Proton Exchange Membrane Fuel Cell Modeling and Simulation using Ansys Fluent

by

Adam Arvay

A Thesis Presented in Partial Fulfillment
of the Requirements for the Degree
Master of Science in Technology

Approved April 2011 by the
Graduate Supervisory Committee:

Arunachalanadar Madakannan, Chair
Xihong Peng
Yong Liang
James Subach

ARIZONA STATE UNIVERSITY

May 2011

ABSTRACT

Proton exchange membrane fuel cells (PEMFCs) run on pure hydrogen and oxygen (or air), producing electricity, water, and some heat. This makes PEMFC an attractive option for clean power generation. PEMFCs also operate at low temperature which makes them quick to start up and easy to handle. PEMFCs have several important limitations which must be overcome before commercial viability can be achieved. Active areas of research into making them commercially viable include reducing the cost, size and weight of fuel cells while also increasing their durability and performance. A growing and important part of this research involves the computer modeling of fuel cells. High quality computer modeling and simulation of fuel cells can help speed up the discovery of optimized fuel cell components. Computer modeling can also help improve fundamental understanding of the mechanisms and reactions that take place within the fuel cell. The work presented in this thesis describes a procedure for utilizing computer modeling to create high quality fuel cell simulations using Ansys Fluent 12.1. Methods for creating computer aided design (CAD) models of fuel cells are discussed. Detailed simulation parameters are described and emphasis is placed on establishing convergence criteria which are essential for producing consistent results. A mesh sensitivity study of the catalyst and membrane layers is presented showing the importance of adhering to strictly defined convergence criteria. A study of iteration sensitivity of the simulation at low and high current densities is performed which demonstrates the variance in the rate of convergence and the absolute difference between solution values derived at low numbers of iterations and high numbers of iterations.

TABLE OF CONTENTS

Page

LIST OF TABLES

LIST OF FIGURES

Figure Page

Figure Page

CHAPTER 1

INTRODUCTION

Fuel cells are an important technology for the generation of renewable energy. A fuel cell is an energy conversion device which takes some form of fuel and converts it directly into electrical energy. The chemical reactions that take place are dependent on the specific type of fuel cell. There are many different fuel cell types that can be classified by their typical operating temperature.

High temperature fuel cells operate at 600℃ or highe r and offer very high-efficiency and high-power outputs. These high-temperature devices usually require long startup times and should not be cycled on and off frequently due to thermal stresses imposed by the large temperature changes. These types of high-temperature fuel cells are generally used for stationary applications where constant power generation is required. High-temperature fuel cells include solid oxide fuel cells and molten carbonate fuel cells.

Medium-temperature fuel cells operate around 100℃ to 300℃ and offer midrange performance and efficiency. These types of cells have less strenuous material properties but still offer good efficiency. Given their greater ability to cycle on and off, they can be used for stationary or mobile applications. These types of cells include alkaline fuel cells, which are used on the space shuttle, and phosphoric acid fuel cells.

Low-temperature fuel cells can operate at less than 100°C. The low temperature operation of these cells means that they can operate efficiently immediately after startup and in ambient temperature conditions. This makes them ideal for portable applications with transient power needs. The most common type of low-temperature fuel cell is the proton exchange membrane (PEM) fuel cell.

PEM fuel cells convert hydrogen and oxygen to water and electricity and heat. These devices can also be thought of as electrochemical engines and they can be used for any application where electrical power is required. One of the most important applications is for transportation applications. Fuel cells do not need to run on fossil fuels. Hydrogen generated by renewable power sources, such as solar or wind, will produce no greenhouse gasses. PEM Fuel cells running on pure hydrogen produce zero emissions creating a clean, sustainable energy cycle.

The chemical reactions that take place within a PEM fuel cell are as follows:

$$H_2 \rightarrow 2H^+ + 2e^- \text{ (anode reaction)} \qquad (1)$$

$$\tfrac{1}{2} O_2 + 2e^- + 2H^+ \rightarrow H_2O \text{ (cathode reaction)} \qquad (2)$$

Hydrogen gas is introduced to the anode side of the fuel cell where it forms hydrogen ions (protons) and electrons. The electrons flow through the gas diffusion layers to the current collector where they can travel through an external circuit. The ions flow through the membrane to meet the electrons from the external circuit, forming a complete electrical circuit.

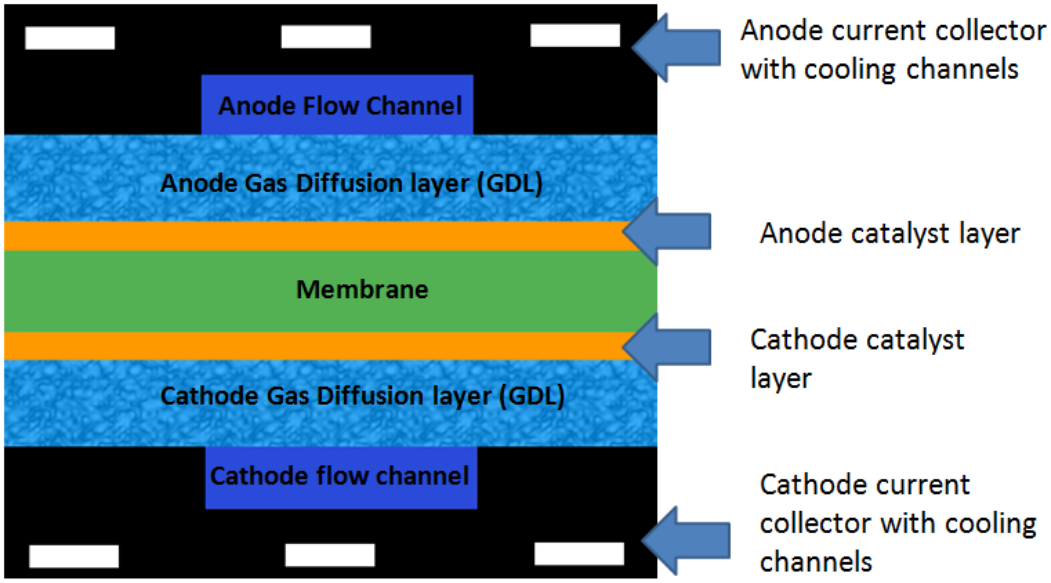

Anode current collector with cooling channels
Anode Flow Channel
Anode Gas Diffusion layer (GDL)
Anode catalyst layer
Membrane
Cathode catalyst layer
Cathode Gas Diffusion layer (GDL)
Cathode flow channel
Cathode current collector with cooling channels

Figure 1. PEM Fuel Cell schematic.

A typical PEM fuel cell is composed of four parts sandwiched across a membrane. These include the current collector, gas diffusion layer (GDL), catalyst layer, and membrane. Each part serves multiple, different purposes and the design and composition of the various parts can have important effects on the performance of a fuel cell.

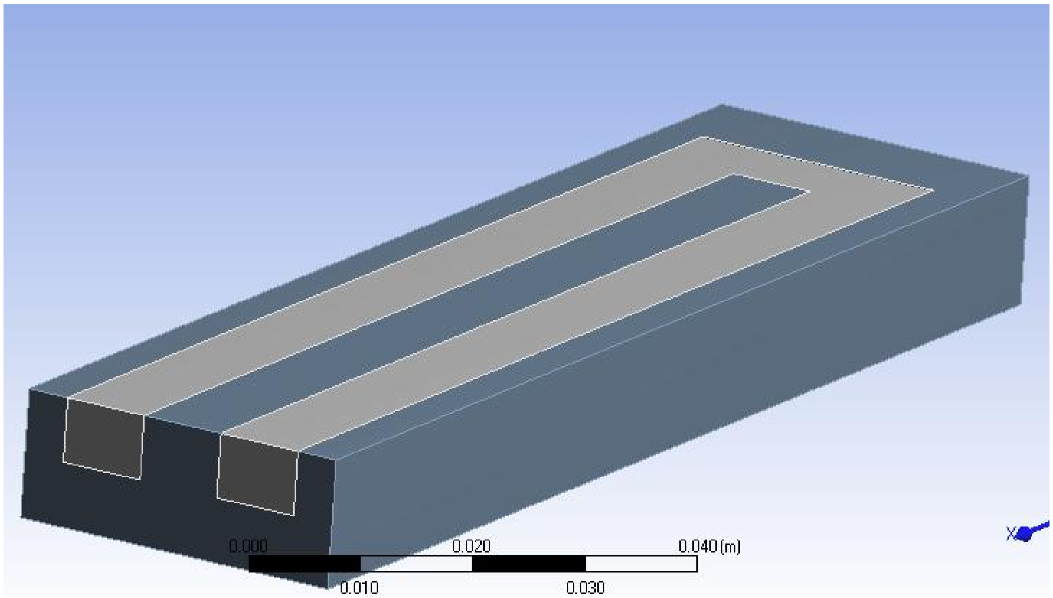


Figure 2. Example current collector with a serpentine flow channel.

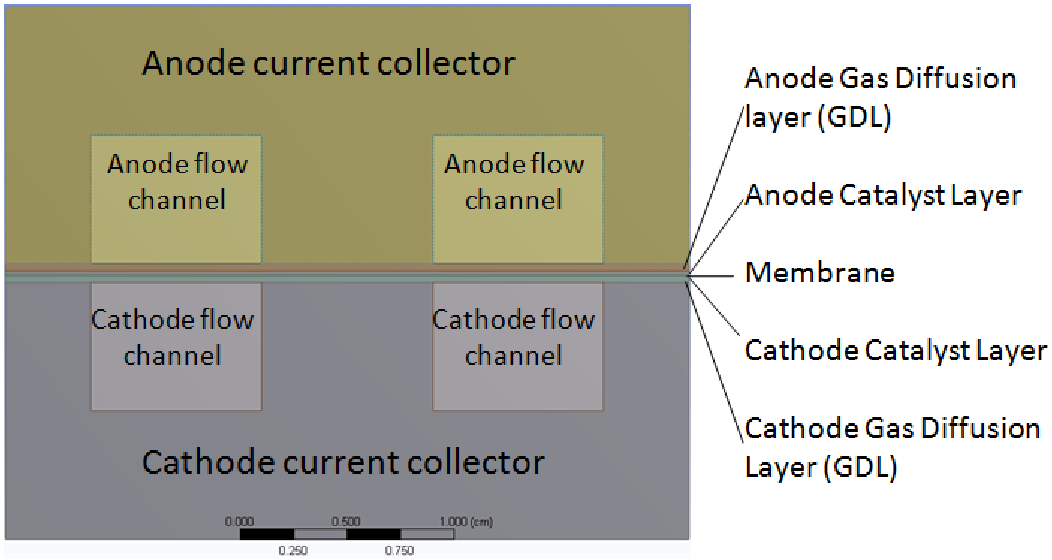


Figure 3. Front view cross section of a modeled fuel cell assembly. Anode side and cathode side parts shown.

The current collector has several roles in a PEM fuel cell and can be made of many different materials. It must be gas-impermeable, electronically conducting, mechanically sturdy, chemically stable, thermally conducting, low cost and easy to form. The gas flow channels are shaped by the current collector. Often times the channels will be voids carved into the collector surface. The current collector is the main avenue for the transport of electrons to and from the different reaction sites, so it must be highly electronically conducting or else the fuel cell system will lose a lot of energy to ohmic losses.

The current collector is generally what gives the cell mechanical stability. Hydrogen and oxygen are both reactive gasses so the current collector must be chemically stable so that it is not dissolved or corroded when it comes into contact with these chemicals. This is especially important because PEM fuel cells operate at relatively high temperatures in the presence of water, which increases corrosion rates. Heat is generated as part of the fuel cell reaction and this heat must be transported away from the membrane to prevent overheating of the membrane. Finally, the material must be low cost and easy to form if these types of cells are to become feasible in a commercial application.

Graphite is a common material for current collectors. It has good mechanical, electronic and thermal properties for this type of low-temperature fuel cell. The most important property of graphite is that it is highly chemically stable, so the hydrogen and oxygen will not destroy it. Unfortunately, graphite is very brittle, which means it can crack if mechanically shocked. In order for graphite to have enough mechanical strength to make commercial application feasible, it must be made in relatively thick amounts, which increases the overall size, weight and cost of a fuel cell system made with graphite plates.

Metals can be an attractive alternative to graphite for the collector plate. Many metals are ductile enough that they can be stamped into a particular flow field design. In addition, metal is highly thermally and electronically conductive and very mechanically sturdy. This means that very thin plates can be used while still maintaining mechanical stability, thereby saving on material costs, weight and size. Unfortunately, many metals will have destructive reactions when exposed to hydrogen and oxygen. Some metals can suffer from hydrogen embrittlement. Other metals may oxidize when exposed to oxygen at high temperatures, which can then affect their mechanical, electronic and thermal properties. This corrosion effect is one of the major drawbacks of using metal current collectors.

The flow field is another important part of the fuel cell. The flow field is generally a void which is cut into the current collector. However, the size, shape and length of the channels can have very important ramifications for the performance of the cell. There are a large number of flow field designs, each of which certain advantages and disadvantages.

The purpose of the flow field is to evenly distribute the reactant gasses to the reaction sites and to transport away the product materials without creating enormous pressure differentials, which would then require powerful external pumps. The reactant gasses are able to transport some heat so the flow field design can affect the thermal properties of the fuel cell. The most important role of the flow field in PEM fuel cells is water management.

The membrane of a PEM fuel cell must be moist in order to work, so some water is necessary for optimal performance. However, too much water will cause water to build up in the flow channels, blocking the flow of gasses, and reducing performance. This is a problem unique to low-temperature fuel cells because they are cool enough for water to condense into a liquid. An optimal flow field design will evenly deliver reactant gasses with minimal channel flooding and minimal pressure differentials between inlet and outlet.

One of the most straight-forward designs is the parallel design. This design features several possible paths from the inlet to the outlet. A typical parallel design is depicted in figure 4.

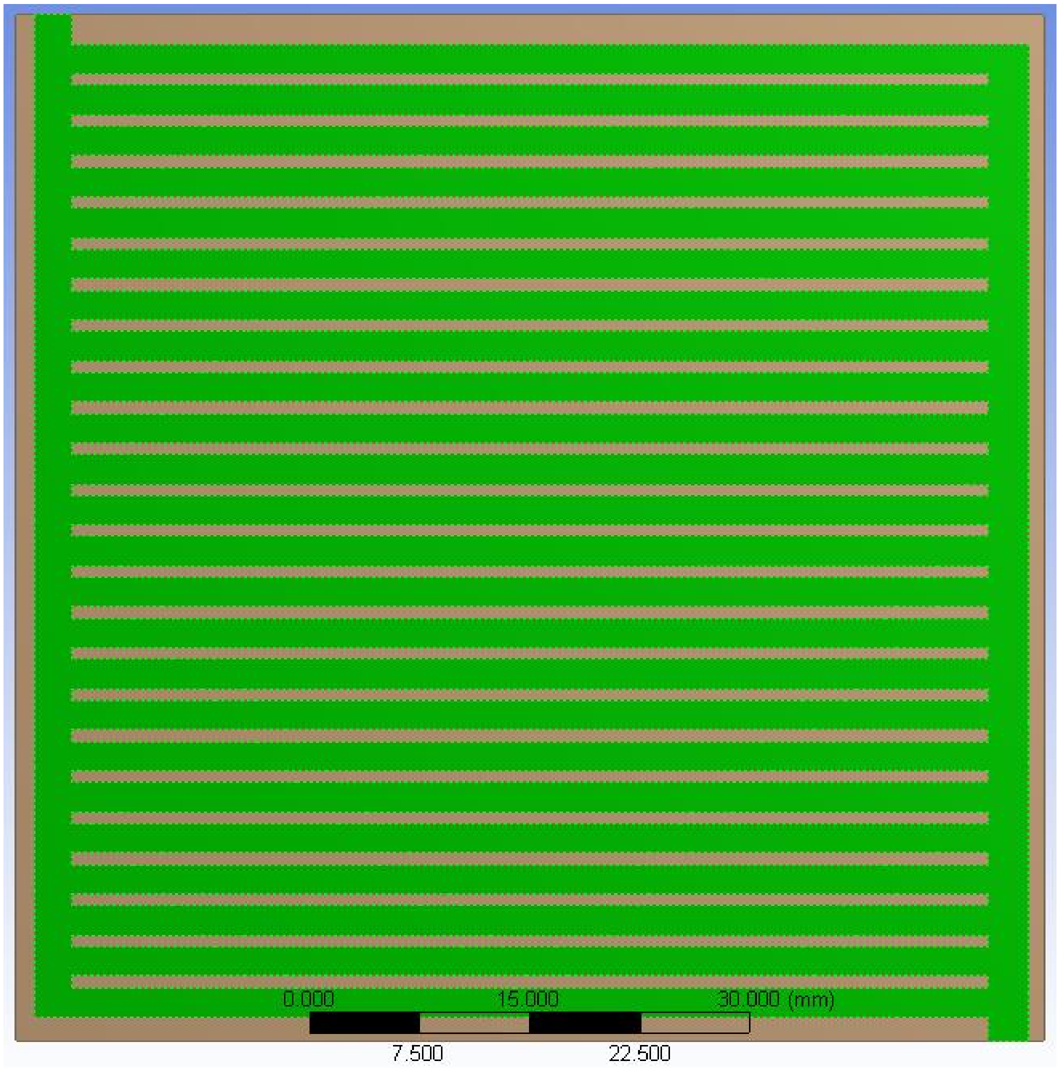


Figure 4. Parallel flow field geometry.

The key advantages to this design are its simplicity and low pressure difference from inlet to outlet. Any blockage in the gas channels can be easily bypassed. The low pressure drop corresponds to generally low gas velocities, which contribute to poor water management properties of this flow field design. Any water build up that occurs in the channels will not be effectively forced to the outlet. Water build up has two immediate negative consequences. First, liquid water will occupy reaction sites in the catalyst layer which will decrease the reaction rate. Second, water buildup in the channels can block the flow of the reactant gases which causes large inactive zones to form.

Another common flow field design is serpentine. This design features a single channel, which winds continuously in a single path from inlet to outlet. A very simple serpentine design is shown in figure 2, while another design with more turns is depicted in figure 5.

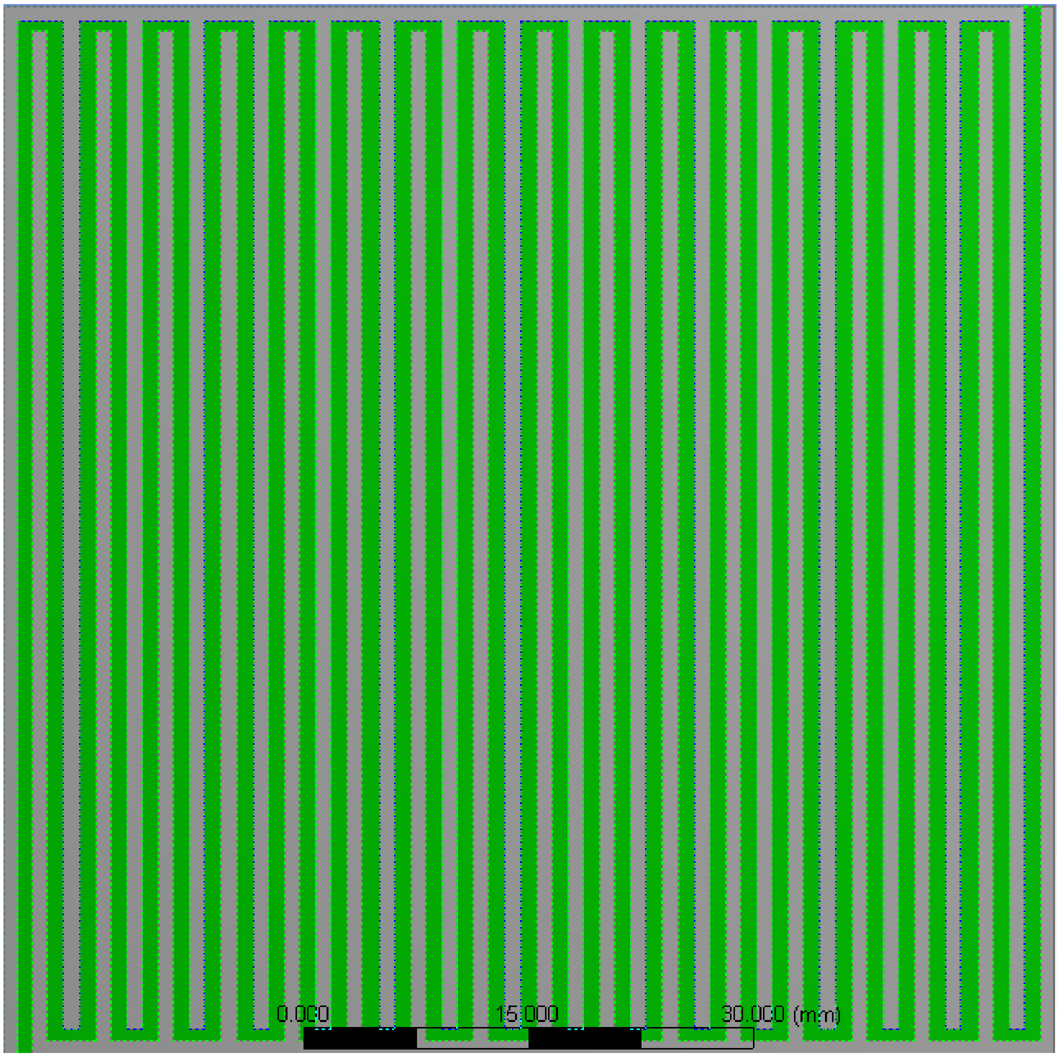


Figure 5. Single channel serpentine geometry showing one continuous channel winds from inlet to outlet.

