_0) F}{RT}} \right] \quad (4-2)$$

Remember that α has anodic and cathodic values at *both* the cathode and the anode. By default, the value of α is set to 0.5 because of the nearly universal assumption that there is a symmetric balance between the forward and backward reactions. In most cases, these default values will be sufficient.

If you find yourself changing the Butler-Volmer transfer coefficients, or if you have some other rate-limiting reaction in your fuel cell simulation, you may also want to consider changing the exponents for the stoichiometric coefficients for the fuel cell reactants. These exponents can be specified in the **Electrochemistry** tab. By default, the exponent values for H_2 , O_2 , and H_2O are 0.5.

The **Enable Temperature Dependant I_0** option allows the exchange current density to change as a function of temperature in an exponential fashion

$$i_0 = A e^{-\frac{1}{BT}} \quad (4-3)$$

where you can provide the values for the constants A and B .

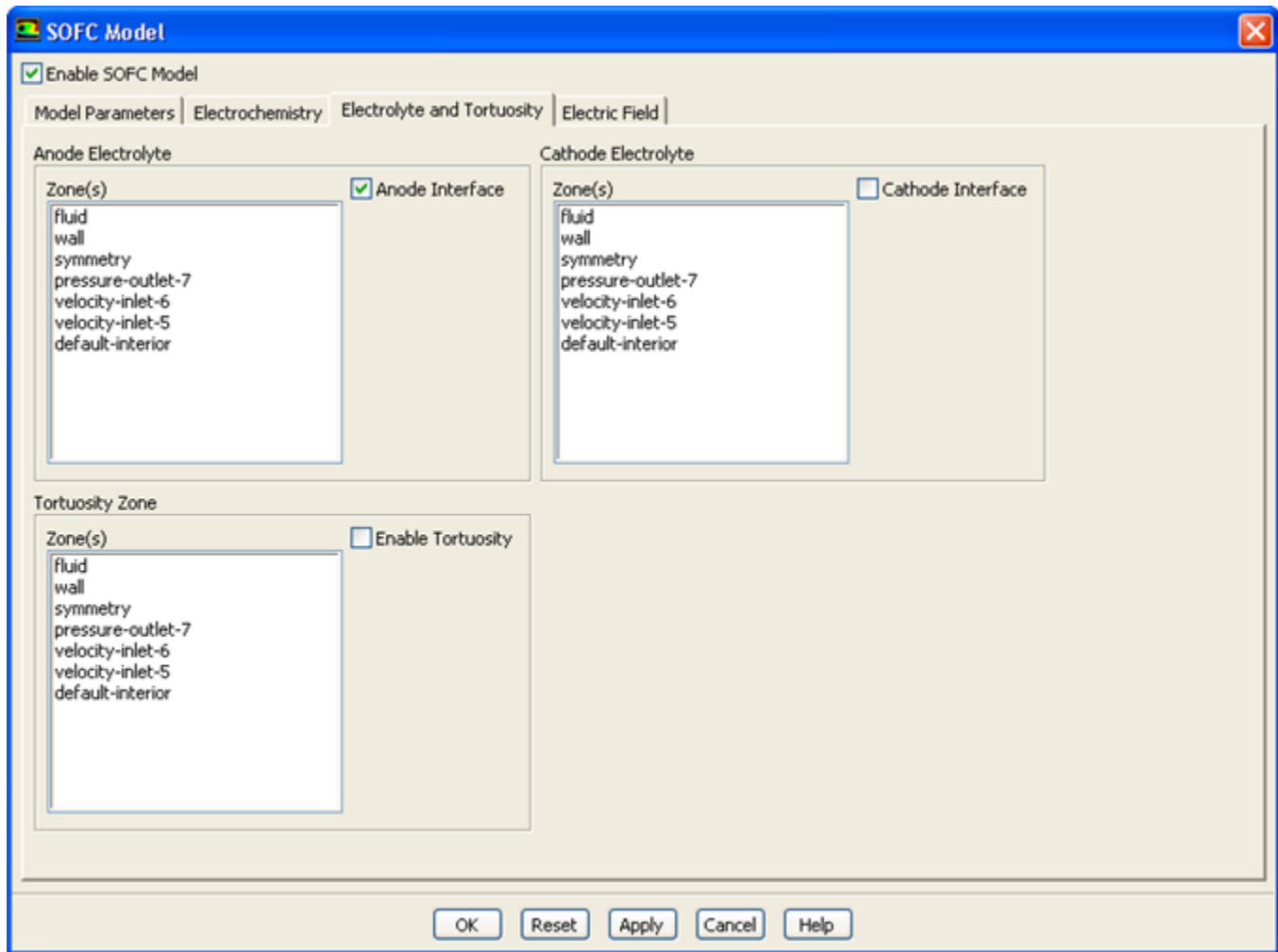
4.6. Setting Up the Electrode-Electrolyte Interfaces

