

3D Simulation of Quench

In An 11T Dipole Magnet

Charles R. Orozco x 15700N

8/9/2012

As with anything, there is always room for improvement. In this manner, CERN wishes to upgrade the LHC by adding collimators to increase the luminosity. To do so, some room must be made by replacing existing Niobium-Titanium dipole magnets with Niobium-Tin dipole magnets. But first, we must understand the consequences of working with a different material to build superconducting accelerator magnets. One of these consequences is that the price of a quench could be much high (i.e. the total destruction of the coil). This paper details how a previously written program has been retooled to give the most accurate simulation of quench in a Niobium-Tin dipole magnet.

Background

On 30 March 2010, the LHC recorded the first 7TeV proton-proton collisions. Since then, they have increased this to 4TeV per beam, for a total of 8TeV. With a design capability of 14TeV, it will be years before the LHC ever needs a full scale energy upgrade. However, at the same time, CERN wishes to upgrade the luminosity in the LHC to increase the number of collisions, and ultimately, the number of rare events they can see. To do this, they must install devices known as collimators. These reduce the diameter of the beam, eliminating scattered particles, and protecting all magnets from radiation damage. This would work well; however, all of the space along the beam line is already occupied by magnets. Therefore, to accomplish the goal of installing collimators, some space must be made free. To do this, CERN will replace certain NbTi magnets with higher field Nb₃Sn magnets.

Materials

The most common material presently in use to make superconducting accelerator magnets is niobium-titanium (NbTi). This material is advantageous to work with because it is ductile, meaning it is easy to manufacture any desired shape. However, there is a drawback. NbTi can only reach a magnetic field of 8.3T which means that as the physics community tries to reach new heights in the energy of collisions, new materials must be developed to allow for higher field magnets.

To achieve this, a different material is being used that can safely operate up to 11T, possibly even 13T. This material is niobium-tin (Nb₃Sn) and it is expected to be the workhorse of tomorrow's superconducting accelerator magnets. As with everything, though, there is a drawback. Nb₃Sn is incredibly brittle, making it much harder to manufacture.

The Quench Problem

One problem inherent to superconducting magnets is quench. This phenomenon occurs when a small part of the superconducting material enters into the normal state. There are three routes by which this

can happen. All of these paths involve one of three critical values. These critical values are determined by a material-specific plot (Figure 1) which defines the relationship between the current density, the local magnetic B field, and the temperature. If at any time and any place, any of these values exceed the critical value, that portion of the magnet will be in a normal state and the magnet will quench.

Quenching in high field superconducting magnets is a problem because when a small part of the magnet enters the normal state, the high current impacts that small part and generates a large amount of heat. This creates a large temperature gradient which can be destructive in two ways. First, the localized heat build-up will spread to the surrounding areas, causing more of the coil to quench, and even more heat to be generated. This heat generation, if unchecked, could heat the coil locally to dangerous temperatures and possibly melt the coil of the magnet. The other method of destruction is through thermal expansion. If a small part of the magnet expands beyond the space available, the epoxy holding the cable of the coil together could crack and the magnetic forces at work could tear the magnet apart. Given that Nb₃Sn is a brittle material, this phenomenon could render the coil of the magnet irreparable.

Introduction

In today's NbTi superconducting accelerator magnets, heaters are installed on the magnets to heat them up in the event of a quench. To avoid the problem of too much current dissipating in a small part of the magnet coil, heating the whole magnet coil with the total stored energy is allowed, resulting in a much reduced temperature overall. This method has proven effective in the low-field NbTi magnets. The question here is will the same method work to prevent the destruction of high-field Nb₃Sn magnets?