One of the most important advantages of the serpentine geometry is that it addresses the issue of channel blockage by water formation by only having one channel. Any blockage that forms within the channel will eventually be forced to the outlet by the gas stream and the increasing of gas pressure behind any water buildup. The drawback of this single channel design is an increase in channel length from inlet to outlet, which increases the total frictional force between the gas flow and the walls.

The inlet gas must be pressurized to overcome the friction. External pressurization of the inlet gasses creates a parasitic load which reduces the efficiency of the fuel cell. This can be somewhat offset by the increase in reaction kinetics created by higher pressure reactants as described by the Nernst equation:

$$E_{cell} = E_o + \frac{RT}{nF} * \ln\left(\frac{P}{P_o}\right) \quad (3)$$

Increasing the pressure will also increase the cell voltage, and therefore the performance. However, for oxygen at 80℃ the fa ctor RT/nF is only 7.6 millivolts. Increasing the operating pressure accounts for a small increase in performance but it may not be enough to offset the parasitic power need of an external compressor.

The third common type of flow field is the interdigiated design. This geometry features flow channels that are not directly connected to each other. Instead, this design forces the gas to flow from an inlet channel through the gas diffusion layer (GDL) to reach an outlet channel.

Interdigiated designs have the major advantage of forcing the reactant gasses through the GDL. This helps to increase the utilization of the fuel since the gasses are forced through the GDL, which is where the reactions take place. One source has shown that interdigiated designs (Prasad, 2010) offer a middle ground between the performance and pressure drops of parallel and single channel serpentine designs.

There are many composite designs which can be created from these basic geometrical concepts as well as several completely different flow field designs. Changing the height, width or shape of the channels can change the performance of the fuel cell. Much work has been done on novel flow field designs such as porous meshes and radial designs, as well as work to find optimized flow field designs, which is discussed in the literature review.

Reactant gasses flow from the flow channel into the gas diffusion layer (GDL). The GDL must have many of the same properties as the current collector. The GDL should be electronically and thermally conductive, mechanically strong, and chemically stable. It should also be hydrophobic, which means it will repel water. However, unlike the current collector, the GDL must also be gas permeable and highly porous so that it can perform its most important function of distributing reactant gasses.

The main function of the GDL is to distribute the reactant gasses to the catalyst layer where they can react with one another. It also transports electrons and heat to the current collector. The GDL is usually 100 to 500 microns thick and must be designed to be hydrophobic so that it pushes water out of reaction sites and into the flow channel, where it can be removed. The GDL is adjacent to both the catalyst layers and the membrane, which are both extremely thin, and which both rely on the GDL to provide the mechanical stability for those parts. This is especially important if there is a pressure differential between anode and cathode.

A good GDL will be highly porous, hydrophobic, electronically conductive, thermally conductive, chemically stable, mechanically stable, low cost and easy to handle. Woven carbon cloths and carbon papers are common materials for

GDLs. One method to make GDLs is to mix a slurry of carbon black then roll it to the desired thickness. The rolled slurry is then dried to form a highly porous carbon cloth. Different carbon nanostructures, such as multi-walled carbon nanotubes, and chemicals such as polytetrafluoroethylene (PTFE), can be introduced into the process to change the physical and chemical properties of the GDL making it more effective at delivering the reactant gasses to the catalyst layer.

Catalysts facilitate chemical reactions but are not consumed in the reaction. Low-temperature fuel cells require a very active catalyst because hydrogen and oxygen are not very reactive at low temperatures. In PEM fuel cells the most common catalyst is platinum metal.

The catalyst layer in a PEM fuel cell must be electronically conductive, ionically conductive, gas permeable and durable. On the anode side of the fuel cell, the catalyst layer must oxidize H_2 molecules to create H^+ ions. The electrons from the oxidation must be transported to the anode side GDL, where they will eventually find their way to the cathode GDL through an external circuit. The H^+ ions must be transported to the membrane. On the cathode side, a similar reaction process takes place in the cathode catalyst layer. O_2 molecules are reduced by the catalyst to O^{2-} ions, which are then combined with H^+ ions coming through the membrane to form H_2O.

Platinum is a one of the most active catalysts, but it is also very expensive. Much work has been done to reduce the platinum loading by a variety of methods. Lu et al. (2007), Marie (2004), and Dhathathreyan et al. (2008) all focused on using nanostructures to reduce platinum loading. Platinum alloys were explored by Rambabu et al. (2010), Qian et al. (2004) and Grigorev et al. (2010) as well as many other groups. The use of substitute materials to completely replace platinum as a catalyst was investigated by Friedmann et al. (2009). Using different deposition techniques to increase the efficiency of the platinum was explored by Teki et al. (2009) and Saegeng et al. (2009) along with many others. These efforts help to reduce overall system costs.

There are a number of mechanisms which can destroy the platinum catalyst within a PEM fuel cell. Platinum can be poisoned by exposure to carbon monoxide which greatly reduces its catalytic activity. Over time the platinum particles can begin to agglomerate, which reduces the overall surface area and efficiency (Zhang, 2009). It is also possible for the platinum particles to migrate into the membrane and make it electronically conducting, which will ruin the fuel cell.

The central part of any PEM fuel cell is the membrane. It serves the same purpose as the electrolyte in a battery. The membrane in a PEM fuel cell should be proton conducting, electronically insulating, thermally and chemically stable and mechanically strong. Ideally, its properties would remain constant over a wide range of temperatures.

Almost all PEM fuel cell membranes are made of Nafion, which has become an industry standard (Larminie, 2003). This is one of the only materials that are able to perform all the necessary tasks of a membrane adequately. Nafion is produced for industrial production processes which makes Nafion easy to acquire and readily available for use in fuel cells.

The thickness of the membrane has a great effect on the performance of the cell and should be made as thin as possible in order to optimize performance. However, the membrane must also be durable, and therefore cannot be made too thin. Typical membrane thicknesses range from 50 microns to 200 microns. Temperature also has an effect on membrane durability. Nafion is stable up to approximately 100℃. Therefore, PEM fuel cells cannot exceed this temperature.

In order to remain ionically conductive, the membrane must be moist. Keeping the membrane hydrated is one of the major challenges during PEM fuel cell operation. It is important to remove water in order to prevent flooding of reaction sites and channels, but the membrane must also remain moist. Very careful management of water must be maintained to ensure optimal performance of the fuel cell.

The interaction of the various parts of a PEM fuel cell represent a challenge for modeling. A successful model can predict the performance of a fuel cell given the geometric parameters, material properties and operating conditions. Having a simulation tool available can help optimize the performance of a cell by simulating its performance under a wide range of operating conditions very quickly and without the need for several experimental setups. Different flow channel geometries can be tested and compared without the need for fabrication. Good models help observers understand the physical phenomena occurring within the cell. The increase in available computing power at lower cost makes sophisticated simulations complete in reasonable time frames.

Modeling and simulation do create some significant challenges. A successful theory must include computational fluid dynamics (CFD) to account for the flow of gasses and liquids within the flow channels, the GDL and the catalyst layers. Electrochemistry must also be included to account for the transformation of one species to another. The chemical reactions within the fuel generate ions and electrons, which then create and react to electric fields, so electric potential equations must also be included in any modeling simulation. These reactions and phase changes generate and consume heat and cause temperature changes, which can have significant effects on electrochemical and material properties. The standard physical conservation equations must also be obeyed, such as momentum, mass, charge, and energy. All of these competing and complimentary effects take place simultaneously during fuel cell operation.

CHAPTER 2

FUEL CELL MODELING PARAMETERS

Fuel cell modeling is a complicated task involving many aspects. An important aspect of fuel cell modeling is model validation. Variations in model predictions can be attributed not only to the accuracy of the equations employed, but also to issues such as mesh sensitivity, convergence errors as well as unknown operating conditions or material properties. Determining and implementing a large number of parameters is required to create a complete fuel cell model, and all of these parameters will inherently have a degree of uncertainty in their measurement or prediction. The predictions of any model should be verified against experimental measurements to ensure that the model is producing realistic and reliable. Every model should be verified before its results are trusted.

Siegel wrote a comprehensive review (Siegel, 2008) of published PEM fuel cell models, which included information about grid sizes, hardware used and computational times. Several one-dimensional, two-dimensional and three-dimensional models were reviewed. It was shown that one and two-dimensional models are sufficient for many types of studies. Various modeling strategies, their implementations, advantages and disadvantages were discussed. A comparison of available PEM modeling software was also provided which demonstrated that Ansys Fluent was used in a majority of the publications .

Effort is also being placed into using computational techniques to explore the use of membranes made of a material other than Nafion. One study from Denver et al. (2006) developed a three-dimensional model to understand the operation of a polybenzimidazole membrane fuel cell. This model was developed from previous work on a one-dimensional and two-dimensional model. This type of membrane has the advantage of being able to operate at higher temperatures, which increases the reaction rates and limits the effect of flooding. The model helped to predict areas within the membrane of high ohmic resistance and areas of oxygen depletion, as well as the effects of temperature and humidity. The model predictions were verified against experimental results from J.T. Wang et al. (1996).

One of the most common applications of three-dimensional fuel cell modeling is in the optimization of flow field designs. Chen et al. (2008) performed an analysis of wave-like gas flow fields using a fuel cell model developed and verified by L. Wang et al. (2003). The waves were set up like speed bumps within the flow channel creating a forced convection zone which was designed to direct the gas stream into the GDL. Their simulation compared the performance of a straight channel simulation to that of their wave-like channels and found that the waves in the channel did indeed improve performance.

Modeling of fuel cells also includes models that allow for the detection of faults within a simulation (Escobet, 2009). This model allows for an intentional defect to be introduced during the simulation, and the resultant change in the simulation behavior can be detected and attributed to faulty fuel cell operation rather than to a faulty simulation. This proposed fault detection model is independent of the fault magnitude and claims to be able to isolate the variable from which the fault originates. A model like this would be valuable in predicting the behavior of a fuel cell that experiences a sudden failure in one or several parts.

Models are often most valuable when they can focus on one single aspect of fuel cell performance and analyze it in extreme detail. This is the approach taken by Meng (2008) in the examination of cold-start conditions for PEM fuel cells. Running a fuel cell when at very cold temperatures, of less than 0℃, will drastically change the performance of the cell . Cold temperatures affect both the material properties of the cell as well as the chemical reaction rate. Close examination of heat flows through the cell are necessary to monitor ice formation and membrane ionic conductivity. The results of the study show a strong correlation between cathode gas humidity and ice formation and further show that most of the ice formation occurs in the cathode catalyst layer under the land channels. The study concludes that high gas flow rates in the cathode will greatly benefit the cold-start process.

An important aspect of fuel cell performance is the formation of water in the flow channels and GDL. Model development to understand water creation and flow within the channels has been investigated by many groups including Y. Wang et al. (2008), Steinkamp et al. (2008), Meng (2006), and Liu et al. (2005). These attempts include developing a model based on percolation theory (Dawes, 2009). Many models simplify the water formation by approximating the water as zero volume droplets that move at the same speed as the gas stream. This approximation can be fine for low current densities where water flooding is not as great of an issue. Many models, including the implementation used in Ansys software, begin to diverge from experimental results at high current density when flooding becomes severe.

Modeling of cooling channels is another important aspect of fuel cell modeling, especially when multiple cell stacks are simulated. Large fuel cell stacks can generate large amounts of excess heat which must be transferred out of the fuel cell by some mechanism in order to prevent overheating of the fuel cell. High temperatures are beneficial to chemical reaction rates so a balance must be struck between high performance and potential destruction of components, so cooling is a vital aspect of fuel cell performance. Ahn (2008) used a system level fuel cell model to design a control scheme to regulate the temperature of a water-cooled fuel cell stack.

Another work published by Min (2008) used computational fluid dynamics modeling to demonstrate the performance of a stepped flow field design. The stepped flow field is a parallel design that gradually decreases the height of the channels along the flow path creating a constriction. This model predicted an increase in performance over a conventional parallel design but it also created an increased pressure drop. To save computational time only a single channel was included in the computational domain.

Another numerical study (Wang, Xia, 2008) showed the effects of GDL deformation. This study used a GDL deformation model to determine the effect of compression from the channel lands on the physical dimensions and porosity of the GDL. This information was then used to generate a model which included these effects in the PEM fuel cell simulation. This study represents a step up in sophistication for computer modeling and the results show that the deformation model introduces significant variation in the performance compared to a base case with no GDL deformation, and concluded that the base case will over-predict actual fuel cell performance.

A study was performed that compared the performance of parallel and interdigitated flow field designs (Wang, Xiao-Dong, 2008). This study used numerical simulation to determine the effect of changing the channel aspect ratios. The channels were made tall and narrow, or wide and short, and the number and changes in overall size or cross sectional area of the channels were also compared. In the end, the study concluded that the best performance for parallel designs is 0.3 mm x 0.3 mm, and 1.0 mm x 1.0 mm, for interdigiated designs.

Another study focused on using numerical analysis to optimize the channels dimensions for a five-turn serpentine flow field (Huang, 2009). Channel heights and widths were adjusted along the flow path to determine the optimal configuration. A fairly non-uniform final design was generated. In order to simplify fabrication, the design was altered to give constant flow channel widths which only slightly decreased overall performance. It was found that this optimal configuration created a a peak power performance that was 22.5% greater than a constant channel dimension.

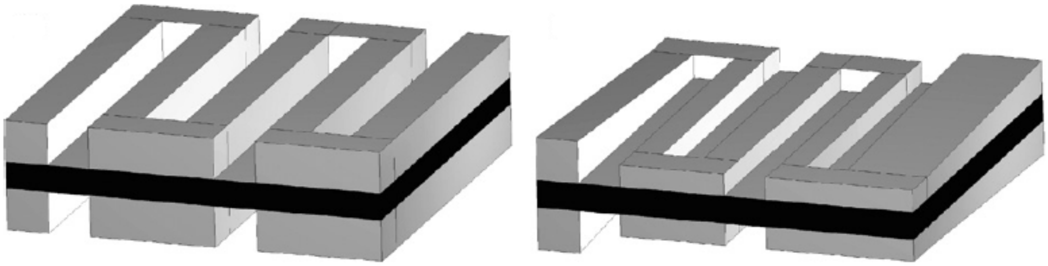


Figure 6. Standard serpentine design with constant flow channel dimensions next to an optimized serpentine design with changing dimensions (Huang, 2009).

This study is a very good example of the power of computer simulations for flow field optimization. In order to recreate a similar study experimentally, thousands of different flow channels would have to be fabricated and tested.

Another numerical study (Zhang, 2010) investigated the effect of the relative directions of the flow gasses in adjacent channels within the flow field and across the membrane electrode area (MEA). The solution domain was limited to just one and a half channels to simplify the calculation. The study produced some interesting results. The authors found that the in-plate adverse-flow did not improve overall cell performance but it did improve the current density distribution. Having a uniform current density is important in PEM fuel cells because it eliminates “hot spots” where large water production can cause flooding.

Another study (Wang, Xiao-Dong, 2011) attempted to model the transient characteristics of PEM fuel cells with different flow fields. This study compared responses to sudden changes in load of fuel cells with serpentine, parallel and interdigiated flow field designs. This approach was different from many other fuel cell simulations because it simulated the behavior of a fuel cell when operating conditions change with time. The experiment was intended to determine the time required for a fuel cell to reach steady state when the operating voltage is suddenly changed from 0.5 V to 0.7 V and back. The results of the study show that there is some variation between the different designs, but that they all reach steady state between 150 and 250 milliseconds.

A radial flow field design was evaluated (Hernandez-Guerrero, 2010) using computer modeling. Radial designs are setup as coaxial pipes where gas enters through the inner radius and exit through the outer radius. The cross sectional area of the flow channel naturally diverges in a radial design. The authors present model simulations to demonstrate improved performance of this design. The model was verified against other published results and then the radial geometry was evaluated. Results show that pressure drops are low and current density peaks near the center of the disc, but the results are not presented in a form that can be easily compared to the performance of other fuel cell designs.

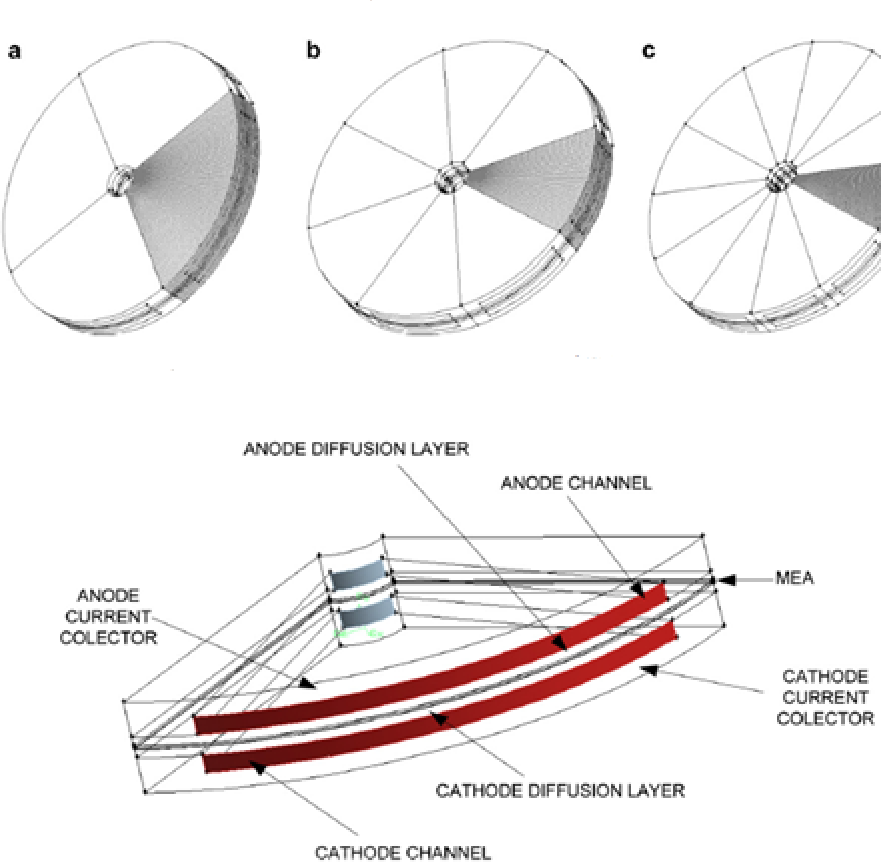


Figure 7. A radial flow field design with varying numbers of channels (a) 4, (b) 8, (c) 12. (Hernandez-Guerrero, 2010).

Other novel geometries can also be investigated using CFD modeling. A three-dimensional model of a tubular-shaped PEM fuel cell was developed by Sadiq Al-Baghadadi (2008). The overall performance of this design is comparable to traditional flat designs. The advantages or drawbacks of this configuration are not discussed in any detail. The results were compared to an experimental setup by L. Wang (2003) that used a flat configuration, but with a similar membrane electrode assembly surface area and which produced results that correlated well with experimental results.

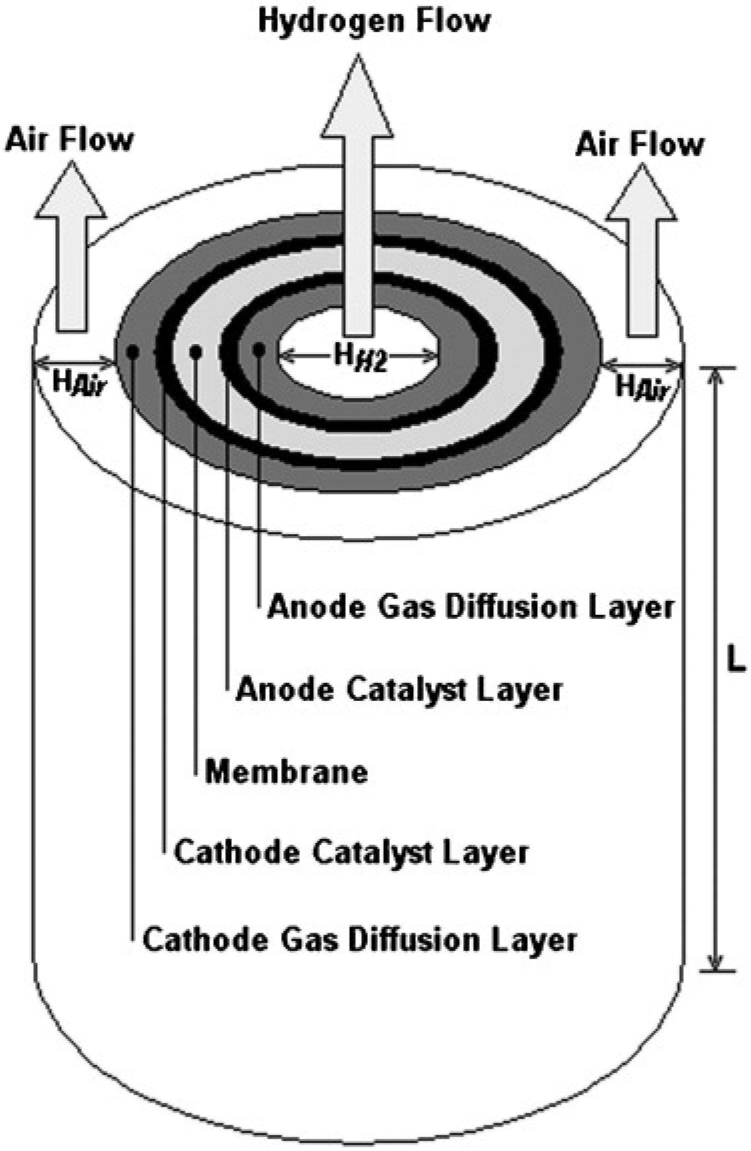


Figure 8. Tubular fuel cell design (Sadiq Al-Baghdadi, 2008).