You can apply specific settings for both your anode and your cathode interfaces as well as set tortuosity parameters using the ANSYS FLUENT SOFC With Unresolved Electrolyte Model through the **Electrolyte and Tortuosity** tab in the **SOFC Model** dialog box.

- 4.6.1. [Setting Up the Anode Electrode-Electrolyte Interface](#)
- 4.6.2. [Setting Up the Cathode Electrode-Electrolyte Interface](#)
- 4.6.3. [Setting Up the Tortuosity Parameters](#)

4.6.1. Setting Up the Anode Electrode-Electrolyte Interface

You can specify the anode settings for the SOFC With Unresolved Electrolyte Model by enabling the **Anode Interface** option in the **Electrolyte and Tortuosity** tab in the **SOFC Model** dialog box.

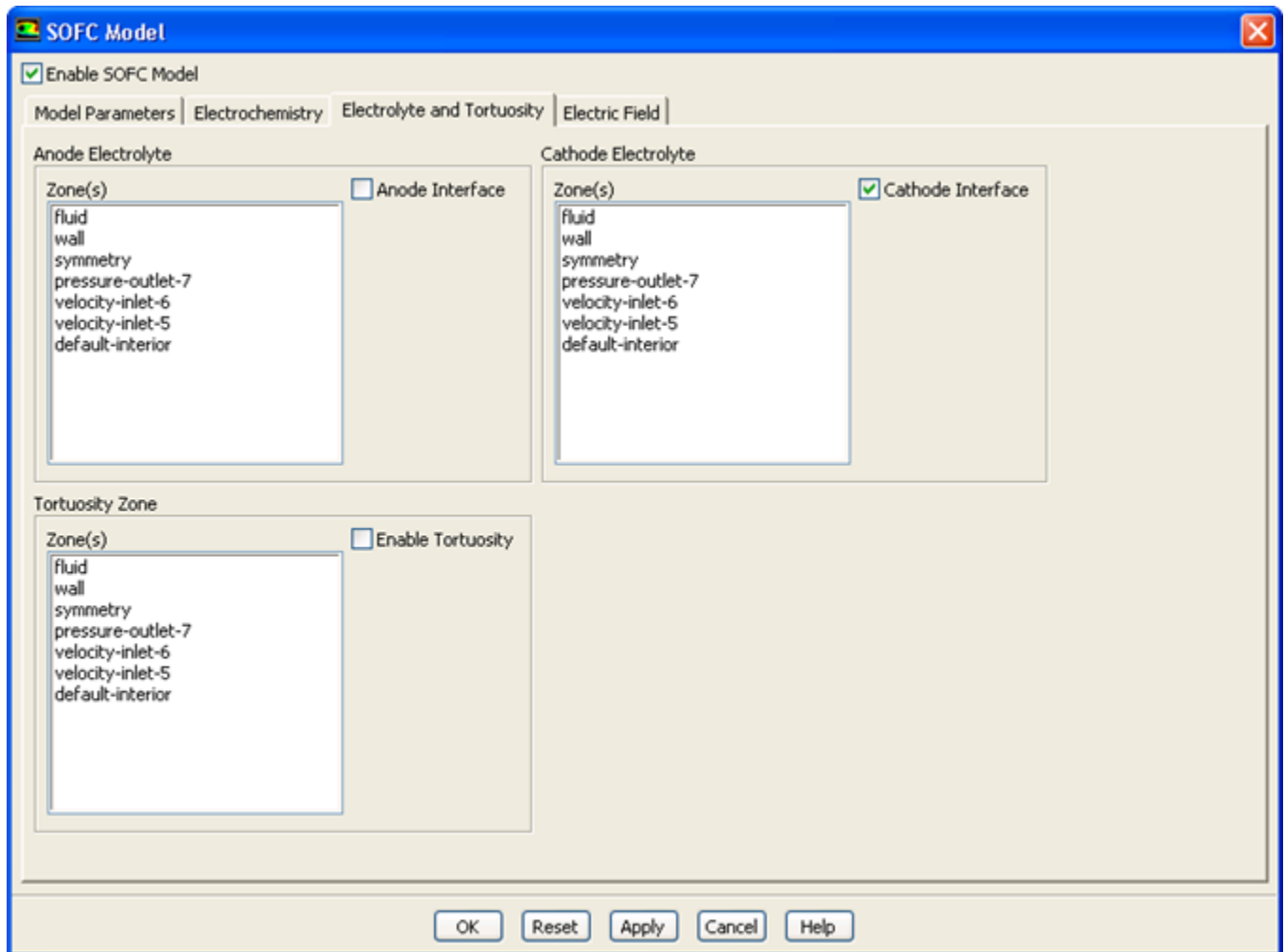
Figure 4.4 The Electrolyte and Tortuosity Tab in the SOFC Model Dialog Box (Anode Interface)