To answer this question, I was tasked with modifying an old program to get the necessary thermal data. Last year I worked on this problem using a 2D model. This year we wanted to solve this problem with a 3D model. The original 3D ANSYS program, written by Prof. Masayoshi Wake (KEK), examined quench in

the 11T dipole magnet by modeling a quarter of the whole magnet assembly (Figure 2). It should be noted here that when I refer to the whole magnet assembly, I am referring only to the straight section of the magnet, as the end sections were too time consuming to accurately model. The main drawback to this program was that it primarily examined how a quench propagates along a turn and through the insulation from turn-to-turn. These are important modes, but the program ignores how a quench propagates around the coil. With that in mind, I expanded the program to include all four quadrants of the geometry (Figure 3).

To update the program, I made changes that can be summarized in three sections. First, there were changes that needed to be made to accommodate the full geometry. These changes do not add any functionality; they simply allow the program to work as it did before, albeit with four times the geometry. Second, I added functionality which allowed the heat from the quench to propagate around the coil, creating a more realistic simulation. The third set are miscellaneous improvements to the code such as upgraded data output, more parameterization, etc.

How the Program Works

The program works by first creating an array ("tt1") which includes a row for each node in the cable containing the node number, the temperature of the node, and the magnetic field strength at the node. It should be noted that the magnetic field strength values come from a 2D analysis using the cross section of the quarter 3D geometry; the magnetic field values are assumed not to change in the z direction. The magnetic field values are stored in an array ("tt2d") which contains the node number, the x and y coordinates of the node, and the magnetic B field strength. The program also creates a second array ("tt2") which includes a row for each node not in the cable containing the node number and the temperature; then the program enters the iterative phase. Once in the iterative phase (a do loop), the program applies the quench energy, the heater power, and the joule heating as needed; applies the

temperature at each node as an initial condition; then solves the nodal equations for the temperature distribution after the next time step.

After the solution completes, the program goes through each node in the conductors and retrieves the temperature of the node from the solution data. If that temperature exceeds the critical temperature, the node is assumed to have quenched. Also using this temperature, the program finds the new critical magnetic flux density and compares it the new magnetic flux density at the node (equation 1).

$$B_{cT} = B_{c0} \left(1 - \left(\frac{T}{T_{c0}} \right)^2 \right) \quad (1)$$

The new magnetic flux density at the node is calculated as a fraction of the previous magnetic flux density equal to the current density/initial critical current density. If the new magnetic flux density is greater than the new critical magnetic flux density, then the node is assumed to have quenched. The program then checks if the new current density exceeds the new critical current density (equation 2).

$$J_c = J_{c0} \frac{\left(\left(\frac{B_{cT} - B}{B_0} \right)^2 + 1 \right)^{1/2} - 1}{\left(\left(\frac{B_{c0} - B}{B_0} \right)^2 + 1 \right)^{1/2} - 1} \quad (2)$$

If it does, then the node is assumed to have quenched. In any case, if a node quenches, the MITs method (described below) is used to find the temperature of the node as if it had undergone joule heating in the previous time step. This new temperature is recorded into the array tt1. If a node does not quench, the temperature queried from the solution data is then recorded into the array tt1. Once all of the nodes in the conductors have been updated, the program then gets the temperature of every

node not in the conductors and updates the information in the array tt2. Here, no quench check is necessary as these nodes are not superconducting, nor do they have any current flowing through them.

After each node has its temperature updated, the program outputs four pictures and 21 data points.

Then the program updates the current and the current density. This marks the end of the iterative phase, so the program goes back to the top of the main do loop and does everything again for the next time step. This runs until a user-specified end time is reached. A flow chart showing how the program works is located in the appendix (Figure 4).

MIITs Method

The MIITs method is a way of accounting for Joule heating in resistive cable without needing to apply it as a load. This is advantageous as loads cause the ANSYS solve time to increase appreciably. The MIITs method works by relating material properties with current decay (equation 3).

$$\int_0^{\infty} I(t)^2 dt = S^2 \int_{T_q}^{T_{max}} \frac{C(T)}{\rho(T)} dt = MIIT \quad (3)$$