CFD modeling of fuel cells can also focus on specific parts of the fuel cell. One study focuses on fuel cell subsystems (Zhukovsky, 2011). They use a computer simulation of nanostructure interactions to generate nanomaterials that are ideally suited for use as catalyst material and GDL material. The work proposes a simulation which could predict the form of a nanomaterial that will be highly reactive with hydrogen atoms, which could serve as a direct and more affordable replacement for costly platinum as a catalyst.

Many experimental studies can be used to verify simulations. A comprehensive transient study of water build up in the channels of various flow field designs is presented in a study by Prasad (2010). This study used a neutron camera to image water formation within the cell. Transparent plates were also created to allow for optical corroboration with neutron camera measurements. Water management within fuel cells is an important issue that is directly affected by flow field design. The results of this study can be used to verify the validity of a fuel cell model that is intended to predict the effects of flooding.

Another study (Iranzo, 2010) aimed to experimentally verify the Ansys Fluent fuel cell model using a 50 cm^2 fuel cell. Model verification included a serpentine and parallel flow field design. All of the material properties and operating conditions from the experimental setup were entered into the model. The single unknown parameter was adjusted to match experimental results. The results of the study showed that the model could make good predictions about fuel cell behavior. There was an important limitation in the study. Hardware limitations prevented a complete mesh sensitivity analysis to be performed. It was found that a 1.8 million element mesh was within 5% of a 1.0 million element mesh. The 1.8 million element mesh should be verified against a larger mesh to ensure that a larger mesh does not affect the solution.

Ansys Fluent was chosen as the fuel cell modeling software because it is currently the most complete three-dimensional modeling software commercially available. Their implementation of a fuel cell model includes a straight forward graphical user interface which simplifies the modeling and simulation effort. The software is also capable of being run in parallel so that larger and more computationally demanding simulations can be completed in reasonable time frames.

The simulation setup involves three major steps. The first step is modeling the geometry of the fuel cell using computer-aided design software. The geometrical model forms the basis for creating a computational mesh.

Fuel cell behavior is governed by a set of coupled, non-linear differential equations. These equations must be solved using a finite volume numerical approximation technique. The approximation requires the geometry to be broken up into volume elements or computational cells. These individual cells compose the computational mesh. The relevant equations can then solved in each individual cell and then the results are summed over the computational domain to give a total solution.

The computational mesh has a strong influence on the accuracy of the solution so it should be generated carefully. It requires the careful balance of creating enough computational cells to capture the geometry without creating so many that it exceeds the available memory of the meshing computer. Many other factors must also be taken into account in order to generate a computational mesh provides representative results when simulated.

The third and final step involves inputting the various physical and operating parameters of the simulation. Some of these include thermal and electrical properties of the various materials, operating temperatures and pressures, inlet gas flow rates, open circuit voltage, porosity, and humidification among many others.

CHAPTER 3

MEA MODELING AND SIMULATION

The process for modeling and simulating fuel cell performance involves three basic steps. The first step is to create the geometry using computer aided design (CAD) software. There are a number of software packages available which are capable of generating a fuel cell geometry. Once the geometry is created it must be decomposed into a computational mesh and have its boundary conditions defined. The next step in the process is to define the parameters for the solver. These include the type of mathematical models which will be applied, the criteria for convergence, solution methods and controls, and parallel processing parameters. Once the solution is calculated it should be verified against experimental results. It is possible that a numerical solution to a given set of equations can produce more than one solution which satisfies the boundary conditions. It must be verified that the solution found correlates with real world fuel cell behavior.

Creating the geometry can be done in a number of ways using many different software packages. The purpose of the geometry is to capture the physical dimensions of the parts and their relation to one another. Creating a well defined geometry is vital step in the process. The boundaries which define the simulation are defined by the geometry. A high quality geometry which accurately reproduces all of the dimensions of the fuel cell is then used as the basis for a computational mesh.

Computer modeling of fuel cells requires that the geometry be broken down into smaller volume elements. A three dimensional computational mesh is composed of both surface and volume elements. Surface elements are generally composed of triangles or quadrilaterals while volume elements can be composed of tetrahedrons (pyramids) or hexahedrons (cubes). The type of element used depends on the type of simulation being run.

The advantage of tetrahedrons and triangles is that they are much more flexible and easy to fit to a given geometry. These types of meshes can be created using relatively simple meshing algorithms and can generate a mesh regardless of the geometrical complexity. However, tetrahedrons can require many more elements to represent volumes compared to hexahedral cells. Tetrahedral cells are generally used only for surfaces or very thin geometries which do not contain very much volume relative to surface area. These types of meshes are commonly used in analyzing the surfaces of aerodynamic bodies.

Quadrilateral faces with hexahedral volume elements can be more complex to fit to a given geometry without creating highly skewed elements. Hexahedral cells are preferred for meshing volume elements since they can generate a more uniform mesh with fewer elements compared to tetrahedral cells. Many fuel parts can be modeled using mathematically regular solids such as cubes which can be decomposed into uniform hexahedral elements. Uniform cube sizes can be controlled using a small number of parameters which makes hexahedral meshes easier to generate for many fuel cell designs.

The number of cells in the mesh depends on the details of the geometry and the type of simulation parameters being used. A very fine mesh will have small individual elements and a large total number of elements which may require long computational times. Using a coarse mesh can reduce the computational times but may produce inaccurate simulation results. The balance between mesh detail and computational time must be carefully considered during the meshing process.

All of the geometries for this project were created using SolidWorks2010. SolidWorks 2010 is a popular computer-aided design software published by Dassault Systèmes SolidWorks Corp. SolidWorks is often available at university campuses as part of the mechanical engineering program or other related or similar departments. SolidWorks was chosen for this project because of its availability on university campuses, as well as its ease of use.

Three different geometries were created during this project. The first was a simple single channel fuel cell design based on the fuel cell tutorial provided by Ansys. The purpose of this design was to provide an introduction to the fuel cell components and how they are arranged. The various components were easy to recognize and generate in the modeler. The geometry provided a straightforward base for creating a simple, high quality mesh.

The dimensions of the components in the single channel geometry were very unrepresentative of actual fuel cell components. The measurements of the components within the MEA were exaggerated to show the distinction between parts. The GDL measured 2100 microns thick, the membrane was 360 microns thick and the catalyst layers were 120 microns thick. Overall dimensions of the fuel cell were 2.4 cm wide, 2.8 cm tall and 12.5 cm long.

The mesh was generated using Ansys Workbench Meshing. Named selections are required to define the boundary conditions of the mesh and naming conventions used are specified in Table 1. The mesh was generated by using all of the default mesh controls except that the maximum face size was specified to be 0.8 mm.

Table 1

Naming conventions for boundary surfaces

Surface Function	Named surface
Anode flow channel inlet (hydrogen)	massflow_inlet_a
Cathode flow channel inlet (air/oxygen)	massflow_inlet_c
Anode flow channel outlet	pressure_outlet_a
Cathode flow channel outlet	pressure_outlet_c
Anode side electrical contact	terminal_a
cathode side electrical contact	terminal_c

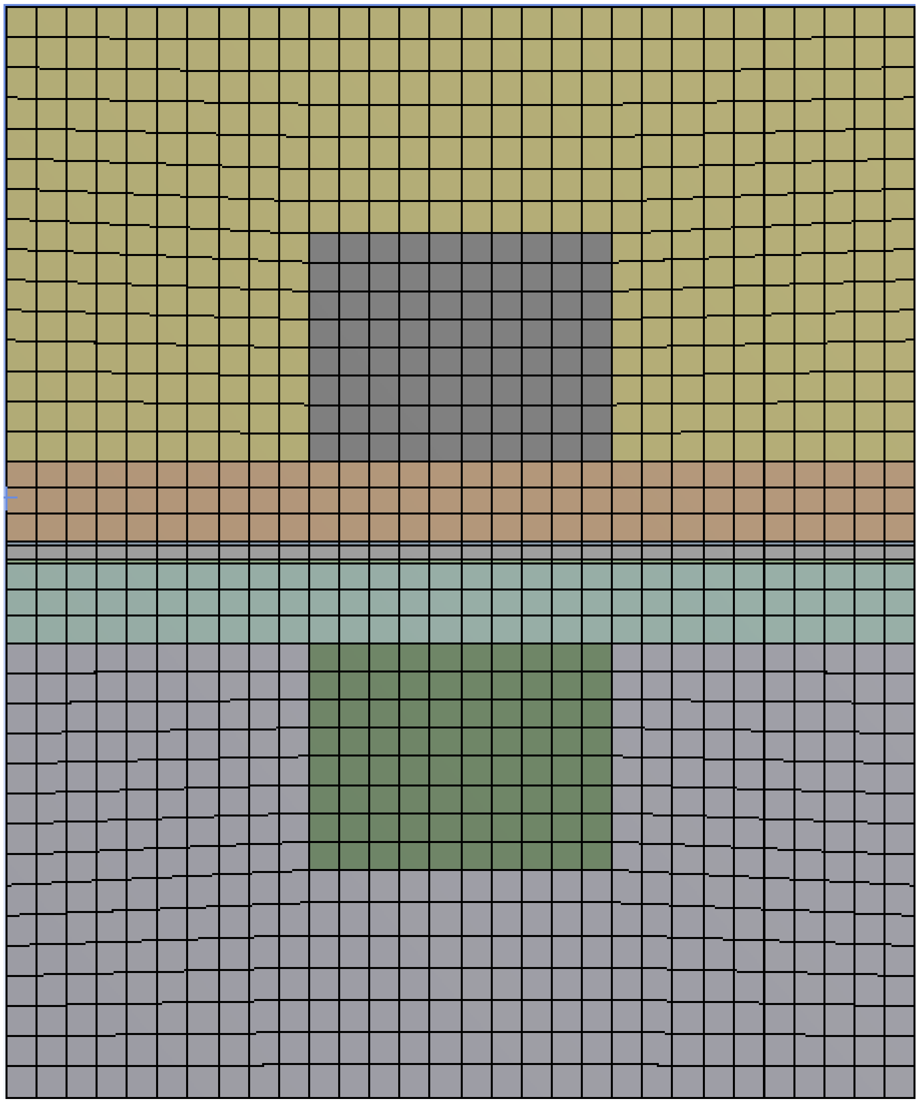


Figure 9. Single channel mesh front view

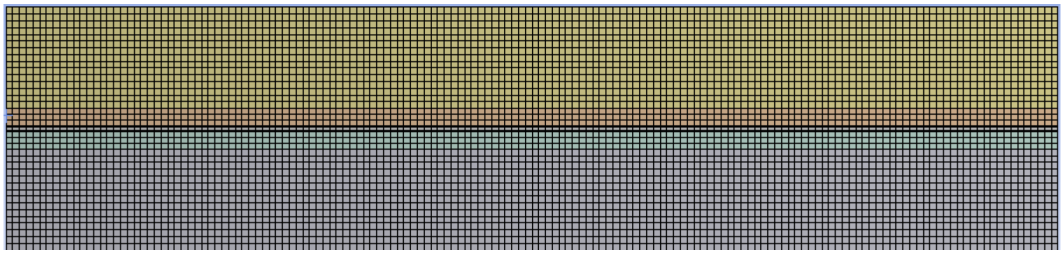


Figure 10. Single channel mesh side view

The geometry used was uniquely suited for meshing using Ansys Workbench. The hexahedral mesh implementation used by Ansys Workbench requires that each individual component have at least 1 pair of symmetrical walls with a constant cross section the whole way through. This arrangement allows the component to be “swept” from one wall to the other without interruption. That means that the mesh pattern can be defined at an originating surface and then replicated as many times as necessary to fill the space between the two walls. This process would be similar to constructing a building from the ground up where each floor is exactly the same. The base floor is designed then repeated upwards floor by floor until the building is full. This process requires that the building have a constant cross section from bottom to top. This method is highly efficient and fast but can only work for geometries which have this type of symmetry. If this method is applied to geometries that have non-sweepable bodies then the hexahedral meshing technique will fail or produce a mesh with uniform elements which may have convergence issues when simulated.

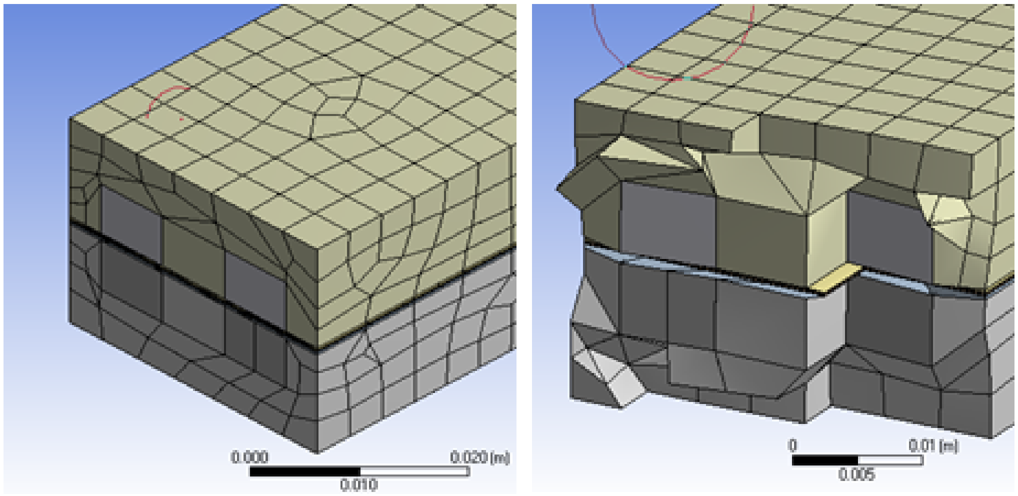


Figure 11. Attempting to use the hex-dominant method built into Ansys Workbench Meshing on a non-sweepable body will result in non-uniform mesh elements

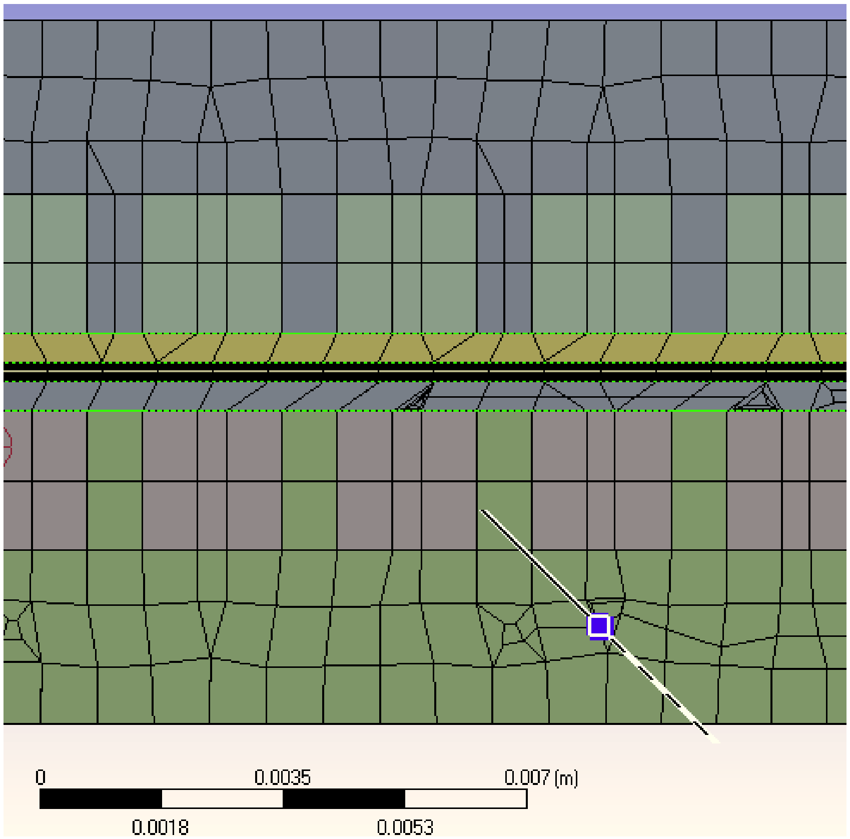


Figure 12. Attempting to use the hex-dominant method on a non-sweepable geometry produces mesh singularities which cause non-convergence

A more sophisticated parallel geometry was created using SolidWorks based on the geometrical parameters described in Iranzo et al. (2010). These parameters were based on experimental measurements of the various geometrical parameters of each fuel cell part. This design was chosen because the overall flow channel design is highly symmetric which simplified the CAD modeling process. Experimental performance measurements of a fuel cell using this geometry were also published by Iranzo et al. (2010) which can be used for model verification.

The overall dimensions of the parallel geometry are 70.0 mm wide,70.0 mm long, and 20.033 mm tall. The overall plate thickness was 9.5 mm.

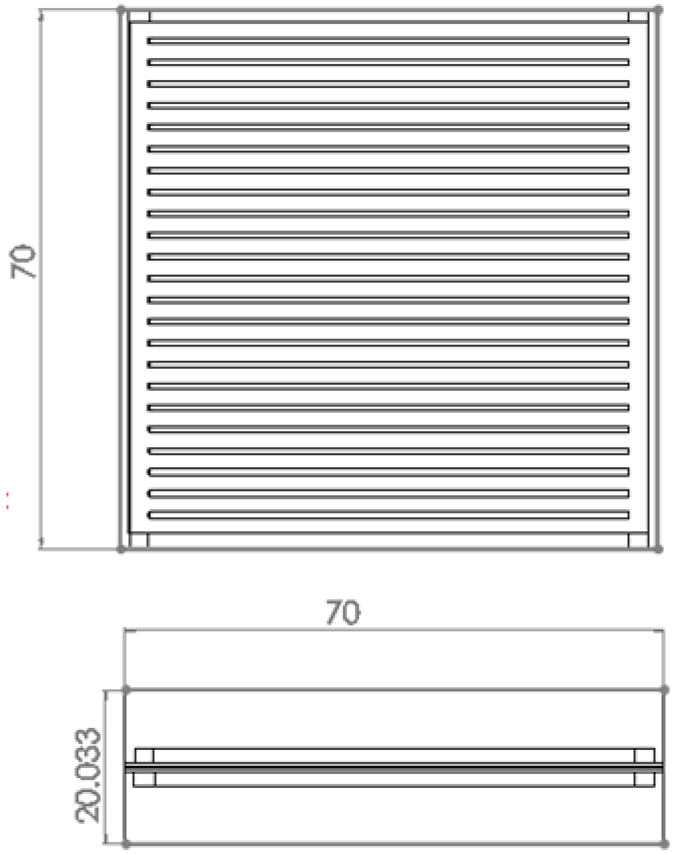


Figure 13. Parallel geometry overall dimensions. Displayed units in mm

Channel height was 2.0 mm and channel thickness was 2.0 mm. Rib thickness was 0.8 mm. The inlet channel was 2.5 mm wide and 2.0mm tall. GDL thickness was set at 420 microns and the membrane was set at 175 microns thick. The anode and catalyst layers had different thicknesses with the anode layer being 6 microns thick and the catalyst layer being 12 microns thick.

Initially the mesh was generated using Ansys Workbench Meshing however it was discovered that the mesh produced was unacceptable. The meshing process produced a mesh with such poor quality that the simulation failed to converge. In addition, the controls available in the workbench meshing program were inadequate to control the element size in each individual part. A different software program was employed to generate the mesh for this geometry.

Ansys ICEM CFD was used to create the mesh for the parallel geometry. This program offers many more options for generating meshes. The Cartesian grid method was specified which allows for the number of divisions in a given coordinate direction to be specified for each individual part. These fuel cell geometries are composed entirely of rectangles with faces at right angles and no curves which made the Cartesian grid method ideal. The number of cells along the length and width of the fuel cell was set at 89 which correspond to 0.795 mm wide elements. This number of cells was chosen because it most closely matched the minimum dimension of 0.8 mm wide for the channel ribs. Using larger element sizes would result in the rib and channel geometries merging into a single mesh element.

Different numbers of cells were specified vertically for each part in the fuel cell mesh. Initially 5 cells were chosen for the current collector zones, the flow channel zones and the GDL zones while 1 cell was specified. These parameters were chosen in order to set a baseline in order to determine the number of cells in each direction and in each zone which produce results which are independent of the mesh elements. A two dimensional grid sensitivity analysis (Kamarajugadda, 2008) suggests that a minimum of 40 computational cells within the membrane zone and 5 within the catalyst layer is required to produce grid independent results.

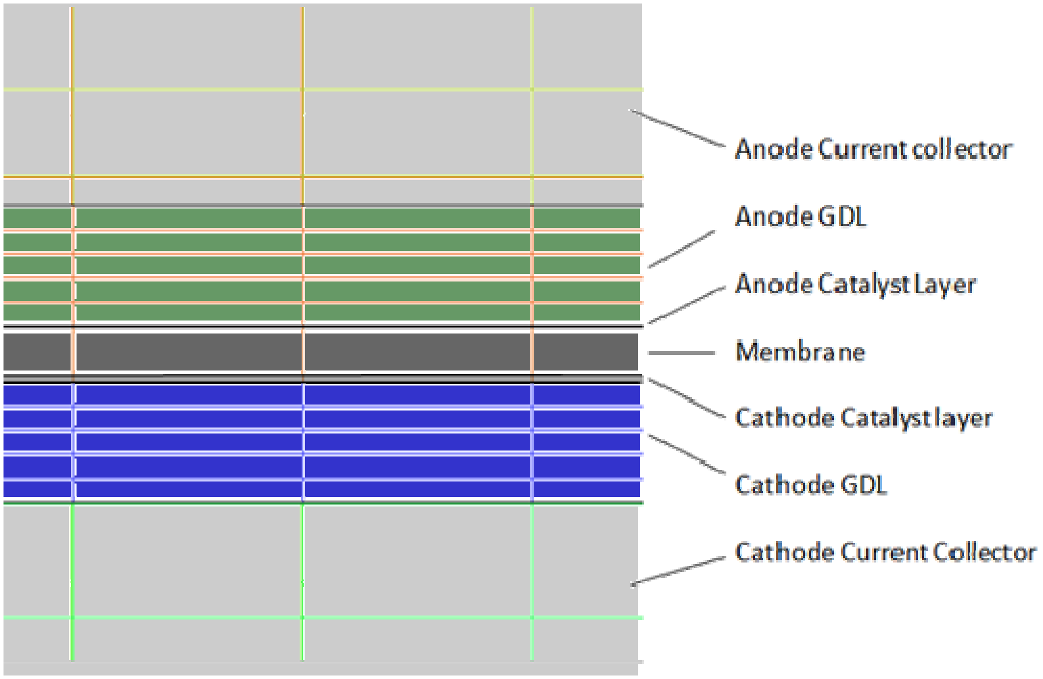


Figure 14. Detailed look at MEA zones with 5 divisions in the GDL layers and a single division in the membrane and catalyst layers.