Once you enable the **Anode Interface** option, you select the surface that represents the anode electrode interface with the electrolyte from the corresponding **Zone(s)** list and click **Apply**. You can do this for as many zones as you need. When you are finished setting up the anode interface, click the **OK** button.

4.6.2. Setting Up the Cathode Electrode-Electrolyte Interface

You can specify the cathode settings for the SOFC With Unresolved Electrolyte Model by enabling the **Cathode Interface** option in the **Electrolyte and Tortuosity** tab in the **SOFC Model** dialog box.

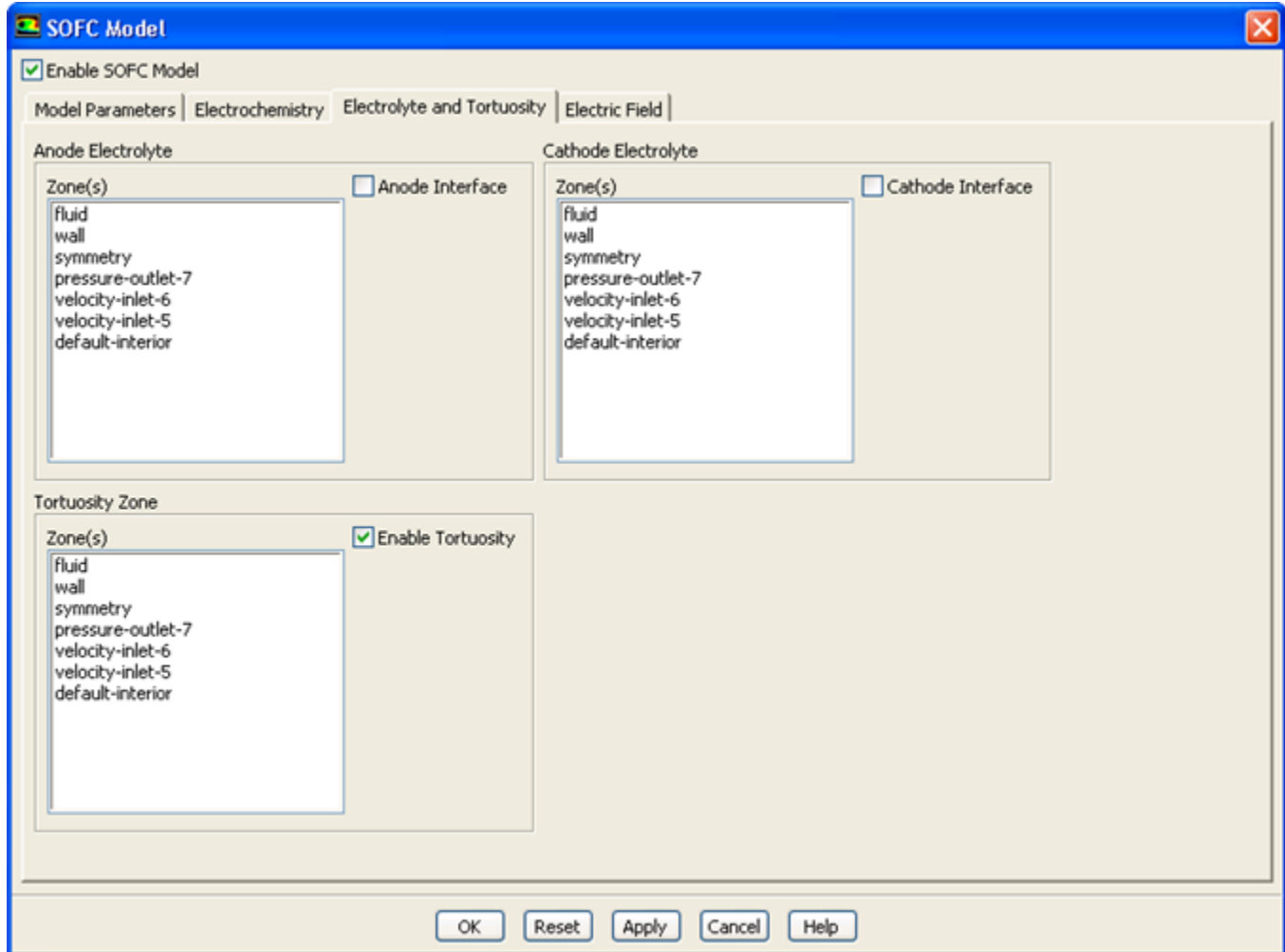
Figure 4.5 The Electrolyte and Tortuosity Tab in the SOFC Model Dialog Box (Cathode Interface)



Once you enable the **Cathode Interface** option, you select the surface that represents the cathode electrode interface with the electrolyte from the corresponding **Zone(s)** list and click **Apply**. You can do this for as many zones as you need. When you are finished setting up the cathode interface, click the **OK** button.

4.6.3. Setting Up the Tortuosity Parameters

In fuel cell modeling, tortuosity is thought of as an effective diffusive path length factor. You can specify tortuosity settings for the SOFC With Unresolved Electrolyte Model by enabling the **Enable Tortuosity** option in the **Electrolyte and Tortuosity** tab in the **SOFC Model** dialog box.

Figure 4.6 The Electrolyte and Tortuosity Tab in the SOFC Model Dialog Box (Tortuosity)

Once you enable the **Enable Tortuosity** option, you select an appropriate zone from the corresponding **Zone(s)** list and click **Apply**. You can do this for as many zones as you need. When you are finished setting up the cathode interface, click the **OK** button.

You can assign tortuosity values to any porous zones in your simulation. In a porous zone, the mass diffusion coefficient is reduced as follows due to the porosity effect:

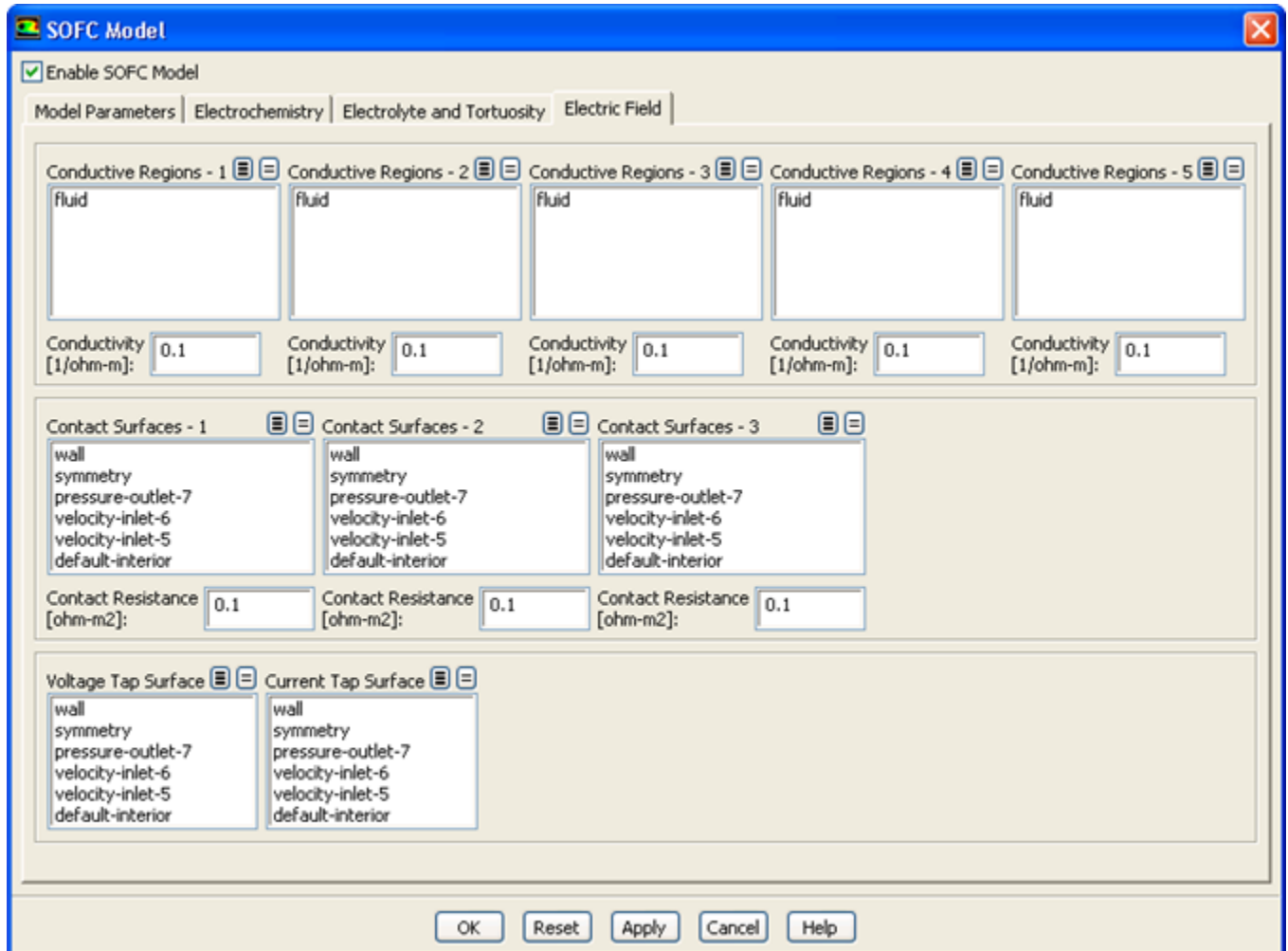
$$D_{eff} = \frac{\text{porosity}}{\text{tortuosity}} D \quad (4-4)$$

There typically are no standard means of measuring tortuosity as it either needs to be measured experimentally or tuned to match other experimental data. Tortuosity value may typically be in the range of 2 to 4, although you can use much higher values.

4.7. Setting Up the Electric Field Model Parameters

You can set up the details of the electric field model using the **Electric Field** tab in the **SOFC Model** dialog box.

Figure 4.7 The Electric Field Tab in the SOFC Model Dialog Box



Here, you can designate conductive regions and specify their conductivity, assign boundaries to be contact surfaces and specify the contact resistances, as well as specify which surfaces are grounded and which surfaces exhibit a current.

The **Voltage Tap Surface** is the surface that is grounded (i.e., $V = 0$). The **Current Tap Surface** is the surface that current is being drawn from (i.e., i is assigned a value).