The material properties integral can be done prior to running any analyses. This integral can be performed at a variety of temperatures, giving rise to a table which relates temperatures with MIITs values. Within the analysis, the program uses the temperature of each node from the solution data to get a MIITs value for that temperature. The program then calculates a new MIITs value by adding the new current density squared times the time step to the old MIITs value (equation 4). This new MIITs value is run back against the table to get a new temperature, which is used as described above.

$$MIIT_{new} = MIIT_{old} + (J - J_c)^2 dt \quad (4)$$

Changes due to adding quadrants

Only one major change was made necessary due to my expansion of the 3D simulation to include all four quadrants of the dipole magnet assembly. This is to the array which contains the node number, temperature, and magnetic flux density of each node in the cable. This originally worked by using the x and y coordinates from tt2d to go through each node in the 2D geometry from the magnetic analysis and select all of the nodes behind it (in the z direction). For each of these nodes, the program would create a new row in tt1, setting the relevant quantities. This worked well for the one quadrant because the program knew the x and y coordinates of each node. However, since I did not retool the initial magnetic analysis to include all four quadrants, there is no coordinate data for the program to use for tt2d in the other three quadrants. This problem was solved by using the nature of the mirror command used to create the geometry. The mirror command preserves the magnitude of the coordinates for each point as well as the numbering scheme for all geometric quantities (important later). This feature allowed for four sections of code, each of which used appropriate negative signs on the coordinate data to build tt1 for each node in the cable. On a related note, tt2 needed no changes as it already worked on every node not associated with the cable material.

Coupling the Ends

Another issue with the previous version of the program was that in only modeling a quarter of the magnet assembly, it was unable to simulate quench propagation around the magnet, allowing for quench propagation only through the insulation. In an ideal scenario, the entire magnet assembly would be modeled. Unfortunately, due to a lack of a full CAD model of the 11T dipole magnet, only the straight section of the magnet could be modeled. This presents a problem, as the initial heat in the quench section needs to be able to spread to the neighboring turns, not only through the insulation, but also along the cable, around the end. Had we modeled the full 3D geometry, this would not have been a

problem because the magnet cable would have been modeled as one volume. However, because we lack the end pieces, some creative thinking was necessary.

ANSYS has a command, “cp”, which couples a set of nodes together. What this means is that what happens at one node is instantly copied to all the other nodes in the set. Using this functionality, I was able to couple the nodes at both ends of the magnet so as to simulate the winding of the magnet. To verify that these coupling commands were working, I shrunk the model to 9cm long with 3 3cm elements along the length. This drastically cut down the amount of time necessary for ANSYS to solve the nodal equations. Also, to make sure that both the upper and lower sections of the magnet were coupled correctly, I increased the initial quench energy by a factor of 10 and eliminated the dump resistor. Removing the dump resistor ensures that all of the current is dumped into the magnet.

This short geometry ran in less than 5 hours and the results demonstrated that, with a few minor errors, the coupling worked quite well. This result allowed me to ignore the end pieces while still allowing for a realistic simulation.

Miscellaneous Improvements

Changes to the Output

In this model, there are nearly half a million nodes. As such, the results file for each time step requires 1.5 GB of disk space. In this simulation, it takes 175 time steps to reach the end. Given that multiple runs are necessary to cover the gamut of scenarios, it is impractical to store all of the data generated. What this means for the simulation is, any data we wish to obtain must be output immediately after each time step, so the results file can be deleted.

In the previous version of the program, one picture and eleven data points were output at each time step. The picture was a standard quality jpeg showing the temperature distribution in the magnet. The data points were written to a text file. When using the quarter geometry, the overall size of the model was small enough that a standard quality image could capture the relevant data. However, when the model is quadrupled, a standard quality image does not have enough pixels to differentiate between the edges of the model and the surfaces of the model. Due to this quality issue, the program now outputs the image at 16 times the number of pixels. Another issue with the images was obfuscation of certain parts of the geometry by way of their orientation. To solve this problem, the program outputs four images, two looking from the left and two looking from the right (Figures 5-8). In two of the pictures (one each of left and right views), the upper layer of the coil is removed to enable the viewing of the lower layer.