The number of vertical cells within the membrane and catalyst layers was varied in order to determine the effect on the solution and to verify that the results from the previous two dimensional analysis are still applicable to a three dimensional implementation.

A final geometry was developed using a single channel serpentine design. This geometry was chosen because single turn serpentine designs are simple to design and are commonly used in fuel cells. Dimensions of the MEA and collector plates were retained from the previous design. Channel height was set at 0.8 mm and channel width was set at 1.1 mm, rib width was set to 1.0 mm. Overall the path makes 32 turns.

A mesh was generated using the Cartesian grid method. 175 cells were used across the path direction and 112 along the flow path direction. In the vertical direction, 12 cells within the plate, 4 cells in the flow channel zone and 5 cells in the GDL zone. This mesh is used to determine the required number of iterations to achieve convergence as well as a mesh sensitivity analysis. These values represent the minimum number of cells required in each direction to capture the geometry and produce highly uniform elements. The number of vertical cells within the membrane and catalyst layers is set at 1 for the convergence analysis.

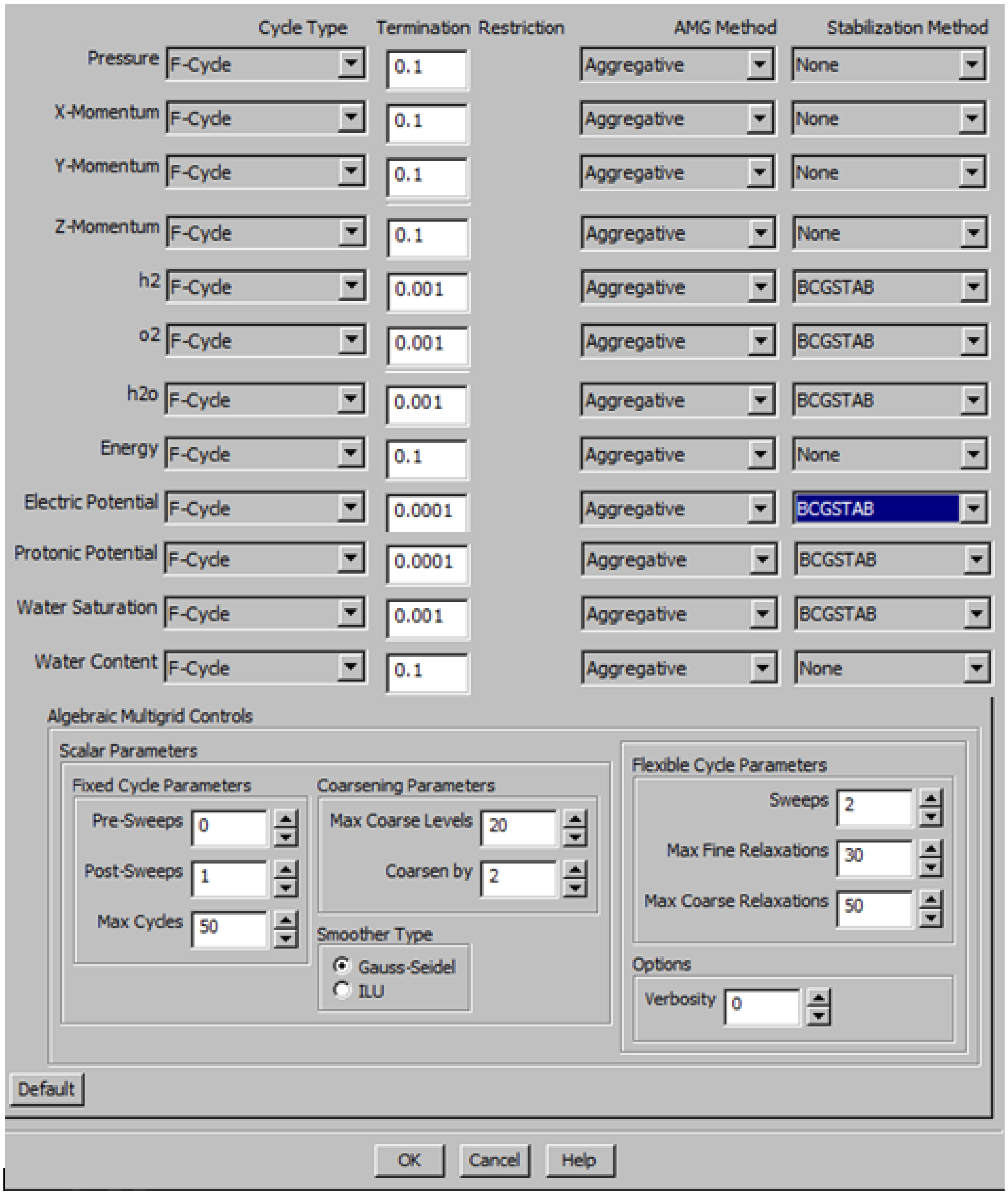
A number of boundary conditions and material properties must also be set for the simulation. Some of these include gas flow rates, operating pressures and temperatures, heat flux rates, resistances and load. These parameters will vary depending on the requirements of the simulation and the materials used. The types of models which can be included in the simulation include joule hating, reaction heating, electrochemistry sources, Butler-Volmer rate, membrane water transport, multiphase, multi-component diffusion and anisotropic e-conductivity in porous electrode.

The joule heating option takes into account ohmic heating due to resistance in the various components of the fuel cell. Reaction heating accounts for heat generated by the chemical reactions within the cell. Electrochemistry sources accounts for the electrochemistry which occurs in a fuel cell. The Butler-Volmer Rate is used to compute transfer currents within the catalyst layers. The membrane water transport option simulates the transport of water within the membrane. Multiphase effects are important for accounting for liquid water formation in the GDL. Multi-component diffusion allows for simulating the effects of having different diffusion rates in different directions within the GDL. Anisotropic e-conductivity in porous electrode is used to model the effects of having different conductivity values in different directions within the GDL. Different models can be enabled or disabled depending on the type of simulation being run.

After the computational mesh is created it must be imported into the solver. The solver applies a set of simultaneous coupled non-linear equations to each cell and determines the solution necessary to satisfy the boundary conditions. The methods to determine the solution depend on how the parameters are set. Ansys recommends changing the multigrid cycle to F-Cycle with BCGSTAB (bi-conjugate gradient stabilized method) for the species, potential and saturation equations in order to help ensure convergence. In addition, the termination restriction for the 3 species equation and the saturation equation should be set to 0.001 while the termination restriction for the two potential equations to 0.0001. An image with the appropriate controls is shown in figure 15. These solution controls were used for all simulations.

Under relaxation factors are also used to control the solution. Numerical solutions in general can be thought of as very sophisticated guess and check algorithms. In these fuel simulatoins there are typically 12 under relaxation factors corresponding to different groups of independent variables. The methodology for determining which value to guess next depends on the solution scheme and the under relaxation factors control how large the next guess will be compared to the previous guess. If the under relaxation factor is set too low it can cause the solution to freeze before converging. If the under relaxation factor is set too high it can cause the solutions to oscillate around a particular value and can cause divergence. In most of these simulations it was found that an under-relaxation factor of 0.3 for momentum, 0.7 for pressure and 0.95 for H_2, O_2, H_2O

and Water Saturation lead to convergence. With out setting these under relaxation factors the solution tends to oscillate rather than converge.



	Cycle Type	Termination	Restriction	AMG Method	Stabilization Method
Pressure	F-Cycle	0.1		Aggregative	None
X-Momentum	F-Cycle	0.1		Aggregative	None
Y-Momentum	F-Cycle	0.1		Aggregative	None
Z-Momentum	F-Cycle	0.1		Aggregative	None
h2	F-Cycle	0.001		Aggregative	BCGSTAB
o2	F-Cycle	0.001		Aggregative	BCGSTAB
h2o	F-Cycle	0.001		Aggregative	BCGSTAB
Energy	F-Cycle	0.1		Aggregative	None
Electric Potential	F-Cycle	0.0001		Aggregative	BCGSTAB
Protonic Potential	F-Cycle	0.0001		Aggregative	BCGSTAB
Water Saturation	F-Cycle	0.001		Aggregative	BCGSTAB
Water Content	F-Cycle	0.1		Aggregative	None

Algebraic Multigrid Controls

Scalar Parameters

Fixed Cycle Parameters

Pre-Sweeps 0

Post-Sweeps 1

Max Cycles 50

Coarsening Parameters

Max Coarse Levels 20

Coarsen by 2

Smoother Type

● Gauss-Seidel

○ ILU

Flexible Cycle Parameters

Sweeps 2

Max Fine Relaxations 30

Max Coarse Relaxations 50

Options

Verbosity 0

Figure 15. Suggested solution controls

Monitoring of the residuals is necessary to verify that under relaxation factors are set appropriately. The solver will automatically graph the solution as it progresses and the graph should show a constant decline in the values of the residuals. If a residual is increasing then than that suggests that the mesh has a singularity or that there is a non-physical parameter in the simulation setup. Residuals that oscillate with time will need their associated under-relaxation factor decreased.

Inspection of the residuals is an important part of determining convergence. The criteria for deciding when a solution has converged varies depending on the accuracy of the results desired. Conversations with researchers familiar with the software have suggested a three-part convergence check, which requires calculating the current flux through the terminals, measuring the reactant species consumption and checking the reported current density. A fourth step is also necessary in order to check that the values of the solution are not changing with further iterations.

Calculating the current flux through the terminals is a straightforward process which requires typing the following command into the TUI console.

```
/report/surface-integrals/integral terminal_c , y-current-flux-density n
```

Which generates the following output.

```
                         Integral
          Y Current Flux Density           (a/m2)(m2)
-------------------------------- --------------------
                       terminal_c           -8.3995056
```

This value can be divided by the membrane area to determine the current density which can then be compared to the value reported by the residuals. Next, the mass species report is generated by using the TUI command

/report/species-mass-flow

This command generates a list of the mass flux of each of the three species in the simulation through the various boundaries.

```
zone 47 (catalyst_c): (0 0 0)
zone 54 (channel_a): (0 0 0)
zone 55 (channel_c): (0 0 0)
zone 23 (current_c-shadow): (0 0 0)
zone 52 (gdl_a): (0 0 0)
zone 5 (gdl_a:005): (0 0 0)
zone 53 (gdl_c): (0 0 0)
zone 15 (gdl_c:007-shadow): (0 0 0)
zone 61 (massflow_inlet_a): (2.5e-06 0 2.5e-06)
zone 58 (massflow_inlet_c): (0 1.5708e-05 1.32e-05)
zone 51 (membrane): (0 0 0)
zone 59 (outlet_a): (-2.39904e-06 -2.1026266e-09 -
1.1919924e-06)
zone 60 (outlet_c): (-6.2110525e-09 -1.4959543e-05 -
4.0301031e-06)

net species-mass-flow: (9.4748948e-08 7.4635437e-07
1.0477904e-05)
```

The three numbers following each zone represent the flux of hydrogen, oxygen and water in kilograms/second.   The net species mass flow is the total amount of each species that enters and exits the simulation. This parameter can be used to calculate the total current generated using the following equation and constants.

$$I = \frac{m\,n\,F}{MM} \qquad (4)$$

Table 2

Mathematical symbols

I	Current (amperes)
m	Mass flux (kg/sec)
n	Number of electrons transferred
F	Faradays constant 96485000 C/kmol
MM	Molar mass (kg/kmol)

Using the mass flux of oxygen and the number electrons transferred per oxygen molecule which is 4 and a molar mass of 32, the current can be calculated

$$I = \frac{7.464 \times 10^{-7} \times 4 \times 96485000}{32} = 9.00\ A$$

This value should be compared to the absolute value of the earlier reported current density of 8.40. There is a variation of 0.6 A which suggests the solution in this case has not converged. The amount of error between the two values that is acceptable depends on the accuracy required. It is possible to achieve values which vary by less than 1%. The calculation can be repeated for the hydrogen species consumption using the electrons transferred per mole value of 2 and the molar mass of 2 kg/kmol.

$$I = \frac{9.475 \times 10^{-8} \times 2 \times 96485000}{2} = 9.14\ A$$

This value also varies from the measured current flux through the cathode by over 0.7 A. Convergence criteria should be defined which includes verifying that species mass flux and current flux agrees within a specified limit.

In addition to species conservation, a check of solution stability must be performed. This can be accomplished by setting criteria for solution convergence with respect to further iterations. The number of iterations required depends on the type of simulation being performed. Without an iteration sensitivity analysis there is no way to know whether or not a solution has converged. A detailed iteration study is performed on the single channel serpentine geometry to set a baseline expectation for when a solution can be considered convergent.

Performing an iteration sensitivity analysis can require many thousands of iterations. A highly detailed mesh may require hundreds of hours of computational time to produce a solution. The simulation software is capable of being run in parallel mode which can reduce the time required for simulations. The single channel and parallel geometries were run using a single processor so no parallel setup was required. The serpentine design was partitioned using Cartesian axes method to generate 4 partitions to be run simultaneously. Currently 4 parallel processes is the maximum allowed by the software licensing available.

Developing a fuel cell simulation is a complex multi step process. Creating a geometry which captures the physical parameters of the fuel cell is the most basic part of the fuel cell simulation process. A geometry which is unrepresentative of a real fuel cell will produce inaccurate results. The geometry serves as the base from which a computational mesh is generated. The quality of the mesh is fundamental to the quality of the solution. The mesh must be detailed enough to capture the geometry. Too many elements in the mesh will increase the computational times greatly. A balance must be struck between mesh sensitivity and computational times. This can be offset somewhat by increasing the hardware available and taking advantage of parallel processing, but this has associated costs and limitations.

CHAPTER 4

RESULTS AND DISCUSSION

The first simulation used a geometry based on the fuel cell tutorial and meshing with Ansys Workbench. The final mesh contained approximately 93,000 elements. Anode voltage was set to 0 and potentiostatic boundary conditions on the cathode were solved for. The simulation was run for 200 iterations. Cathode set voltage and the calculated y-current-flux-density integral on the cathode terminal were recorded. Anode inlet flow rate was set to 6x10^{-7} kg/s with a mass fraction of 0.8 hydrogen and 0.2 water. Cathode inlet flow rate was set at 5x10^{-6} kg/s with 0.2 oxygen and 0.1 water mass fraction for the air data set. For oxygen the mass fraction was set at 0.9 oxygen 0.1 air. Open circuit voltage was set to 0.98V and all other model and material parameters were left at their default values. Temperature was set constant at 353K. Models included for this simulation included joule heating, reaction heating, electrochemistry sources, Butler-Volmer rate, and membrane water transport.

The current-voltage graphs that are generated have the independent variable, voltage, on the vertical axis while the dependent variable, current, is on the horizontal axis. This is an inversion of the typical orientation of independent and dependent variables but many experimental current-voltage curves generated for the purpose of displaying fuel cell performance are orientated with the current on the horizontal axis. For this reason the graphs are displayed with inverted axes so that the results may be more easily compared to experimental measurements.

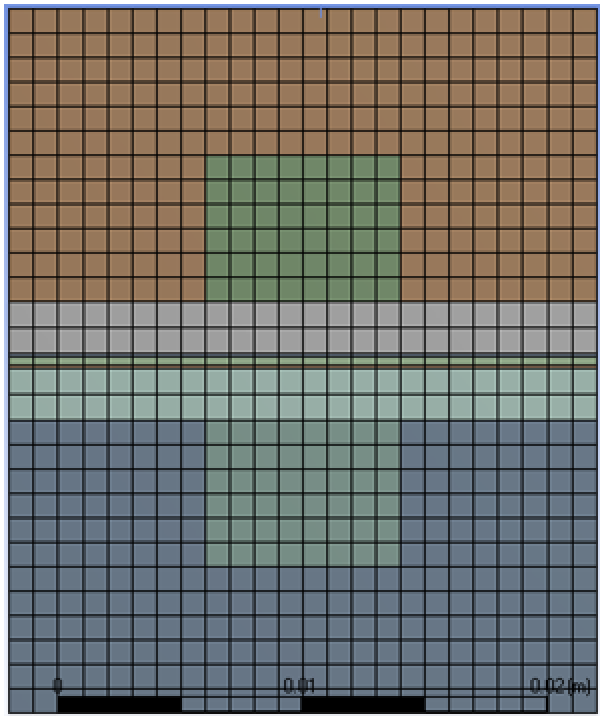


Figure 16. Mesh cross section. Each element width approximately 0.1 mm.

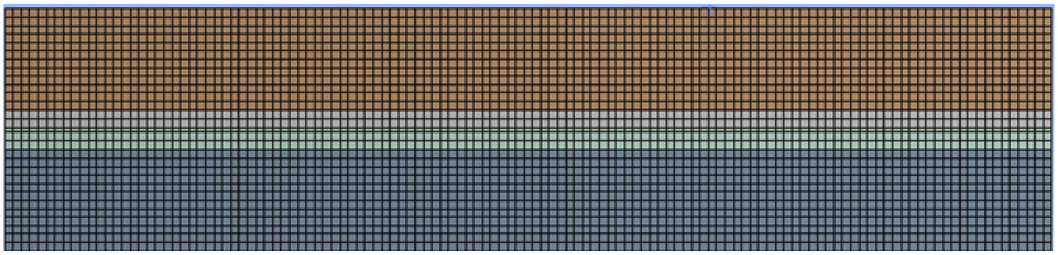


Figure 17. Single channel mesh side view.

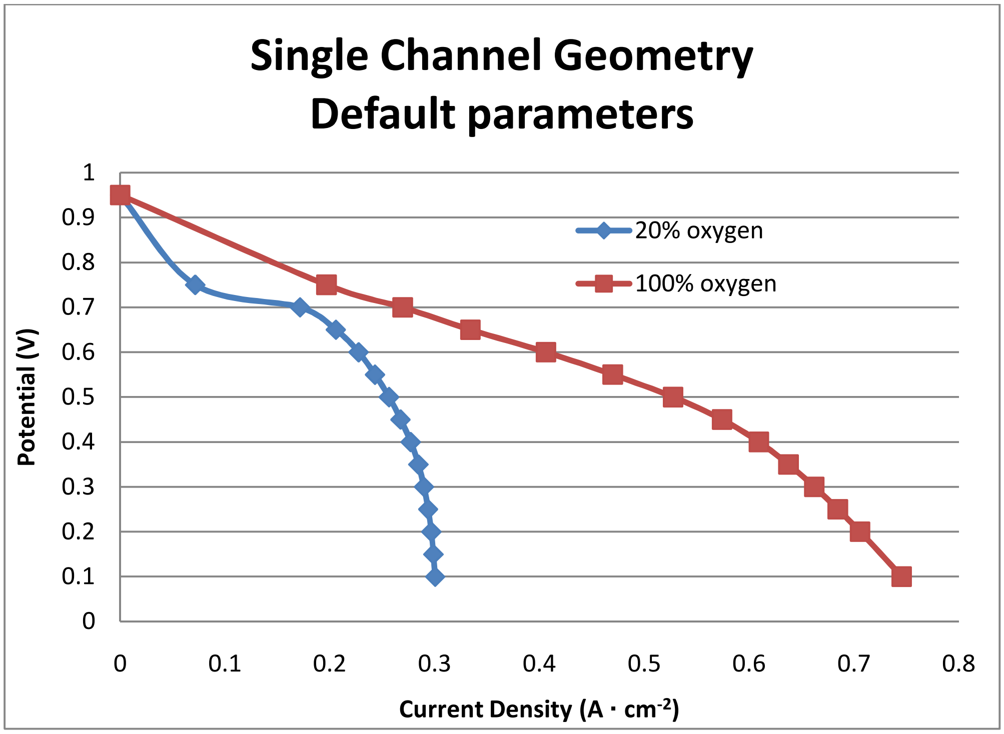


Figure 18. Single channel simulation results.

The purpose of the single channel simulation was a proof of concept and exploration of the usability of the software and the feasibility of running more complicated simulations. With those goals in mind, this simulation was a major success and represents a tangible first-step toward generating simulations based on real geometries, model parameters and operating conditions. The simulations produce results consistent with experimental measurements between fuel cells operating on air and pure oxygen. This feasibility study also demonstrated the need for tighter data points at voltages very close to the open circuit voltage.

The next fuel cell simulations represent a large step-up in complexity. A fuel cell geometry based on the parallel design described by Iranzo et al. (2010) was modeled using the basic steps for meshing with ICEM CFD. The workbench mesher was not sufficiently flexible to mesh a parallel geometry without creating many bad elements. Channel height and width is 2.0 mm. Rib width is 0.8 mm. Overall fuel cell width and length is 7.0 cm.