4.8. User-Accessible Functions for the Solid Oxide Fuel Cell With Unresolved Electrolyte Model

You can directly incorporate your own formulations and data for the properties of the solid oxide fuel cell model using the `constit.c` source code file.

The following listing represents a description of the contents of the `constit.c` source code file:

- `real Nernst (real sp_a[], real sp_c[], real P_a, real P_c, real T_a, real T_c, real *DNDY):` Returns the Nernst potential using [Equation 3–9 \(p. 53\)](#).
- `real Activation (real Y[], real P, real T, real i, real i_0, real alpha_a, real alpha_b, int species, real *dA_di):` Returns the activation overpotential by solving [Equation 3–15 \(p. 54\)](#).

- `real Resistivity (real T)`: Returns the electrolyte resistance (either a constant value, or computed using *Equation 4-1* (p. 68)).
- `real Leakage_Current_Density(realT)`: Returns the leakage current density specified in the **SOFC Model** dialog box and can be overwritten as a function of temperature.
- `real Solve_Butler_Volmer_NR (real i, real i0, real A, real B, real *de-tadi) ;`: Returns the activation overpotential by solving *Equation 3-15* (p. 54).
- `real CONDUCTIVITY_CELL(cell_t c, Thread *t)`: Returns the cell conductivity value used in the **Electric Field** tab of the **SOFC Model** dialog box. This function can be overwritten to look up the conductive zone group ID (1 through 5). A conditional statement allows you to perform various tasks for respective conductive zones. For non-conductive zones, this function returns a value of 0.
- `real CONTACT_RESIST_FACE (face_t f, Thread *t)`: Returns the contact resistance value used in the **Electric Field** tab of the **SOFC Model** dialog box. This function can be overwritten to look up the contact resistance group ID (1 through 3). A conditional statement allows you to perform various tasks for respective contact resistance surfaces. For non-contact resistance surfaces, this function returns a value of 0.
- `real h2_co_split_func(cell_t c_an, Thread *t_an)`: Returns the H_2/CO split factor defined in *Equation 3-36* (p. 58). This can be overwritten as needed to define your own split factor.

For more information, please see the following section:

[4.8.1. Compiling the Customized Solid Oxide Fuel Cell With Unresolved Electrolyte Source Code](#)

4.8.1. Compiling the Customized Solid Oxide Fuel Cell With Unresolved Electrolyte Source Code

This section includes instructions on how to compile a customized Solid Oxide Fuel Cell With Unresolved Electrolyte user-defined module. Note that you can also refer to the file `INSTRUCTIONS-CLIENT` that comes with your distribution (see `addons/sofc`).

Important

It is assumed that you have a basic familiarity with compiling user-defined functions (UDFs). For an introduction on how to compile UDFs, please refer to the separate UDF manual.

You will first want to use a local copy of the `sofc` directory in the `addons` directory before you recompile the Solid Oxide Fuel Cell With Unresolved Electrolyte module.

4.8.1.1. Compiling the Customized Source Code Under Linux

1. Make a local copy of the `sofc` directory. Do not create a symbolic link.

Important

The custom version of the library must be named according to the convention used by ANSYS FLUENT: e.g., `sofc`.

2. Change directories to the `sofc/src` directory.
3. Make changes to the `constit.c` file.

4. Define the `FLUENT_ADDONS` environment variable to correspond to your customized version of the Solid Oxide Fuel Cell With Unresolved Electrolyte module.
5. Change directories to the `sofc/` directory.
6. Issue the following `make` command:

```
make FLUENT_INC=[ansys_inc/v140/fluent] FLUENT_ARCH=[arch]
-f Makefile-client
```

where `your_arch` is `lnx86` on LINUX, or `ultra` on the Sun operating system, etc.