In addition to modifying the picture output, the program now outputs 21 data points. The original eleven are still there; however, the program now also outputs the temperature of the center of the conductor which is directly below each of the heaters (eight more points) as well as two nodes in the quenching turn which are used to determine the quench speed. These can be made into a graph, an example of which is shown in Figure 9.

Changes to Quench Location Selection

Another change was necessary in the way the program initiated the quench. More specifically, the problem was in the way the program selected the nodes on which to apply the quench energy. In the previous version of the program, the selection technique required that the quench begin at either end of the magnet. This occurs because it was choosing nodes associated with the selected keypoints. Because of the way the 3D geometry is created, there are only keypoints at either end of the magnet. This is a problem for two reasons. For one, a quench can start anywhere, and the program is made

richer by allowing for such a possibility. For the other, given the constraint of not modeling the end sections and instead using coupling equations (see below), initiating a quench at the end would not easily allow verification that the coupling equations are working.

To solve this problem, I used the same selection technique described above for the array tt1. The program first selects all the nodes associated with the x and y coordinates of the two innermost corners of the chosen conductor. It then reselects from this set the nodes which correspond to a particular, user-specified, z value.

Results

All of the above changes resulted in a program which is capable of thermally simulating a quench in the entire magnet assembly. This program is capable of outputting the temperature at every location in the magnet at any time which is the necessary data for performing a thermal stress analysis to determine if the magnet will be destroyed in the event of a quench.

Conclusion

Since the first manufacture of a superconducting magnet, quench has been one of the most basic problems that engineers face in making superconducting magnets. As higher field magnets become a necessity, new materials must be developed to cope with higher demands. The solution to this problem is Niobium-Tin; however, with all new materials comes a cost. In this case, the cost is that the magnets are rendered essentially irreparable in the event of even minor damage. To prevent this, it is necessary to have a better understanding of how a quench occurs in a magnet and how well preventative measures work to keep the magnet intact should it unexpectedly quench. To that end, an old program has been modified to simulate these conditions thermally. This new program can be used to facilitate a stress analysis of the magnet and will ultimately determine the best way to protect the magnet.

Acknowledgements

At this point, I would like to acknowledge the contributions of certain individuals to my success in this summer's project. First, my supervisor Dr. Ryuji Yamada deserves special thanks for providing me with meaningful research to perform both this summer and last. Also, Dr. Igor Novitski and Roman Novitski must be recognized for their assistance with ANSYS Mechanical APDL and material properties respectively. Also, special recognition should go to my friend and colleague Arnaldo Rodriguez Gonzalez for being an effective sounding board for me this summer.

References

Marscin, Eric, Ryuji Yamada, and Ang Lee. United States. 2D/3D ANSYS Quench Simulation Model.

Batavia: Fermilab, 2002. Print.

Marcin, Eric, and Ryuji Yamada. United States. *3D ANSYS Quench Simulation Program with Full Magnet*

Coil Geometry. Batavia: Fermilab, 2003. Print.

Yamada, Ryuji, Eric Marscin, Ang Lee, and Masayoshi Wake. United States. *3D ANSYS Quench Simulation*

of Cosine Theta Nb3Sn High Field Dipole Magnets. Batavia: Fermilab, 2003. Print.

Karppinen, M., and A.V. Zlobin. United States. *11 T NB3SN DIPOLE DEMONSTRATOR MODEL MAGNET*.

Batavia: Fermilab, 2010. Print.