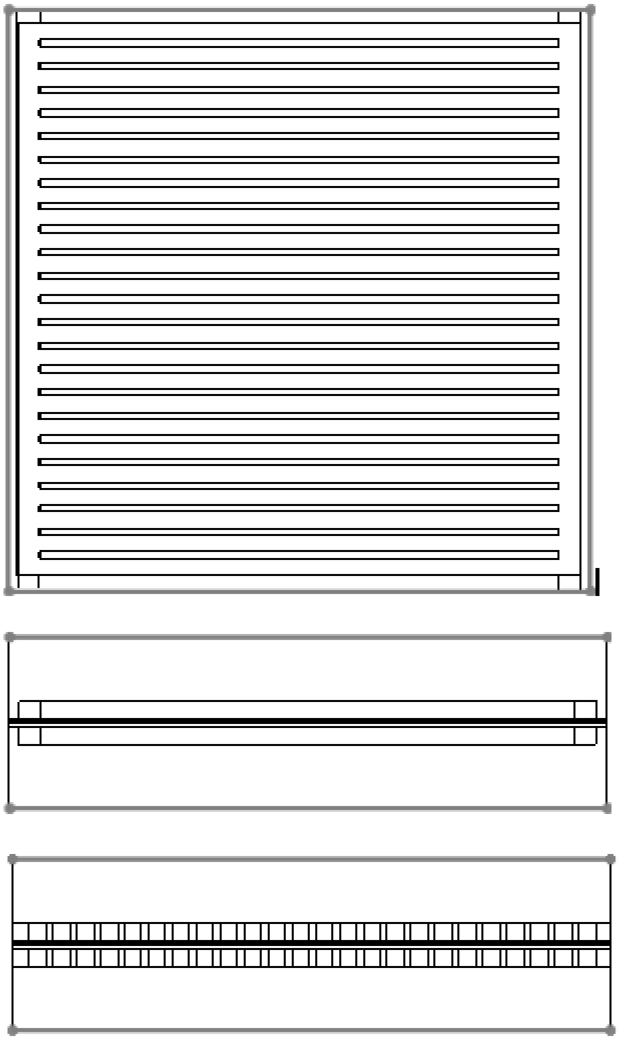


Figure 19. Parallel fuel cell geometry top, front and side wireframe views.

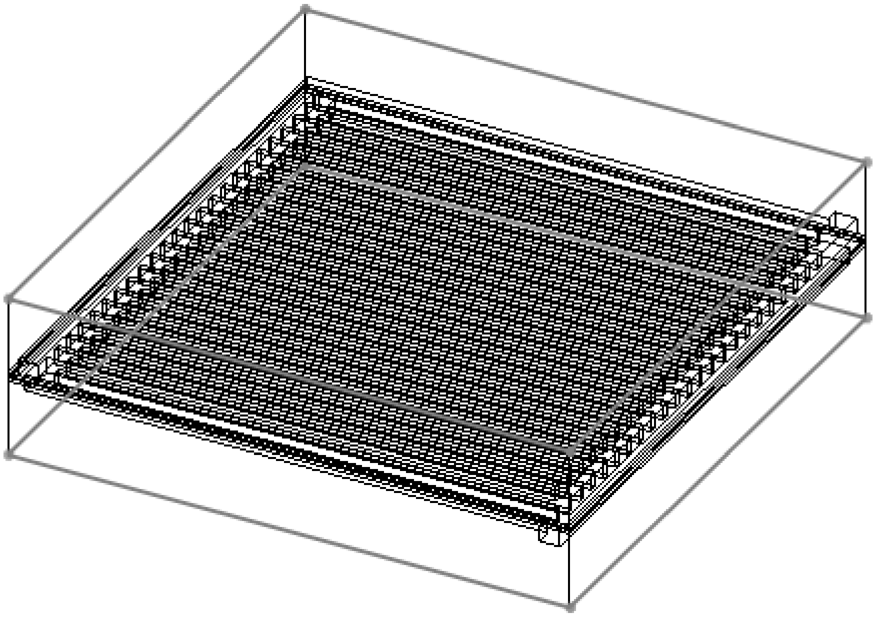


Figure 20. Parallel fuel cell geometry isometric view.

An initial run of this geometry was performed using the default model parameters to determine the effect of inputting more realistic model parameters. Temperature was fixed at 333K for all simulations. Anode gas flow rates were 1x10^{-6} at the anode with mass fractions of 0.8 hydrogen and 0.2 water. Cathode gas flow rate was set at 5x10^{-5} with mass fractions of 0.2 oxygen and 0.1 water. Operating pressure was set at 1 atm for the default value and 4 atm for all other simulations. The default models included joule heating, reaction heating, electrochemistry sources, Butler-Volmer rate and membrane water transport. All other simulations included the default models plus multiphase. The results of this analysis showed some important inconsistencies in the way the model parameters were input. These inconsistencies were caused by differences in unit from standard fuel cell conventions and the units required for the model.

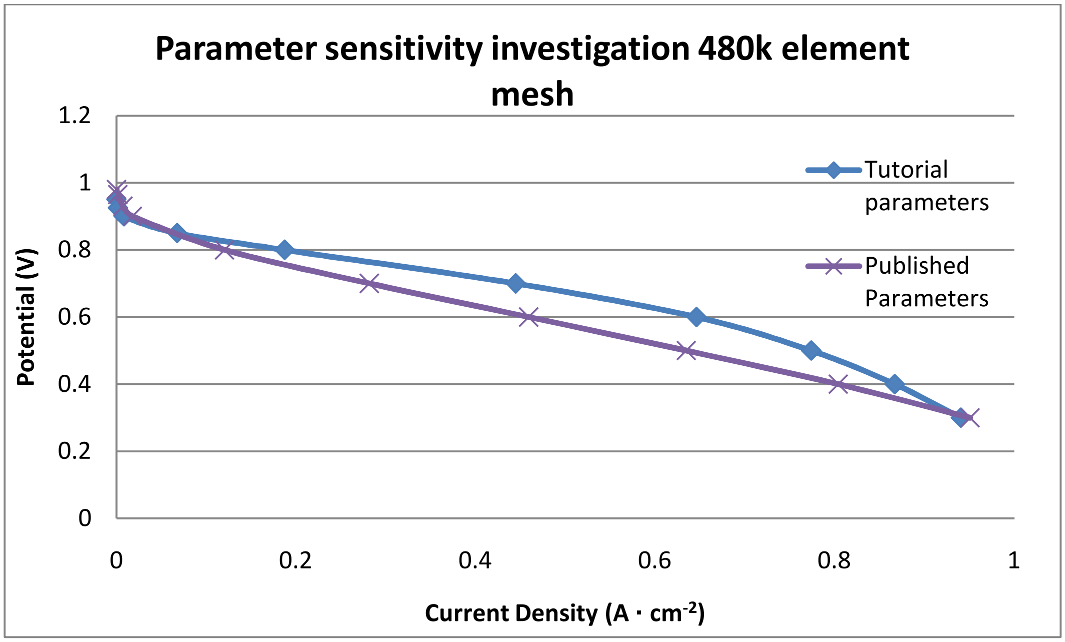


Figure 21. Parameter sensitivity investigation with 480k element mesh.

This data shows the results of running this geometry with tutorial model parameters and with the parameters from the simulation by Iranzo et al. (2010). The data demonstrates that using different model and material parameters will change the solution output. This is an important basic verification of the software functionality.

The next simulation used the following parameters based on the simulation parameters used by Iranzo et al (2010). These parameters were used in all the following simulation results. All available modeling options were included in the simulations except multicomponent diffusion and anisotrpic e-conductivity.

Table 3

Model Parameters

Parameter	Units	Value
BP density	kg/m^3	1990
BP specific heat capacity	J/(kg·K)	710
BP thermal conductivity	W/(m·K)	120
BP electric conductivity	1/(Ω·m)	92,600
BP thickness	mm	9.5
BP GDL contact resistance	$\Omega \cdot m^2$	4.56×10^{-6}
GDL density	kg/m^3	321.5
GDL porosity	e	0.82
GDL electric conductivity	1/(Ω·m)	280
GDL viscous resistance (anode)	$1/m^2$	1.00×10^{12}
GDL viscous resistance (cathode)	$1/m^2$	3.86×10^{12}
GDL wall contact angle	deg	110
GDL thickness	μm	420
CL surface-to-volume ratio	m^2/m^3	1.25×10^7
CL thickness (anode)	μm	6
CL thickness (cathode)	μm	12
Membrane density	kg/m^3	1980
Membrane thermal conductivity	W/(m·K)	0.16
Membrane equivalent weight	kg/kmol	1100
Membrane thickness	μm	175
Open circuit voltage	V	0.98
H_2 diffusivity	m^2/s	8.0×10^{-5}
O_2 diffusivity	m^2/s	2.0×10^{-5}
H_2O diffusivity	m^2/s	$5 .0 \times 10^{-5}$
Pore blockage saturation exponent	e	2
Concentration exponent (anode)	e	0.5
Concentration exponent (cathode)	e	1
Charge transfer coefficient (anode)	e	1
Charge transfer coefficient (cathode)	e	1
Reference exchange current density (anode)	A/m^2	4.48×10^5
Reference exchange current density (cathode)	A/m^2	4.48
Electrolyte Projected Area	m^2	0.0049

The goal for this round of simulations was to perform a catalyst layer and membrane mesh sensitivity analysis. The mesh parameters in the current collector and flow channels were held constant while the number of cells within the membrane and catalyst layers were varied independently. Operating temperature was set at 333K and operating pressure was 401,325 kPa. Anode mass flow inlet rate was 5 x10^{-6} with 0.5/0.5 H_2 to H_2O mass ratio. Cathode inlet rate was set at 8.8 x10^{-5} kg/s with a 0.1785/0.15 ratio of O_2 and H_2O representative of 100% humidification at this temperature. Solution was deemed convergent after 200 iterations based on results from the previous simulation. Mesh size varied depending on the number of divisions in the catalyst layer and membrane. The smallest mesh size contained approximately 300k elements with the largest containing approximately 750k elements.

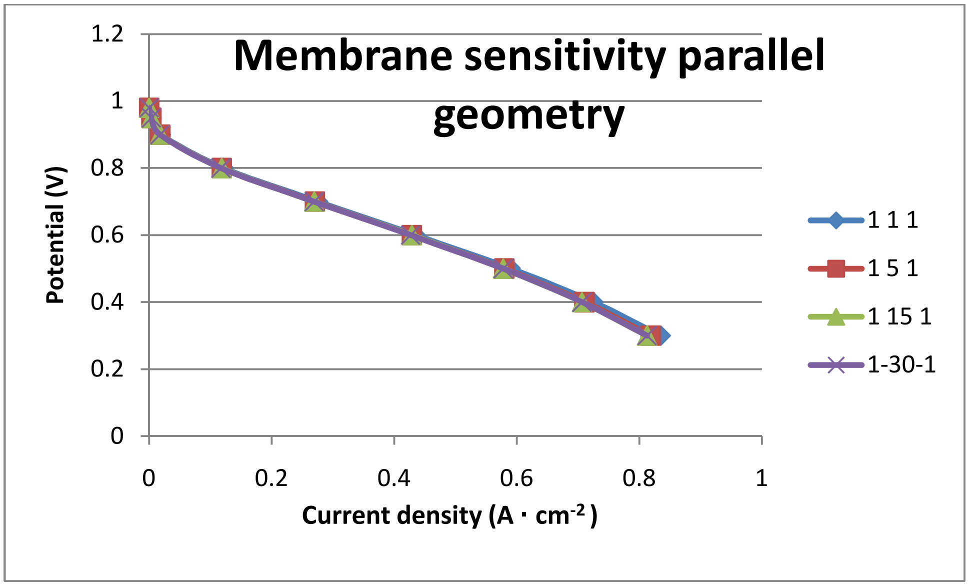


Figure 22. Results from holding the catalyst layer constant and varying the number of vertical cells in the membrane.

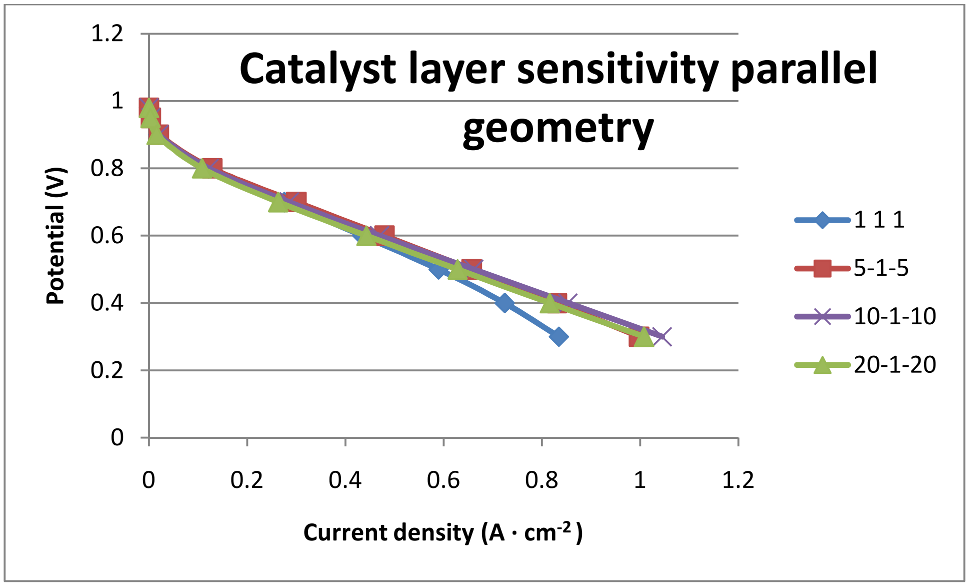


Figure 23. Results from holding the membrane layer constant and varying the number of cells in the catalyst layer

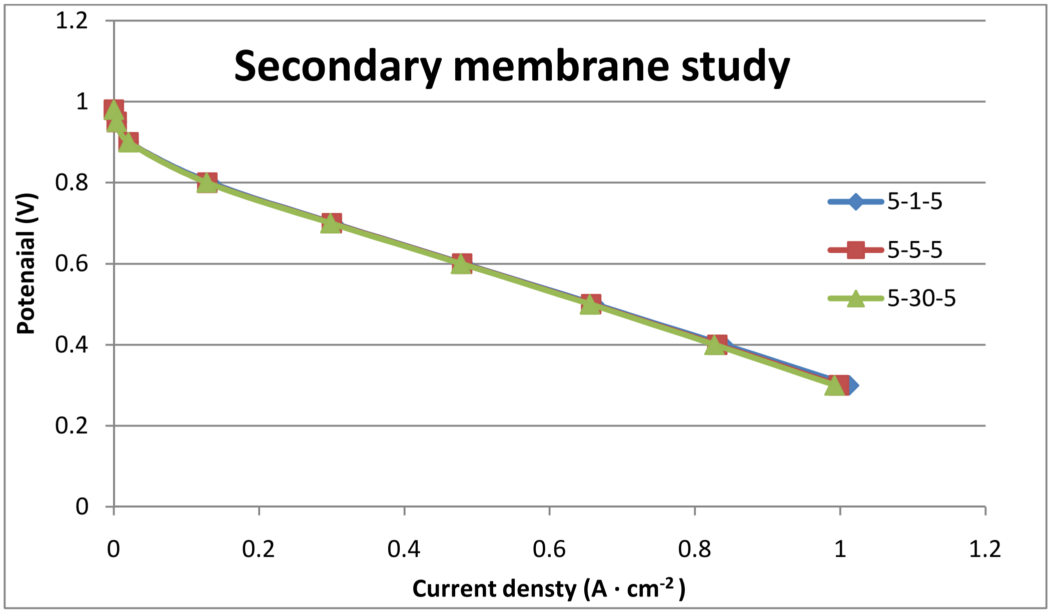


Figure 24. Composite study showing 5 divisions in the catalyst layer held constant while membrane cell-number varied.

The results of this study show that varying the number of computational cells within the membrane makes no difference, while varying the catalyst layer does make a significant difference. However, the values in the catalyst layer are inconsistent with expected results. As the number of divisions increases, the overall current density increases until 10 divisions and the results for 20 divisions show lower current densities. This is an indication that the solution is not converging.

Conversations with experts in the field revealed the need for the multiple part conversion criteria discussed in chapter 3 as well as an iteration sensitivity analysis. Reported current density needs to be matched with fuel consumption rates and the solution must be independent of the number of iterations. This new information invalidated the previous results as it cannot be determined if the results had converged.

Another analysis was initiated using a single channel serpentine geometry containing 640k elements. The operating and material properties for this geometry were identical to the previous simulation. The dimensions of the MEA and the overall dimensions of the current collectors are identical to the parallel geometry. A single channel serpentine design was created. Channel width is 1.1 mm wide and 0.8 mm tall. Land channels were 1.0 mm wide.

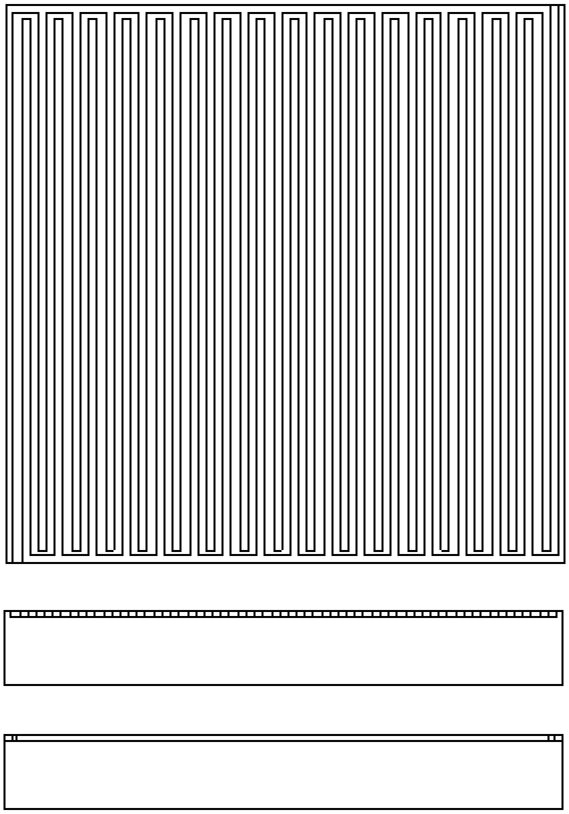


Figure 25. Single channel serpentine flow field geometry. Top and side views.

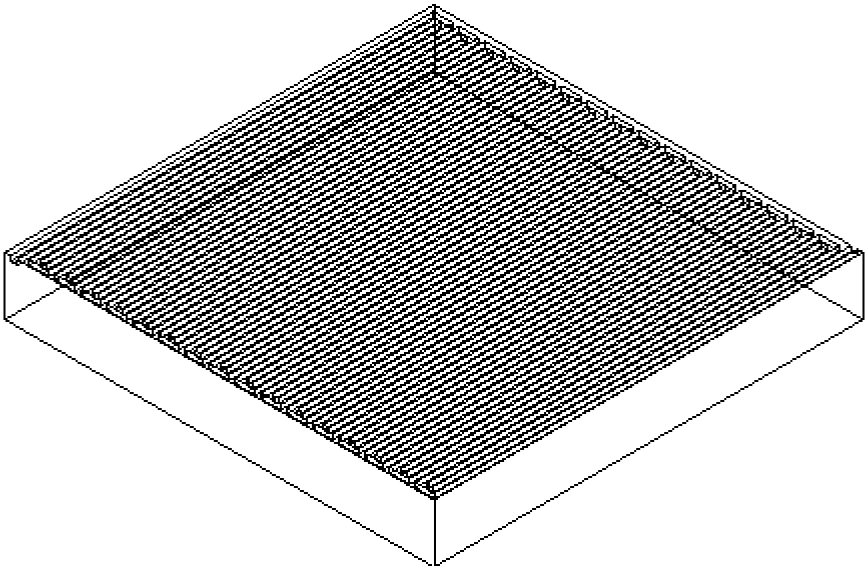


Figure 26. Serpentine flow field isometric view.

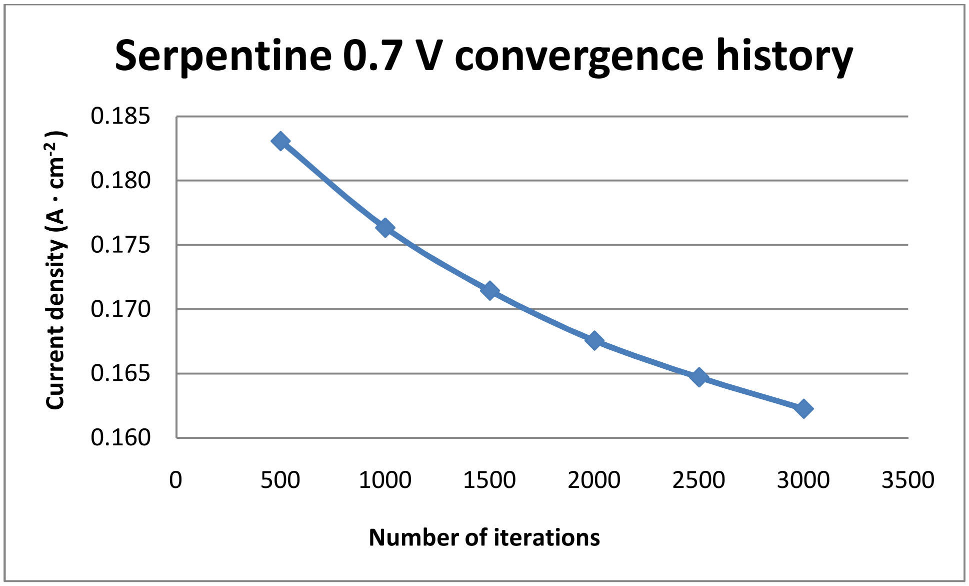


Figure 90. Serpentine convergence history at 0.7 V

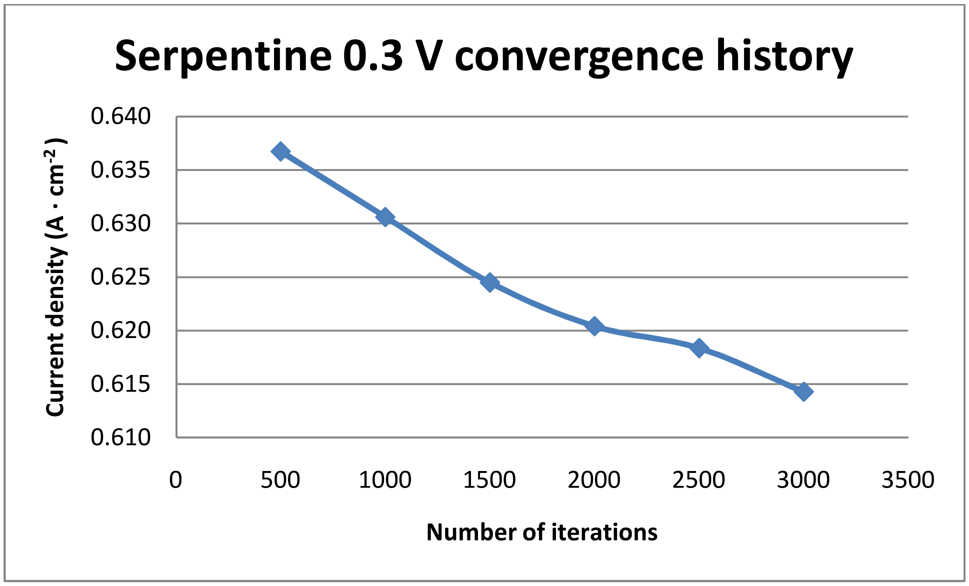


Figure 91. Serpentine convergence history at 0.3 V

These graphs show the convergence history at a low current density and at a high current density. Expected behavior for a convergent result would be a flat curve that does not increase or decrease with increasing iterations. These data suggest that far more than 3000 iterations per data point is necessary to ensure that solutions are insensitive to further iterations. The 0.7 V graph shows signs of decreasing in slope which suggests convergence while the 0.3 V graph shows a non-linear behavior. The non-linear behavior of the 0.3 V graph suggests there is a problem with the mesh. It was determined that the number of elements in that mesh was insufficient to capture the geometry and parts of the anode catalyst layer had merged with the anode GDL.