The following example demonstrates the steps required to set up and run a customized version of the Fuel Cell and Electrolysis module that is located in a folder call `home/sample`:

- Make a directory.

```
mkdir -p /home/sample
```

- Copy the default `addon` library to this location.

```
cp -RH [ansys_inc/v140/fluent]/fluent14.0.0/addons/sofc
/home/sample/sofc
```

- Using a text editor, make the appropriate changes to the `constit.c` file located in `/home/sample/sofc/src/constit.c`
- Build the library.

```
cd /home/sample/sofc
make FLUENT_INC=[ansys_inc/v140/fluent] FLUENT_ARCH=[arch]
-f Makefile-client
```

- Set the `FLUENT_ADDONS` environment variable (using CSH, other shells will differ).

```
setenv FLUENT_ADDONS /home/sample
```

- Start ANSYS FLUENT and load the customized module using the text interface command.

4.8.1.2. Compiling the Customized Source Code Under Windows

1. Open **Visual Studio.NET** at the DOS prompt.
2. Make sure that the `$FLUENT_INC` environment variable is correctly set to the current ANSYS FLUENT installation directory (e.g., `ANSYS Inc\v140\fluent`).
3. Make a local copy of the `sofc` folder. Do not create a shortcut.
4. Enter the `sofc\src` folder.
5. Make changes to the `constit.c` file.
6. Define the `FLUENT_ADDONS` environment variable to correspond to your customized version of the Solid Oxide Fuel Cell With Unresolved Electrolyte module.
7. Return to the `sofc` folder.
8. Issue the following command in the command window:

```
nmake /f makefile_master-client.nt
```

4.9. Using the Solid Oxide Fuel Cell With Unresolved Electrolyte Text User Interface

All of the features for the Solid Oxide Fuel Cell With Unresolved Electrolyte Model that are available through the graphical user interface are also available through text user interface commands.

Once the fuel cell module is loaded (see *Loading the Solid Oxide Fuel Cell With Unresolved Electrolyte Module* (p. 61)), you can access the text user interface through the Console Window under `sofc-model`. A listing of the various text commands is as follows:

```
sofc-model/  
SOFC model menu  
  
enable-sofc-model?  
Enable/disable SOFC model  
  
model-parameters  
Set model parameters  
  
electrochemistry  
Set electrochemistry parameters  
  
anode-interface  
Set fuel cell anode interface  
  
cathode-interface  
Set fuel cell cathode interface  
  
tortuosity-interface  
Set fuel cell tortuosity interface  
  
electric-field-model/  
Electric field model  
  
voltage-tap  
Set voltage tap surface  
  
current-tap  
Set current tap surface  
  
conductive-regions  
Set conductive regions  
  
contact-resistance-regions  
Set contact resistance regions
```

Bibliography

- [1] *Fuel Cell Handbook (5th Edition)*. E G & G Services and Parsons Inc. and Science Applications International Corporation. 2000.
- [2] Brandon M. Eaton. "One dimensional, Transient Model of Heat, Mass, and Charge Transfer in a Proton Exchange Membrane". *M.S. Thesis*. 2001.
- [3] A. A. Kulikovskiy, J. Divisek, and A. A. Kornyshev. "Modeling the Cathode Compartment of Polymer Electrolyte Fuel Cells: Dead and Active Reaction Zones". *J. Electrochemical Society*. 146(11). 3981–3991. 1999.
- [4] S. Mazumder and J. V. Cole ".Rigorous 3-D Mathematical Modeling of PEM Fuel Cells II. Model Predictions with Liquid Water Transport". *J. Electrochemical Soc.*. 150(11). A1510–1517. 2003.
- [5] J. H. Nam and M. Karvianiy. "Effective Diffusivity and Water-Saturation Distribution in Single- and Two-layer PEMFC Diffusion Medium". *Int. J. Heat Mass Transfer*. 2003.
- [6] J. S. Newman. *Electrochemical Systems*. Prentice Hall. Englewood Cliffs, New Jersey 1973.
- [7] T. V. Nguyen. "Modeling Two-Phase Flow in the Porous Electrodes of Proton Exchange Membrane Fuel Cells Using the Interdigitated Flow Fields". *Tutorials in Electrochemical Engineering Mathematical Modeling*. 99(14). 222–241. 1999.
- [8] T. E. Springer, T. A. Zawodzinski, and S. Gottesfeld. "Polymer Electrolyte Fuel Cell Model". *J. Electrochemical Soc.*. 138. 2334–2342. 1991.
- [9] Sukkee Um, C. Y. Wang, and K. S. Chen.. "Computational Fluid Dynamics Modeling of Proton Exchange Membrane Fuel Cells". *J. Electrochemical Society*. 147(12). 4485–4493. 2000.