An iteration sensitivity analysis of using 19,000 iterations was used to determine the final behavior of the graphs. A new mesh was created for the serpentine geometry with approximately 985k elements. Only 1 computational cell within the membrane and catalyst layers was used to generate a baseline simulation. Three curves were generated at 0.7V, 0.5V and 0.3V to determine the effect of current density on convergence.

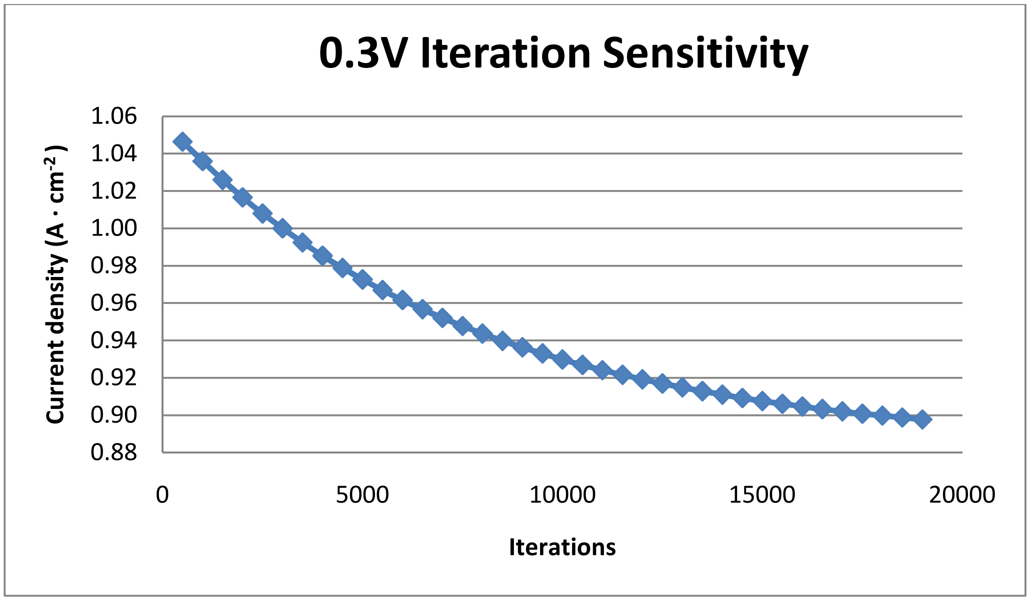


Figure 27. Graph shows that the current density value begins to converge to a single value after nearly 20,000 iterations

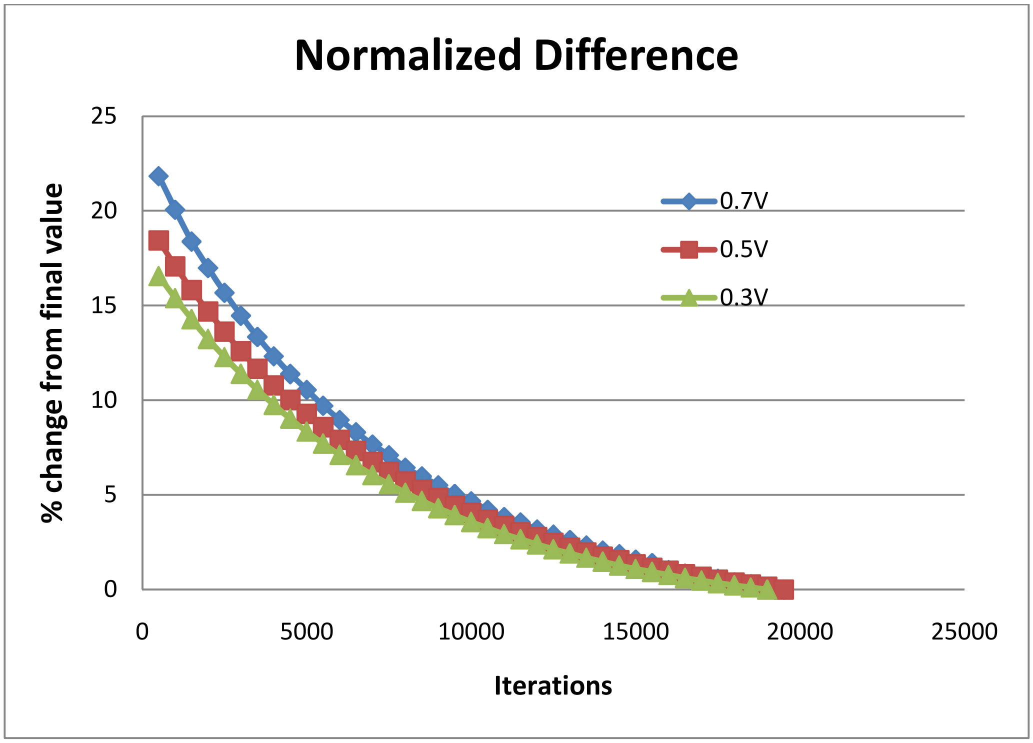


Figure 28. Normalized difference from final value for different cell voltage values.

Results from the 0.3 V iteration sensitivity test show that the difference between successive data points begins to shrink as the number of iterations increase. Furthermore, the curve resembles an exponential decay suggesting asymptotic end behavior. In figure 28, the final value is assumed as correct and the normalized difference from the final point is displayed. This data shows that lower voltage simulations converge after fewer iterations compared to higher current density simulations. The data also shows that at approximately 10,000 iterations are required to reach within 5% of the value at 19,000 iterations.

Two important assumptions are made from the presented iteration study. First, due to the limited precision available in real computing, the curve is asymptotic and will never become perfectly flat. Second, the slope of the curve will continue to approach zero. An estimation of the minimum number of iterations required to reach further change can be estimated. Using the average slope of the final 5 data points it can be calculated that in order to produce a 1% change from the value calculated at 19,000 iterations for the 0.7v case, an additional 8,000 iterations would be required.

The allocated computing power was 4 cores of a 3.2 GHz Xeon EM64T compute node. 640,000 element mesh used for the serpentine simulation was able to generate approximately 6 seconds per iteration. The iteration sensitivity analysis used a mesh with 985,000 elements. The computation rate using the same hardware was approximately 9 seconds per iteration. The parallel processing is limited by the number of available licenses, which is currently capped at 4 due to cost and a lack of regular users of this software at the facility.

The data was collecting using automatic job scripts which produced a data file for every point. The contents of the job script and the output data file for the 0.7 V at 500 iterations can be found in Appendix C. An example queue file necessary for running a job on a Linux based cluster is also in Appendix C.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

There are myriad challenges and difficulties in creating successful fuel cell simulations. The prohibitive cost of the software package is an important factor in erecting barriers to creating and experimenting with more comprehensive fuel cell simulations. However, many large institutions that already do a lot of CFD modeling may already have the software available, and the additional software license to run the fuel cell software is all that is needed to begin additional modeling work in this important area of research.

Research into PEM fuel cells is becoming more important with the growing concern over the ever-decreasing supply of fossil fuels and the corresponding concern over the increase in greenhouse gasses. Working to improve fuel cells so that they are commercially viable and practical in their application is critical to creating a feasible alternative to our current global dependence on fossil fuels. . Fuel cells are a fantastic alternative to fossil fuels because they don’t produce greenhouse gasses, and because hydrogen can be produced from renewable energies. PEM Fuel cells are also efficient, quiet and run at low temperatures, so they can be used in transportation applications, which is arguably where viable renewable energy technologies are needed the most. However, many improvements are needed before PEM fuel cells are sufficiently resilient and reliable enough to be used in wide-scale transportation applications. Part of the process of improving PEM fuel cells absolutely must include CFD modeling and simulation. Modeling helps decrease the cost, and increase the speed at which, designs can be verified while improving fundamental understanding of the fuel cell's operation. Computing power continues a trend of decreasing in cost, so any simulations can be run relatively quickly and cheaply. The availability of commercial packages makes developing models from scratch unnecessary, thereby saving a lot of development time. Additionally, there are also many published model verifications against which you can check your simulation results to ensure accuracy.

However, even with a developed software package, there are many difficulties in producing a good model. The geometrical model must accurately represent the physical object. Oftentimes, the boundaries between regions will not be clearly defined and approximations or simplifications will have to be made. CAD modeling is an area used in many engineering fields, so the subject is well understood and there is plenty of help available both from the software vendors and the CAD community.

Decomposing the geometry into a high-quality computational mesh represents one of the major challenges to CFD modeling of fuel cells. A balance must be struck between creating a fine-enough-mesh to capture the geometry, while also keeping computational times low. Sometimes clever meshing schemes must be employed to avoid creating bad elements which will invalidate simulation results. There are many subtle problems that can arise during the meshing process, which will prevent a solution from being possible. Meshes can quickly reach very large sizes requiring high powered computers just to render the mesh.

Creating an accurate simulation of a PEM fuel cell requires that a large number of parameters are known. These parameters can be difficult to measure and can require sophisticated experimental techniques to determine. Other parameters can be assumed or estimated with differing degrees of confidence. There are many publications with measured or estimated values for many materials used in fuel cells which can give acceptable parameters for a simulation.

Solving the system of non-linear coupled equations that describe the operation of a fuel cell is a difficult task. CFD is a widely developed field and there are many techniques to ensure convergent solutions. The Fluent software package has many available algorithms built-in and the fuel cell manual gives some suggestions on which ones are most appropriate to solve this type of simulation.

Regardless of the algorithm chosen, it is important to verify the convergence of your simulation. The three-part verification process of using iteration sensitivity, total current flux integral, current density and mass species conservation outlined in this work should ensure that subsequent simulations will produce consistent results within the error margin defined by the user.

Future work in this field at this facility will require a dedicated work station or computer lab with all of the available software installed on powerful machines making the entire modeling workflow possible at a single location. Having easily accessible hardware and software resources in a single facility will greatly reduce the time required to learn the software. Parallel computing will play an important role with larger and more sophisticated simulations in order to take advantage of the parallel computing available thereby reducing the computation times.

REFERENCES

Ahn, J. (2008). Coolant controls of a PEM fuel cell system. *Journal of Power Sources, 179*(1), 252-264.

Ansys Corporation. Retrieved 04/01, 2011, from https://www1.ansys.com/customer/content/documentation/121/fluent/flfuelcells.pdf

Cheddie, D. F., & Munroe, N. D. H. (2006). Three dimensional modeling of high temperature PEM fuel cells. *Journal of Power Sources, 160*(1), 215-223. doi:DOI: 10.1016/j.jpowsour.2006.01.035

Chen, C. (2008). Three-dimensional numerical analysis of PEM fuel cells with straight and wave-like gas flow fields channels. *Journal of Power Sources, 177*(1), 96-103.

Dawes, J. E., Hanspal, N. S., Family, O. A., & Turan, A. (2009). Three-dimensional CFD modelling of PEM fuel cells: An investigation into the effects of water flooding. *Chemical Engineering Science, 64*(12), 2781-2794. doi:DOI: 10.1016/j.ces.2009.01.060

Dhathathreyan, K. S. (2008). Nanostructured platinum catalyst layer prepared by pulsed electrodeposition for use in PEM fuel cells. *International Journal of Hydrogen Energy, 33*(20), 5672-5677.

Escobet, T., Feroldi, D., de Lira, S., Puig, V., Quevedo, J., Riera, J., & Serra, M. (2009). Model-based fault diagnosis in PEM fuel cell systems. *Journal of Power Sources, 192*(1), 216-223. doi:DOI: 10.1016/j.jpowsour.2008.12.014

Friedmann, R. (2009). *Optimization of the Cathode Catalyst Layer Composition of a PEM Fuel Cell using a Novel 2-Step Preparation Method,*

Grigoriev, S. A. (2010). Platinum and palladium nano-particles supported by graphitic nano-fibers as catalysts for PEM water electrolysis. *International Journal of Hydrogen Energy,*

Hernandez-Guerrero, A. (2010). Current density and polarization curves for radial flow field patterns applied to PEMFCs (proton exchange membrane fuel cells). *Energy (Oxford), 35*(2), 920-927.

Huang, Y. (2009). Flow field optimization for proton exchange membrane fuel cells with varying channel heights and widths. *Electrochimica Acta, 54*(23), 5522-5530.

Iranzo, A., Muñoz, M., Rosa, F., & Pino, J. (2010). Numerical model for the performance prediction of a PEM fuel cell. model results and experimental validation. *International Journal of Hydrogen Energy, 35*(20), 11533-11550. doi:DOI: 10.1016/j.ijhydene.2010.04.129

Kamarajugadda, S. (2008). On the implementation of membrane models in computational fluid dynamics calculations of polymer electrolyte membrane fuel cells. *Computers & Chemical Engineering, 32*(7), 1650-1660.

Larminie, J. (2003). Fuel cell-technology handbook. *Proceedings of the Institution of Mechanical Engineers.Part D, Journal of Automobile Engineering, 217*(5), 415-415.

Liu, H. (2005). A two-phase flow and transport model for PEM fuel cells. *Journal of Power Sources, 155*(2), 219-230.

Lu, W. (2007). Surface-modified carbons as platinum catalyst support for PEM fuel cells. *Carbon (New York), 45*(7), 1506-1517.

MARIE, J. (2004). Highly dispersed platinum on carbon aerogels as supported catalysts for PEM fuel cell-electrodes: Comparison of two different synthesis paths. *Journal of Non-Crystalline Solids, 350*, 88-96.

Meng, H. (2006). A three-dimensional PEM fuel cell model with consistent treatment of water transport in MEA. *Journal of Power Sources, 162*(1), 426-435.

Meng, H. (2008). A PEM fuel cell model for cold-start simulations. *Journal of Power Sources, 178*(1), 141-150. doi:DOI: 10.1016/j.jpowsour.2007.12.035

Min, C. (2008). Performance of a proton exchange membrane fuel cell with a stepped flow field design. *Journal of Power Sources, 186*(2), 370-376.

Prasad, A. (2010). In situ comparison of water content and dynamics in parallel, single-serpentine, and interdigitated flow fields of polymer electrolyte membrane fuel cells. *Journal of Power Sources, 195*(11), 3553-3568.

Qian, Y. (2004). *Preparation of Platinum-Based Bimetallic Catalysts for the Oxygen Reduction Reaction in PEM Fuel Cells,*

Rambabu, B. (2010). Platinum/tin oxide/carbon cathode catalyst for high temperature PEM fuel cell. *Journal of Power Sources, 195*(13), 3977-3983.

Sadiq Al-Baghdadi, M. (2008). Three-dimensional computational fluid dynamics model of a tubular-shaped PEM fuel cell. *Renewable Energy, 33*(6), 1334-1345.

Saejeng, Y. (2010). Catalyst electrode preparation for PEM fuel cells by electrodeposition. *Journal of Applied Electrochemistry, 40*(5), 903-910.

Siegel, C. (2008). Review of computational heat and mass transfer modeling in polymer-electrolyte-membrane (PEM) fuel cells. *Energy, 33*(9), 1331-1352. doi:DOI: 10.1016/j.energy.2008.04.015

Steinkamp, K. (2008). A nonisothermal PEM fuel cell model including two water transport mechanisms in the membrane. *Journal of Fuel Cell Science and Technology, 5*(1)

Teki, R. (2009). Sputter-deposited pt PEM fuel cell electrodes: Particles vs layers. *Journal of the Electrochemical Society, 156*(5), B614-B619.

Wang, J. -., Savinell, R. F., Wainright, J., Litt, M., & Yu, H. (1996). A H2/O2 fuel cell using acid doped polybenzimidazole as polymer electrolyte. *Electrochimica Acta, 41*(2), 193-197. doi:DOI: 10.1016/0013-4686(95)00313-4

Wang, L. (2003). A parametric study of PEM fuel cell performances. *International Journal of Hydrogen Energy, 28*(11), 1263-1272.

Wang, L., Husar, A., Zhou, T., & Liu, H. (2003). A parametric study of PEM fuel cell performances. *International Journal of Hydrogen Energy, 28*(11), 1263-1272. doi:DOI: 10.1016/S0360-3199(02)00284-7

Wang, X. (2008). A numerical study of flow crossover between adjacent flow channels in a proton exchange membrane fuel cell with serpentine flow field. *Journal of Power Sources, 185*(2), 985-992.

Wang, X. (2008). Effects of flow channel geometry on cell performance for PEM fuel cells with parallel and interdigitated flow fields. *Electrochimica Acta, 53*(16), 5334-5343.

Wang, X. (2011). Transient characteristics of proton exchange membrane fuel cells with different flow field designs. *Journal of Power Sources, 196*(1), 235.

Wang, Y., Basu, S., & Wang, C. (2008). Modeling two-phase flow in PEM fuel cell channels. *Journal of Power Sources, 179*(2), 603-617. doi:DOI: 10.1016/j.jpowsour.2008.01.047

Zhang, H. (2010). The conception of in-plate adverse-flow flow field for a proton exchange membrane fuel cell. *International Journal of Hydrogen Energy, 35*(17), 9124-9133.

Zhang, S., Yuan, X., Hin, J. N. C., Wang, H., Friedrich, K. A., & Schulze, M. (2009). A review of platinum-based catalyst layer degradation in proton exchange membrane fuel cells. *Journal of Power Sources, 194*(2), 588-600. doi:DOI: 10.1016/j.jpowsour.2009.06.073

Zhukovsky, M. S., Fomina, L. V., & Beznosyuk, S. A. (2011). Computer modeling of hydrogen fuel cell subsystems: Carbon nanogel electrodes and fractal nanoparticle catalysts. *International Journal of Hydrogen Energy, 36*(1), 1212-1216. doi:DOI: 10.1016/j.ijhydene.2010.06.128

APPENDIX A

GRAPH DATA

Single channel geometry default parameters graph data		
OCV=.95	Electrode area = 30cm^2	
	Total Current Flux (A)	
Voltage	20% O$_2$(0.1 H$_2$0, 0.2 O$_2$)	100% O$_2$ (.1 H$_2$O, .9 O$_2$)
0.95	0	0
0.75	2.147404	5.9
0.7	5.143	8.08
0.65	6.17	10.02
0.6	6.82	12.18
0.55	7.29	14.09
0.5	7.69	15.82
0.45	8.02	17.22
0.4	8.31	18.27
0.35	8.53	19.12
0.3	8.69	19.86
0.25	8.81	20.53
0.2	8.9	21.17
0.15	8.96	
0.1	9.01	22.36

Membrane sensitivity parallel geometry				
Voltage	1-1-1 Current(A)	1-5-1 (A)	1-15-1 (A)	1-30-1 (A)
0.98	0	0	0	0
0.95	0.181	0.169	0.167	0.167
0.9	0.927	0.875	0.867	0.866
0.8	5.95	5.8	5.8	5.74
0.7	13.5	13.25	13.2	13.2
0.6	21.3	21	21	20.9
0.5	28.9	28.4	28.3	28.3
0.4	35.5	34.8	34.6	34.6
0.3	40.9	40.1	39.8	39.8
Catalyst layer sensitivity parallel geometry				
voltage	1-1-1 (A)	5-1-5 (A)	10-1-10 (A)	20-1-20 (A)
0.98	0	0	0	0
0.95	0.181	0.19	0.187	0.145
0.9	0.927	0.99	0.97	0.759
0.8	5.95	6.3	6	5.3
0.7	13.5	14.7	14	12.9

0.6	21.3	23.5	23	21.7
0.5	28.9	32.2	32.4	30.8
0.4	35.5	40.7	41.8	40
0.3	40.9	48.9	51.2	49.4
Secondary Membrane study				
Voltage	5-1-5 (A)	5-5-5 (A)	5-30-5 (A)	
0.98	0	0	0	
0.95	0.2	0.19	0.186	
0.9	1	0.99	0.98	
0.8	6.5	6.3	6.26	
0.7	14.8	14.7	14.6	
0.6	23.6	23.5	23.4	
0.5	32.4	32.2	32.1	
0.4	41.1	40.7	40.5	
0.3	49.6	48.9	48.6	

	Iteration sensitivity analysis					
	Number of iterations					
	500	1000	1500	2000	2500	3000
Voltage	Total current (A)					
0.7	8.97 A	8.64	8.4	8.21	8.07	7.95
0.3	31.2 A	30.9	30.6	30.4	30.3	30.1

Iteration Sensitivity Analysis							
	0.7v	0.5V	0.3V		0.7v	0.5V	0.3V
Iteration	Current (A)	Current (A)	Current (A)	Iteration	Current (A)	Current (A)	Current (A)
500	13.06	32.19	51.27	10500	11.17	28.18	45.42
1000	12.87	31.82	50.76	11000	11.13	28.09	45.28
1500	12.69	31.48	50.27	11500	11.10	28.00	45.16
2000	12.54	31.17	49.81	12000	11.06	27.93	45.04
2500	12.40	30.88	49.39	12500	11.03	27.85	44.93
3000	12.27	30.60	49.00	13000	11.00	27.78	44.83
3500	12.15	30.35	48.63	13500	10.97	27.71	44.73
4000	12.04	30.11	48.28	14000	10.94	27.65	44.64
4500	11.94	29.90	47.96	14500	10.92	27.60	44.55
5000	11.85	29.70	47.66	15000	10.89	27.54	44.47
5500	11.76	29.51	47.38	15500	10.87	27.49	44.40
6000	11.68	29.33	47.12	16000	10.83	27.45	44.33
6500	11.61	29.17	46.88	16500	10.81	27.40	44.26
7000	11.54	29.01	46.65	17000	10.79	27.36	44.20
7500	11.48	28.87	46.44	17500	10.78	27.32	44.14
8000	11.41	28.73	46.24	18000	10.76	27.28	44.09
8500	11.36	28.61	46.05	18500	10.75	27.25	44.04
9000	11.31	28.49	45.88	19000	10.73	27.22	43.99
9500	11.26	28.38	45.72	19500	10.72	27.18	
10000	11.22	28.28	45.56				

APPENDIX B

TABLE OF MASS FLOW RATES CONVERSIONS UP TO 400 AMPERES

	SCCM			kg/s		
Current (A)	H_2	O_2	air	H_2	O_2	air
1	7	3.5	16.67	1.0E-08	8.3E-08	3.6E-07
2	14	7	33.34	2.1E-08	1.7E-07	7.1E-07
3	21	10.5	50.01	3.1E-08	2.5E-07	1.1E-06
4	28	14	66.68	4.2E-08	3.3E-07	1.4E-06
5	35	17.5	83.35	5.2E-08	4.2E-07	1.8E-06
10	70	35	166.7	1.0E-07	8.3E-07	3.6E-06
15	105	52.5	250.05	1.6E-07	1.3E-06	5.4E-06
20	140	70	333.4	2.1E-07	1.7E-06	7.1E-06
25	175	87.5	416.75	2.6E-07	2.1E-06	8.9E-06
30	210	105	500.1	3.1E-07	2.5E-06	1.1E-05
35	245	122.5	583.45	3.6E-07	2.9E-06	1.3E-05
40	280	140	666.8	4.2E-07	3.3E-06	1.4E-05
45	315	157.5	750.15	4.7E-07	3.8E-06	1.6E-05
50	350	175	833.5	5.2E-07	4.2E-06	1.8E-05
60	420	210	1000.2	6.3E-07	5.0E-06	2.1E-05
70	490	245	1166.9	7.3E-07	5.8E-06	2.5E-05
80	560	280	1333.6	8.3E-07	6.7E-06	2.9E-05
90	630	315	1500.3	9.4E-07	7.5E-06	3.2E-05
100	700	350	1667	1.0E-06	8.3E-06	3.6E-05
120	840	420	2000.4	1.3E-06	1.0E-05	4.3E-05
140	980	490	2333.8	1.5E-06	1.2E-05	5.0E-05
160	1120	560	2667.2	1.7E-06	1.3E-05	5.7E-05
180	1260	630	3000.6	1.9E-06	1.5E-05	6.4E-05
200	1400	700	3334	2.1E-06	1.7E-05	7.1E-05
220	1540	770	3667.4	2.3E-06	1.8E-05	7.9E-05
240	1680	840	4000.8	2.5E-06	2.0E-05	8.6E-05
260	1820	910	4334.2	2.7E-06	2.2E-05	9.3E-05
280	1960	980	4667.6	2.9E-06	2.3E-05	1.0E-04
300	2100	1050	5001	3.1E-06	2.5E-05	1.1E-04
320	2240	1120	5334.4	3.3E-06	2.7E-05	1.1E-04
340	2380	1190	5667.8	3.5E-06	2.8E-05	1.2E-04
360	2520	1260	6001.2	3.8E-06	3.0E-05	1.3E-04
380	2660	1330	6334.6	4.0E-06	3.2E-05	1.4E-04
400	2800	1400	6668	4.2E-06	3.3E-05	1.4E-04

APPENDIX C

640K SERPENTINE ITERATION SENSITIVITY JOB SCRIPT AND OUTPUTS

CONTENS OF S640k.jou

```
;
;
/file/read-case /home/aarvay/serp/s640.cas
;
;
;
/solve/set/under-relaxation species-0 .95
/solve/set/under-relaxation species-1 .95
/solve/set/under-relaxation species-2 .95
;
/solve/initialize/initialize-flow
;
/define/boundary-conditions/wall terminal_c , , , , , , , , , , , , .70 , ,
/solve/iterate 500
/file/start-transcript  /home/aarvay/serp/s640-70v-500.txt
/report/surface-integrals/vertex-avg terminal_c , electric-potential n
/report/surface-integrals/integral terminal_c , y-current-flux-density n
/report/species-mass-flow
/file/stop-transcript
/solve/iterate 500
/file/start-transcript  /home/aarvay/serp/s640-70v-1000.txt
/report/surface-integrals/vertex-avg terminal_c , electric-potential n
/report/surface-integrals/integral terminal_c , y-current-flux-density n
/report/species-mass-flow
/file/stop-transcript
/solve/iterate 500
/file/start-transcript  /home/aarvay/serp/s640-70v-1500.txt
/report/surface-integrals/vertex-avg terminal_c , electric-potential n
/report/surface-integrals/integral terminal_c , y-current-flux-density n
/report/species-mass-flow
/file/write-case-data /home/aarvay/serp/s640-70v-1500.cas
/file/stop-transcript
/solve/iterate 500
/file/start-transcript  /home/aarvay/serp/s640-70v-2000.txt
/report/surface-integrals/vertex-avg terminal_c , electric-potential n
/report/surface-integrals/integral terminal_c , y-current-flux-density n
/report/species-mass-flow
/file/stop-transcript
/solve/iterate 500
/file/start-transcript  /home/aarvay/serp/s640-70v-2500.txt
/report/surface-integrals/vertex-avg terminal_c , electric-potential n
/report/surface-integrals/integral terminal_c , y-current-flux-density n
/report/species-mass-flow
/file/stop-transcript
/solve/iterate 500
/file/start-transcript  /home/aarvay/serp/s640-70v-3000.txt
/report/surface-integrals/vertex-avg terminal_c , electric-potential n
/report/surface-integrals/integral terminal_c , y-current-flux-density n
/report/species-mass-flow
```

```
/file/write-case-data /home/aarvay/serp/s640-70v-3000.cas
/file/stop-transcript
/define/boundary-conditions/wall terminal_c , , , , , , , , , , , , .30 , ,
/solve/iterate 500
/file/start-transcript  /home/aarvay/serp/s640-30v-500.txt
/report/surface-integrals/vertex-avg terminal_c , electric-potential n
/report/surface-integrals/integral terminal_c , y-current-flux-density n
/report/species-mass-flow
/file/stop-transcript
/solve/iterate 500
/file/start-transcript  /home/aarvay/serp/s640-30v-1000.txt
/report/surface-integrals/vertex-avg terminal_c , electric-potential n
/report/surface-integrals/integral terminal_c , y-current-flux-density n
/report/species-mass-flow
/file/stop-transcript
/solve/iterate 500
/file/start-transcript  /home/aarvay/serp/s640-30v-1500.txt
/report/surface-integrals/vertex-avg terminal_c , electric-potential n
/report/surface-integrals/integral terminal_c , y-current-flux-density n
/report/species-mass-flow
/file/write-case-data /home/aarvay/serp/s640-30v-1500.cas
/file/stop-transcript
/solve/iterate 500
/file/start-transcript  /home/aarvay/serp/s640-30v-2000.txt
/report/surface-integrals/vertex-avg terminal_c , electric-potential n
/report/surface-integrals/integral terminal_c , y-current-flux-density n
/report/species-mass-flow
/file/stop-transcript
/solve/iterate 500
/file/start-transcript  /home/aarvay/serp/s640-30v-2500.txt
/report/surface-integrals/vertex-avg terminal_c , electric-potential n
/report/surface-integrals/integral terminal_c , y-current-flux-density n
/report/species-mass-flow
/file/stop-transcript
/solve/iterate 500
/file/start-transcript  /home/aarvay/serp/s640-30v-3000.txt
/report/surface-integrals/vertex-avg terminal_c , electric-potential n
/report/surface-integrals/integral terminal_c , y-current-flux-density n
/report/species-mass-flow
/file/write-case-data /home/aarvay/serp/s640-30v-3000.cas
/file/stop-transcript
exit y
```

CONTENTS OF s640-70v-1500.txt

```
> /report/surface-integrals/vertex-avg terminal_c , electric-potential n

Average of Surface Vertex Values
           Electric Potential                (v)
------------------------------- --------------------
                  terminal_c          0.70000017

> /report/surface-integrals/integral terminal_c , y-current-flux-density n

                    Integral
        Y Current Flux Density          (a/m2)(m2)
------------------------------- --------------------
                  terminal_c          -8.3995056

> /report/species-mass-flow
zone 47 (catalyst_c): (0 0 0)
zone 54 (channel_a): (0 0 0)
zone 55 (channel_c): (0 0 0)
zone 23 (current_c-shadow): (0 0 0)
zone 52 (gdl_a): (0 0 0)
zone 5 (gdl_a:005): (0 0 0)
zone 53 (gdl_c): (0 0 0)
zone 15 (gdl_c:007-shadow): (0 0 0)
zone 61 (massflow_inlet_a): (2.5e-06 0 2.5e-06)
zone 58 (massflow_inlet_c): (0 1.5708e-05 1.32e-05)
zone 51 (membrane): (0 0 0)
zone 59 (outlet_a): (-2.39904e-06 -2.1026266e-09 -1.1919924e-06)
zone 60 (outlet_c): (-6.2110525e-09 -1.4959543e-05 -4.0301031e-06)

net species-mass-flow: (9.4748948e-08 7.4635437e-07 1.0477904e-05)

> /file/stop-transcript
```

Sample queue file for Linux based job script.

```
#!/bin/bash
#PBS -l nodes=4
#PBS -j oe
#PBS -o $PBS_JOBID.output
#PBS -l walltime=24:00:00

cd $PBS_O_WORKDIR

use ansysfluent
fluent -g 3ddp -t 4 -i s640k.jou
```

```
        exit
```