Index

A

- activation overpotential, 53
- advanced properties, 31
- anode
 - catalyst properties, 25
 - cell zone conditions, 25
 - channel properties, 23
 - current collector properties, 22
 - porous electrode properties, 24
 - properties, 22

B

- boundary conditions, 37

C

- cathode
 - catalyst layer properties, 30
 - cell zone conditions, 30
 - current collector properties, 27
 - flow channel properties, 28
 - porous electrode properties, 29
 - properties, 27
- cell potential, 52
- cell zone conditions, 25
 - electrolyte/membrane, 27
- compiling custom source code, 42, 76
- contact resistivities, 31
- conventions used in this guide, vi
- coolant channel properties, 32
- current collector modeling, 35
- current conservation, 7
- current transport, 52

E

- electrochemistry modeling, 4
- Electrolysis, 3
- electrolyte/membrane properties, 26

F

- Fuel Cell and Electrolysis model
 - advanced properties, 31
 - anode
 - catalyst properties, 25
 - cell zone conditions, 25
 - channel properties, 23
 - current collector properties, 22
 - porous electrode properties, 24
 - anode properties, 22

- boundary conditions, 37
- cathode
 - catalyst layer properties, 30
 - cell zone conditions, 30
 - current collector properties, 27
 - flow channel properties, 28
 - porous electrode properties, 29
- cathode properties, 27
- compiling custom source code, 42
- current and mass conservation, 7
- electrochemistry modeling, 4
- electrolyte/membrane cell zone conditions, 27
- electrolyte/membrane properties, 26
- geometry definition, 15
- heat source, 8
- installing, 16
- introduction, 1, 15
- loading, 16
- managing stacks, 33
- modeling current collectors, 35
- modeling fuel cells and electrolysis, 17
- options, 18
- parameters, 20
- postprocessing, 38
- properties, 10
- set up, 16
- setting contact resistivities, 31
- setting coolant channel properties, 32
- solution guidelines, 37
- solution reporting, 35
- specific leakage current, 13
- text interface, 43
 - IV curve calculations, 45
- theory, 1
- transient simulations, 12
- user-accessible functions, 39
- using, 15

- Fuel Cell and Electrolysis Models dialog box, 17

G

- geometry definition, 15

H

- heat source, 8

I

- installing the Fuel Cell and Electrolysis model, 16
- installing the SOFC with Unresolved Electrolyte model, 61

L

loading the Fuel Cell and Electrolysis model, 16
loading the SOFC with Unresolved Electrolyte model, 61

M

mass conservation, 7

P

PEMFC, 2
 water formation and transport, 9
postprocessing, 38
potential field, 52
properties, 10

R

reporting, 35

S

setting up the Fuel Cell and Electrolysis model, 16
setting up the SOFC with Unresolved Electrolyte model, 62
SOFC, 3
SOFC with Unresolved Electrolyte model
 activation overpotential, 53
 cell potential, 52
 compiling custom source code, 76
 current transport and potential field, 52
 electric field model parameters, 74
 electrochemistry parameters, 69
 electrode-electrolyte interfaces, 71
 anode, 71
 cathode, 72
 energy treatment in conducting regions, 57
 energy treatment in electrolyte, 55
 heat and mass transfer, 51
 installing, 61
 introduction, 49
 loading, 61
 parameters, 67
 reactions, 57
 CO electrochemistry, 58
 electrochemical, 57
 set up, 62
 strategy, 50
 text interface, 78
 theory, 49
 tortuosity, 73
 user-accessible functions, 75
 using, 61
solution guidelines, 37

specific leakage current, 13
stacks, 33

T

text interface, 43, 78
 IV curve calculations, 45
transient simulations, 12

U

user-accessible functions, 39