APPENDIX D

DIVERGENT SOLUTION CONSOLE OUTPUT

```
/packages/ansys_inc/v121/fluent/fluent12.1.2/bin/fluent -r12.1.2 -g 3ddp
-t 4 -i s784k-8-fix.jou
/packages/ansys_inc/v121/fluent/fluent12.1.2/cortex/lnamd64/cortex.12.1.2
-f fluent -g -i s784k-8-fix.jou -newcx (fluent "3ddp -pshmem  -host -
r12.1.2 -t4 -mpi=hp -path/packages/ansys_inc/v121/fluent")
Loading "/packages/ansys_inc/v121/fluent/fluent12.1.2/lib/fluent.dmp.114-
64"
Done.
/packages/ansys_inc/v121/fluent/fluent12.1.2/bin/fluent -r12.1.2 3ddp -
pshmem -host -t4 -mpi=hp -path/packages/ansys_inc/v121/fluent -cx s11-
7.local:43829:59436
Starting
/packages/ansys_inc/v121/fluent/fluent12.1.2/lnamd64/3ddp_host/fluent.12.
1.2 host -cx s11-7.local:43829:59436 "(list (rpsetvar (QUOTE
parallel/function) "fluent 3ddp -node -r12.1.2 -t4 -pshmem -mpi=hp ")
(rpsetvar (QUOTE parallel/rhost) "") (rpsetvar (QUOTE parallel/ruser) "")
(rpsetvar (QUOTE parallel/nprocs_string) "4") (rpsetvar (QUOTE
parallel/auto-spawn?) #t) (rpsetvar (QUOTE parallel/trace-level) 0)
(rpsetvar (QUOTE parallel/remote-shell) 0) (rpsetvar (QUOTE
parallel/path) "/packages/ansys_inc/v121/fluent") (rpsetvar (QUOTE
parallel/hostsfile) "") )"

     Welcome to ANSYS FLUENT 12.1.2

     Copyright 2009 ANSYS Inc.
     All Rights Reserved. Unauthorized use, distribution or duplication
is prohibited. ANSYS and FLUENT are trademarks or registered trademarks
of ANSYS, Inc. or its subsidiaries in the United States or other
countries.

Loading
"/packages/ansys_inc/v121/fluent/fluent12.1.2/lib/flprim.dmp.1119-64"
Done.


     --------------------------------------------------------------
     This is a version of FLUENT intended for use by students,
     academic staff, and faculty.  Usage of studentFLUENT is
     limited to the terms and conditions specified in the
     Clickwrap Software License Agreement for studentFLUENT.
     --------------------------------------------------------------
Host spawning Node 0 on machine "s11-7.local" (unix).
/packages/ansys_inc/v121/fluent/fluent12.1.2/bin/fluent -r12.1.2 3ddp -
node -t4 -pshmem -mpi=hp -mport 10.0.11.126:10.0.11.126:50881:0
Starting
/packages/ansys_inc/v121/fluent/fluent12.1.2/multiport/mpi/lnamd64/hp/bin
/mpirun -np 4
/packages/ansys_inc/v121/fluent/fluent12.1.2/lnamd64/3ddp_node/fluent_mpi
.12.1.2 node -mpiw hp -pic shmem -mport 10.0.11.126:10.0.11.126:50881:0
HP-MPI licensed for FLUENT.

-------------------------------------------------------------------------
-----
ID     Comm.   Hostname        O.S.        PID     Mach ID HW ID   Name
-------------------------------------------------------------------------
-----
host   net     s11-7.local     Linux-64    13629   0       5       Fluent
Host
n3     hp      s11-7.local     Linux-64    13773   0       3       Fluent
Node
n2     hp      s11-7.local     Linux-64    13772   0       2       Fluent
Node
n1     hp      s11-7.local     Linux-64    13771   0       1       Fluent
Node
n0*    hp      s11-7.local     Linux-64    13770   0       0       Fluent
Node

Selected system interconnect: shared-memory
```

```
-------------------------------------------------------------------------
-----

Cleanup script file is /home/aarvay/cleanup-fluent-s11-7.local-13629.sh

> ;
;
/file/read-case-data /home/aarvay/serp/s784k-4-90v-1500.cas

Multicore processors detected. Processor affinity set!

CPU clock is currently running lower than peak (2000/2667 MHz)
Check CPU frequency governors if any performance issues!


Reading "/home/aarvay/serp/s784k-4-90v-1500.cas"...

.........................................................................
...........................................
  ...............

   34125 hexahedral cells, zone 29, binary.
   34125 hexahedral cells, zone 30, binary.
   68250 hexahedral cells, zone 31, binary.
   33928 hexahedral cells, zone 32, binary.
  239072 hexahedral cells, zone 33, binary.
  238710 hexahedral cells, zone 34, binary.
   34290 hexahedral cells, zone 35, binary.
   68250 hexahedral cells, zone 36, binary.
   34125 hexahedral cells, zone 37, binary.
   34125 cell partition ids, zone 29, 4 partitions, binary.
   34125 cell partition ids, zone 30, 4 partitions, binary.
   68250 cell partition ids, zone 31, 4 partitions, binary.
   33928 cell partition ids, zone 32, 4 partitions, binary.
  239072 cell partition ids, zone 33, 4 partitions, binary.
  238710 cell partition ids, zone 34, 4 partitions, binary.
   34290 cell partition ids, zone 35, 4 partitions, binary.
   68250 cell partition ids, zone 36, 4 partitions, binary.
   34125 cell partition ids, zone 37, 4 partitions, binary.
   34125 quadrilateral interior faces, zone  1, binary.
   34125 quadrilateral interior faces, zone  2, binary.
   34125 quadrilateral interior faces, zone  3, binary.
    8057 quadrilateral interior faces, zone  6, binary.
   34125 quadrilateral interior faces, zone  8, binary.
   67880 quadrilateral interior faces, zone 38, binary.
   67880 quadrilateral interior faces, zone 39, binary.
  169885 quadrilateral interior faces, zone 40, binary.
   72472 quadrilateral interior faces, zone 41, binary.
  667795 quadrilateral interior faces, zone 42, binary.
  666709 quadrilateral interior faces, zone 43, binary.
   73377 quadrilateral interior faces, zone 44, binary.
  169885 quadrilateral interior faces, zone 45, binary.
   67880 quadrilateral interior faces, zone 46, binary.
    8569 quadrilateral interior faces, zone 10116, binary.
    9088 quadrilateral interior faces, zone 10117, binary.
    8395 quadrilateral interior faces, zone 10118, binary.
     350 quadrilateral wall faces, zone  4, binary.
   16980 quadrilateral wall faces, zone  5, binary.
   17161 quadrilateral wall faces, zone  7, binary.
   17161 quadrilateral wall faces, zone 15, binary.
   16980 quadrilateral wall faces, zone 17, binary.
   41829 quadrilateral wall faces, zone 24, binary.
   41648 quadrilateral wall faces, zone 25, binary.
     740 quadrilateral wall faces, zone 47, binary.
     390 quadrilateral wall faces, zone 48, binary.
    5908 quadrilateral wall faces, zone 49, binary.
    5908 quadrilateral wall faces, zone 50, binary.
     740 quadrilateral wall faces, zone 51, binary.
    1480 quadrilateral wall faces, zone 52, binary.
```

```
     1480 quadrilateral wall faces, zone 53, binary.
    41829 quadrilateral wall faces, zone 54, binary.
    41648 quadrilateral wall faces, zone 55, binary.
    34125 quadrilateral wall faces, zone 56, binary.
    34125 quadrilateral wall faces, zone 57, binary.
        6 quadrilateral pressure-outlet faces, zone 10104, binary.
        6 quadrilateral pressure-outlet faces, zone 10107, binary.
        6 quadrilateral mass-flow-inlet faces, zone 58, binary.
        6 quadrilateral mass-flow-inlet faces, zone 61, binary.
    16980 shadow face pairs, binary.
    17161 shadow face pairs, binary.
    41829 shadow face pairs, binary.
    41648 shadow face pairs, binary.
   920096 nodes, binary.
   920096 node flags, binary.
Fast-loading
"/packages/ansys_inc/v121/fluent/fluent12.1.2/addons/fuelcells/lib/addon.
bin"
Done.

Opening library
"/packages/ansys_inc/v121/fluent/fluent12.1.2/addons/fuelcells"...
Library
"/packages/ansys_inc/v121/fluent/fluent12.1.2/addons/fuelcells/lnamd64/3d
dp_host/libudf.so" opened
       resist_chan
       density
       diff
       cond
       uds_flux
       source_s
       source_m
       source_h2
       source_o2
       source_h2o
       source_mass
       source_lambda
       source_energy
       source_liquid
       lam_bc
       initialize
       adjust
       unsteady
       write_sec
       read_sec
       pem_lib_load
       set_new_stack_voltage
       set_new_stack_current
       get_old_stack_voltage
       get_old_stack_current
       list_pemfc_udf
       report_current
       aniso_econd
       UDFUtilityFunctions:
         fl_uds_contact_resistance

UDF Autorun:
pem_lib_load
  Renaming 15 UDMs
  Renaming  4 UDSs

Done.
    UDM  0 is named X Current Flux Density [elec-current-flux]
    UDM  1 is named Y Current Flux Density [elec-current-flux]
    UDM  2 is named Z Current Flux Density [elec-current-flux]
    UDM  3 is named Current Flux Density Magnitude [elec-current-flux]
    UDM  4 is named Ohmic Heat Source [heat-generation-rate]
    UDM  5 is named Reaction Heat Source [heat-generation-rate]
    UDM  6 is named Overpotential [elec-potential]
    UDM  7 is named Phase Change Source (PEM) [mass-transfer-rate]
    UDM  8 is named Osmotic Drag Coefficient (PEM) []
    UDM  9 is named Liquid Water Activity (PEM) []
    UDM 10 is named Membrane Water Content (PEM) []
    UDM 11 is named Protonic Conductivity [elec-conductivity]
    UDM 12 is named Back Diffusion Source(PEM) [mass-transfer-rate]
    UDM 13 is named Transfer Current [elec-current-per-unit-volume]
    UDM 14 is named Osmotic Drag Source (PEM) [mass-transfer-rate]
    UDS  0 is named Electric Potential [elec-potential]
    UDS  1 is named Protonic Potential [elec-potential]
    UDS  2 is named Water Saturation []
    UDS  3 is named Water Content []

Opening library
"/packages/ansys_inc/v121/fluent/fluent12.1.2/addons/fuelcells"...
Library
"/packages/ansys_inc/v121/fluent/fluent12.1.2/addons/fuelcells/lnamd64/3d
dp_node/libudf.so" opened    UDM  0 is named X Current Flux Density
[elec-current-flux]
    UDM  1 is named Y Current Flux Density [elec-current-flux]
    UDM  2 is named Z Current Flux Density [elec-current-flux]
    UDM  3 is named Current Flux Density Magnitude [elec-current-flux]
    UDM  4 is named Ohmic Heat Source [heat-generation-rate]
    UDM  5 is named Reaction Heat Source [heat-generation-rate]
    UDM  6 is named Overpotential [elec-potential]
    UDM  7 is named Phase Change Source (PEM) [mass-transfer-rate]
    UDM  8 is named Osmotic Drag Coefficient (PEM) []
    UDM  9 is named Liquid Water Activity (PEM) []
    UDM 10 is named Membrane Water Content (PEM) []
    UDM 11 is named Protonic Conductivity [elec-conductivity]
    UDM 12 is named Back Diffusion Source(PEM) [mass-transfer-rate]
    UDM 13 is named Transfer Current [elec-current-per-unit-volume]
    UDM 14 is named Osmotic Drag Source (PEM) [mass-transfer-rate]
    UDS  0 is named Electric Potential [elec-potential]
    UDS  1 is named Protonic Potential [elec-potential]
    UDS  2 is named Water Saturation []
    UDS  3 is named Water Content []

Opening library
"/packages/ansys_inc/v121/fluent/fluent12.1.2/addons/fuelcells"...
Library
"/packages/ansys_inc/v121/fluent/fluent12.1.2/addons/fuelcells/lnamd64/3d
dp_node/libudf.so" opened    UDM  0 is named X Current Flux Density
[elec-current-flux]
    UDM  1 is named Y Current Flux Density [elec-current-flux]
    UDM  2 is named Z Current Flux Density [elec-current-flux]
    UDM  3 is named Current Flux Density Magnitude [elec-current-flux]
    UDM  4 is named Ohmic Heat Source [heat-generation-rate]
    UDM  5 is named Reaction Heat Source [heat-generation-rate]
    UDM  6 is named Overpotential [elec-potential]
    UDM  7 is named Phase Change Source (PEM) [mass-transfer-rate]
    UDM  8 is named Osmotic Drag Coefficient (PEM) []
    UDM  9 is named Liquid Water Activity (PEM) []
    UDM 10 is named Membrane Water Content (PEM) []
    UDM 11 is named Protonic Conductivity [elec-conductivity]
    UDM 12 is named Back Diffusion Source(PEM) [mass-transfer-rate]
    UDM 13 is named Transfer Current [elec-current-per-unit-volume]
    UDM 14 is named Osmotic Drag Source (PEM) [mass-transfer-rate]
    UDS  0 is named Electric Potential [elec-potential]
    UDS  1 is named Protonic Potential [elec-potential]
    UDS  2 is named Water Saturation []
    UDS  3 is named Water Content []

Opening library
"/packages/ansys_inc/v121/fluent/fluent12.1.2/addons/fuelcells"...
Library
"/packages/ansys_inc/v121/fluent/fluent12.1.2/addons/fuelcells/lnamd64/3d
dp_node/libudf.so" opened    UDM  0 is named X Current Flux Density
[elec-current-flux]
```

```
    UDM  1 is named Y Current Flux Density [elec-current-flux]
    UDM  2 is named Z Current Flux Density [elec-current-flux]
    UDM  3 is named Current Flux Density Magnitude [elec-current-flux]
    UDM  4 is named Ohmic Heat Source [heat-generation-rate]
    UDM  5 is named Reaction Heat Source [heat-generation-rate]
    UDM  6 is named Overpotential [elec-potential]
    UDM  7 is named Phase Change Source (PEM) [mass-transfer-rate]
    UDM  8 is named Osmotic Drag Coefficient (PEM) []
    UDM  9 is named Liquid Water Activity (PEM) []
    UDM 10 is named Membrane Water Content (PEM) []
    UDM 11 is named Protonic Conductivity [elec-conductivity]
    UDM 12 is named Back Diffusion Source(PEM) [mass-transfer-rate]
    UDM 13 is named Transfer Current [elec-current-per-unit-volume]
    UDM 14 is named Osmotic Drag Source (PEM) [mass-transfer-rate]
    UDS  0 is named Electric Potential [elec-potential]
    UDS  1 is named Protonic Potential [elec-potential]
    UDS  2 is named Water Saturation []
    UDS  3 is named Water Content []

Opening library
"/packages/ansys_inc/v121/fluent/fluent12.1.2/addons/fuelcells"...
Library
"/packages/ansys_inc/v121/fluent/fluent12.1.2/addons/fuelcells/lnamd64/3d
dp_node/libudf.so" opened    UDM  0 is named X Current Flux Density
[elec-current-flux]
    UDM  1 is named Y Current Flux Density [elec-current-flux]
    UDM  2 is named Z Current Flux Density [elec-current-flux]
    UDM  3 is named Current Flux Density Magnitude [elec-current-flux]
    UDM  4 is named Ohmic Heat Source [heat-generation-rate]
    UDM  5 is named Reaction Heat Source [heat-generation-rate]
    UDM  6 is named Overpotential [elec-potential]
    UDM  7 is named Phase Change Source (PEM) [mass-transfer-rate]
    UDM  8 is named Osmotic Drag Coefficient (PEM) []
    UDM  9 is named Liquid Water Activity (PEM) []
    UDM 10 is named Membrane Water Content (PEM) []
    UDM 11 is named Protonic Conductivity [elec-conductivity]
    UDM 12 is named Back Diffusion Source(PEM) [mass-transfer-rate]
    UDM 13 is named Transfer Current [elec-current-per-unit-volume]
    UDM 14 is named Osmotic Drag Source (PEM) [mass-transfer-rate]
    UDS  0 is named Electric Potential [elec-potential]
    UDS  1 is named Protonic Potential [elec-potential]
    UDS  2 is named Water Saturation []
    UDS  3 is named Water Content []

Building...
     mesh
       distributing mesh
             pairs....,
             parts....,
             faces....,
             nodes....,
             shadow nodes....,
             cells....,
     materials,
     interface,
     domains,
       mixture
     zones,
       gdl_c:007
       gdl_a:005
       membrane:004
       pressure_outlet_c
       massflow_inlet_a
       terminal_c
       pressure_outlet_a
       massflow_inlet_c
       terminal_a
       channel_c
       channel_a
       gdl_c
       gdl_a
       membrane
       current_c
       current_a
       catalyst_a
       catalyst_c
       int_current_a_vol
       int_channel_a_vol
       int_gdl_a_vol
       int_catalyst_a_vol
       int_membrane_vol
       int_catalyst_c_vol
       int_gdl_c_vol
       int_channel_c_vol
       int_current_c_vol
       gdl_c:007-shadow
       gdl_a:005-shadow
       channel_a-shadow
       channel_c-shadow
       gdl_c:008
       gdl_c:009
       gdl_a:006
       channel_c:011
       channel_a:010
       catalyst_c:001
       catalyst_a:003
       catalyst_a:002
       current_a_vol
       channel_a_vol
       gdl_a_vol
       catalyst_a_vol
       membrane_vol
       catalyst_c_vol
       gdl_c_vol
       channel_c_vol
       current_c_vol
     parallel,
Done.

Reading "/home/aarvay/serp/s784k-4-90v-1500.dat"...
Parallel variables...
Done.

> /parallel/partition/method cartesian-axes 8

>> Dividing domain into 8 partitions using Cartesian Axes.
....... 7 bisections.
   Smoothing partition boundaries...
   Time = 0.608905 seconds.
   Done.

>> 8 Stored Partitions:
   ----------------------------------------------------------------------
   Collective Partition Statistics:        Minimum   Maximum   Total
   ----------------------------------------------------------------------
   Cell count                              97962     98259     784875
   Mean cell count deviation               -0.2%     0.2%
   Partition boundary cell count           3106      5452      33764
   Partition boundary cell count ratio     3.2%      5.6%      4.3%

   Face count                              316758    318019    2514878
   Mean face count deviation               -0.1%     0.3%
   Partition boundary face count           4026      7047      22309
   Partition boundary face count ratio     1.3%      2.2%      0.9%

   Cell weights                            1e+05     1e+05     8e+05
   Mean cell weight deviation              -0.2%     0.2%
   Partition neighbor count                2        5
   ------------------------------------------------------------------------
   Partition Method                        Cartesian Axes
   Stored Partition Count                  8
Done.

> ;
;
/solve/set/under-relaxation species-0 .95

> /solve/set/under-relaxation species-1 .95

> /solve/set/under-relaxation species-2 .95

> /solve/set/under-relaxation momentum invalid command [momentum]
0.7

> ;
;
/define/boundary-conditions/wall (channel_c-shadow channel_a-shadow
gdl_a:005-shadow gdl_c:007-shadow catalyst_c catalyst_a current_a
current_c membrane gdl_a gdl_c channel_a channel_c terminal_a terminal_c
membrane:004 gdl_a:005 gdl_c:007)
terminal_c , , , , , , , , , , , , .80 , ,

> /solve/iterate 500
  iter continuity x-velocity y-velocity z-velocity     energy         h2
o2        h2o      uds-0      uds-1      uds-2      uds-3     time/iter
  3000 5.9314e-05 5.1371e-07 1.8620e-07 1.8943e-06 1.2067e-11 2.9065e-06
5.7161e-08 1.0296e-06 1.6350e-09 4.0482e-04 2.5804e-06 2.7930e-06
0:00:00  500
       I_anode = 0.000000 (A/cm^2) ... I_cathode = 0.000000 (A/cm^2) ...
dI_mea = 0.000000 (A/cm^2) ... dI_stk = 0.000000 (A/cm^2)
       V_cell  = 0.800000 (V)      ... V_open    = 0.980000 (V)
       I_anode = 0.004206 (A/cm^2) ... I_cathode = 0.004205 (A/cm^2) ...
dI_mea = 0.000001 (A/cm^2) ... dI_stk = 0.000001 (A/cm^2)
       I_anode = 0.004206 (A/cm^2) ... I_cathode = 0.004205 (A/cm^2) ...
dI_mea = 0.000001 (A/cm^2) ... dI_stk = 0.000001 (A/cm^2)
       I_anode = 0.004206 (A/cm^2) ... I_cathode = 0.004205 (A/cm^2) ...
dI_mea = 0.000001 (A/cm^2) ... dI_stk = 0.000001 (A/cm^2)
       I_anode = 0.004206 (A/cm^2) ... I_cathode = 0.004205 (A/cm^2) ...
dI_mea = 0.000001 (A/cm^2) ... dI_stk = 0.000001 (A/cm^2)
  3001 6.1456e-05 4.9499e-07 1.7947e-07 1.8882e-06 1.2054e-11 2.9064e-06
5.7110e-08 1.0292e-06 5.8469e-04 1.4112e-02 2.5784e-06 5.4691e-06
0:57:19  499
       I_anode = 0.000000 (A/cm^2) ... I_cathode = 0.000000 (A/cm^2) ...
dI_mea = 0.000000 (A/cm^2) ... dI_stk = 0.000000 (A/cm^2)
       V_cell  = 0.800000 (V)      ... V_open    = 0.980000 (V)
       I_anode = 0.076654 (A/cm^2) ... I_cathode = 0.085652 (A/cm^2) ...
dI_mea = -0.008998 (A/cm^2) ... dI_stk = -0.008998 (A/cm^2)
       I_anode = 0.076654 (A/cm^2) ... I_cathode = 0.085652 (A/cm^2) ...
dI_mea = -0.008998 (A/cm^2) ... dI_stk = -0.008998 (A/cm^2)
       I_anode = 0.076654 (A/cm^2) ... I_cathode = 0.085652 (A/cm^2) ...
dI_mea = -0.008998 (A/cm^2) ... dI_stk = -0.008998 (A/cm^2)
       I_anode = 0.076654 (A/cm^2) ... I_cathode = 0.085652 (A/cm^2) ...
dI_mea = -0.008998 (A/cm^2) ... dI_stk = -0.008998 (A/cm^2)
  3002 2.8858e-04 5.1791e-07 1.8799e-07 1.8869e-06 1.7102e-08 2.9347e-06
3.5431e-06 7.1008e-06 5.0285e-05 1.0376e-02 2.6460e-06 2.6813e-06
1:01:01  498
       I_anode = 0.000000 (A/cm^2) ... I_cathode = 0.000000 (A/cm^2) ...
dI_mea = 0.000000 (A/cm^2) ... dI_stk = 0.000000 (A/cm^2)
       V_cell  = 0.800000 (V)      ... V_open    = 0.980000 (V)
       I_anode = 0.078571 (A/cm^2) ... I_cathode = 0.079666 (A/cm^2) ...
dI_mea = -0.001095 (A/cm^2) ... dI_stk = -0.001095 (A/cm^2)
       I_anode = 0.078571 (A/cm^2) ... I_cathode = 0.079666 (A/cm^2) ...
dI_mea = -0.001095 (A/cm^2) ... dI_stk = -0.001095 (A/cm^2)
       I_anode = 0.078571 (A/cm^2) ... I_cathode = 0.079666 (A/cm^2) ...
dI_mea = -0.001095 (A/cm^2) ... dI_stk = -0.001095 (A/cm^2)
```

```
       I_anode = 0.078571 (A/cm^2) ... I_cathode = 0.079666 (A/cm^2) ...
dI_mea = -0.001095 (A/cm^2) ... dI_stk = -0.001095 (A/cm^2)
  3003 1.8658e-03 1.4042e-04 4.6560e-05 1.5618e-05 8.8798e-05 4.2928e-06
3.5311e-06 6.7524e-06 1.5620e-05 1.2548e-02 2.6679e-06 6.3425e-05
1:04:06  497
       I_anode = 0.000000 (A/cm^2) ... I_cathode = 0.000000 (A/cm^2) ...
dI_mea = 0.000000 (A/cm^2) ... dI_stk = 0.000000 (A/cm^2)
 temperature limited to 1.000000e+00 in 129 cells on zone 29 in domain 1
  temperature limited to 1.000000e+00 in 204 cells on zone 30 in domain 1
  temperature limited to 1.000000e+00 in 407 cells on zone 31 in domain 1
  temperature limited to 1.000000e+00 in 85 cells on zone 33 in domain 1
  temperature limited to 1.000000e+00 in 188 cells on zone 36 in domain 1
  temperature limited to 1.000000e+00 in 178 cells on zone 37 in domain 1
        V_cell  = 0.800000 (V)      ... V_open    = 0.980000 (V)
       I_anode = 0.048205 (A/cm^2) ... I_cathode = 0.044731 (A/cm^2) ...
dI_mea = 0.003474 (A/cm^2) ... dI_stk = 0.003474 (A/cm^2)
       I_anode = 0.048205 (A/cm^2) ... I_cathode = 0.044731 (A/cm^2) ...
dI_mea = 0.003474 (A/cm^2) ... dI_stk = 0.003474 (A/cm^2)
       I_anode = 0.048205 (A/cm^2) ... I_cathode = 0.044731 (A/cm^2) ...
dI_mea = 0.003474 (A/cm^2) ... dI_stk = 0.003474 (A/cm^2)
       I_anode = 0.048205 (A/cm^2) ... I_cathode = 0.044731 (A/cm^2) ...
dI_mea = 0.003474 (A/cm^2) ... dI_stk = 0.003474 (A/cm^2)
  3004 8.3477e-02 1.8386e-04 4.7126e-05 2.0558e-05 4.8775e-05 5.0007e-06
6.9301e-06 6.0737e-03 1.3812e-04 3.0336e-02 4.6554e-04 9.9060e-03
1:03:05  496
       I_anode = 0.000000 (A/cm^2) ... I_cathode = 0.000000 (A/cm^2) ...
dI_mea = 0.000000 (A/cm^2) ... dI_stk = 0.000000 (A/cm^2)
 temperature limited to 5.000000e+03 in 34125 cells on zone 29 in domain
1
  temperature limited to 5.000000e+03 in 34125 cells on zone 30 in domain
1
  temperature limited to 5.000000e+03 in 68250 cells on zone 31 in domain
1
  temperature limited to 5.000000e+03 in 33928 cells on zone 32 in domain
1
  temperature limited to 5.000000e+03 in 239072 cells on zone 33 in
domain 1
  temperature limited to 5.000000e+03 in 238710 cells on zone 34 in
domain 1
  temperature limited to 5.000000e+03 in 34289 cells on zone 35 in domain
1
  temperature limited to 5.000000e+03 in 68250 cells on zone 36 in domain
1
  temperature limited to 5.000000e+03 in 34125 cells on zone 37 in domain
1

 absolute pressure limited to 1.000000e+00 in 2103 cells on zone 29

 absolute pressure limited to 5.000000e+10 in 2135 cells on zone 30

 absolute pressure limited to 5.000000e+10 in 4287 cells on zone 31

 absolute pressure limited to 5.000000e+10 in 1832 cells on zone 32

 absolute pressure limited to 1.000000e+00 in 2296 cells on zone 35

 absolute pressure limited to 1.000000e+00 in 4216 cells on zone 36
       V_cell  = 0.800000 (V)      ... V_open    = 0.980000 (V)
       I_anode = 0.048039 (A/cm^2) ... I_cathode = 33016236.366049
(A/cm^2) ... dI_mea = -33016236.318010 (A/cm^2) ... dI_stk = -
33016236.318010 (A/cm^2)
       I_anode = 0.048039 (A/cm^2) ... I_cathode = 33016236.366049
(A/cm^2) ... dI_mea = -33016236.318010 (A/cm^2) ... dI_stk = -
33016236.318010 (A/cm^2)
       I_anode = 0.048039 (A/cm^2) ... I_cathode = 33016236.366049
(A/cm^2) ... dI_mea = -33016236.318010 (A/cm^2) ... dI_stk = -
33016236.318010 (A/cm^2)
```

```
       I_anode = 0.048039 (A/cm^2) ... I_cathode = 33016236.366049
(A/cm^2) ... dI_mea = -33016236.318010 (A/cm^2) ... dI_stk = -
33016236.318010 (A/cm^2)
  3005 2.9350e+04 1.3768e-02 2.6790e-03 4.0510e-03 2.1942e+02 1.3316e-02
9.4951e-01 1.9928e+02 5.2086e-06 9.7835e-01 7.8361e-01 1.0299e-02
1:30:40  495
       I_anode = 0.000000 (A/cm^2) ... I_cathode = 0.000000 (A/cm^2) ...
dI_mea = 0.000000 (A/cm^2) ... dI_stk = 0.000000 (A/cm^2)
Warning: boundary Mach number exceeds maximum limit on mass-flow-inlet-61
for 3 faces
bounding Mach number to maximum limit =9.800000e-01

 reversed flow in 6 faces on pressure-outlet 10104.
# Divergence detected in AMG solver: temperature -> Increasing relaxation
sweeps!
 temperature limited to 1.000000e+00 in 33992 cells on zone 29 in domain
1
  temperature limited to 5.000000e+03 in 133 cells on zone 29 in domain 1
  temperature limited to 1.000000e+00 in 34125 cells on zone 30 in domain
1
  temperature limited to 1.000000e+00 in 68250 cells on zone 31 in domain
1
  temperature limited to 1.000000e+00 in 33928 cells on zone 32 in domain
1
  temperature limited to 1.000000e+00 in 239072 cells on zone 33 in
domain 1
  temperature limited to 1.000000e+00 in 238710 cells on zone 34 in
domain 1
  temperature limited to 1.000000e+00 in 34241 cells on zone 35 in domain
1
  temperature limited to 5.000000e+03 in 49 cells on zone 35 in domain 1
  temperature limited to 1.000000e+00 in 67967 cells on zone 36 in domain
1
  temperature limited to 5.000000e+03 in 283 cells on zone 36 in domain 1
  temperature limited to 1.000000e+00 in 34125 cells on zone 37 in domain
1

 absolute pressure limited to 1.000000e+00 in 589 cells on zone 29

 absolute pressure limited to 1.000000e+00 in 56 cells on zone 30

 absolute pressure limited to 1.000000e+00 in 74 cells on zone 31

 absolute pressure limited to 5.000000e+10 in 10 cells on zone 31

 absolute pressure limited to 1.000000e+00 in 208 cells on zone 32

 absolute pressure limited to 5.000000e+10 in 507 cells on zone 32

 absolute pressure limited to 1.000000e+00 in 2284 cells on zone 35

 absolute pressure limited to 1.000000e+00 in 4200 cells on zone 36
       V_cell  = 0.800000 (V)      ... V_open    = 0.980000 (V)
       I_anode = 0.000000 (A/cm^2) ... I_cathode = 0.000000 (A/cm^2) ...
dI_mea = -0.000000 (A/cm^2) ... dI_stk = -0.000000 (A/cm^2)

Divergence detected in AMG solver: species-0
Divergence detected in AMG solver: species-1
Divergence detected in AMG solver: species-2
Divergence detected in AMG solver: temperature       I_anode = 0.000000
(A/cm^2) ... I_cathode = 0.000000 (A/cm^2) ... dI_mea = -0.000000
(A/cm^2) ... dI_stk = -0.000000 (A/cm^2)

Divergence detected in AMG solver: species-0
Divergence detected in AMG solver: species-1
Divergence detected in AMG solver: species-2
Divergence detected in AMG solver: temperature       I_anode = 0.000000
(A/cm^2) ... I_cathode = 0.000000 (A/cm^2) ... dI_mea = -0.000000
(A/cm^2) ... dI_stk = -0.000000 (A/cm^2)
```

```
Divergence detected in AMG solver: species-0
Divergence detected in AMG solver: species-1
Divergence detected in AMG solver: species-2
Divergence detected in AMG solver: temperature       I_anode = 0.000000
(A/cm^2) ... I_cathode = 0.000000 (A/cm^2) ... dI_mea = -0.000000
(A/cm^2) ... dI_stk = -0.000000 (A/cm^2)

Divergence detected in AMG solver: species-0
Divergence detected in AMG solver: species-1
Divergence detected in AMG solver: species-2
Divergence detected in AMG solver: temperature
Primitive Error at Node 1: floating point exception

Primitive Error at Node 2: floating point exception

Primitive Error at Node 3: floating point exception

Primitive Error at Node 0: floating point exception

Error: floating point exception
Error Object: #f

Warning: An error or interrupt occurred while reading the journal file.
Some commands may not have been completed.

> Halting due to end of file on input.
```