Appendix

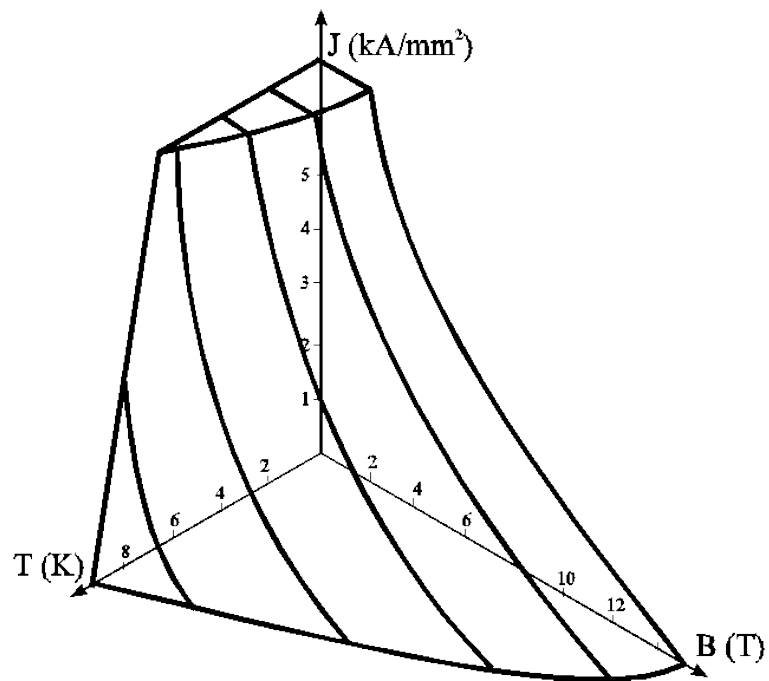


Figure 1: The critical surface

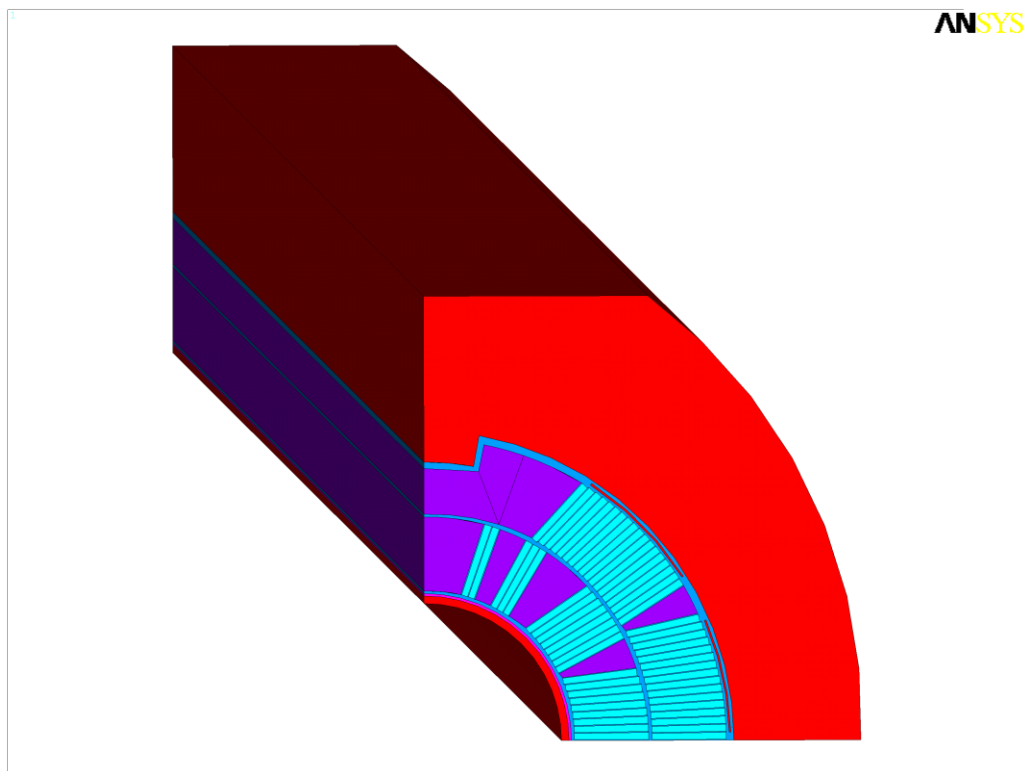


Figure 2: The Old Geometry

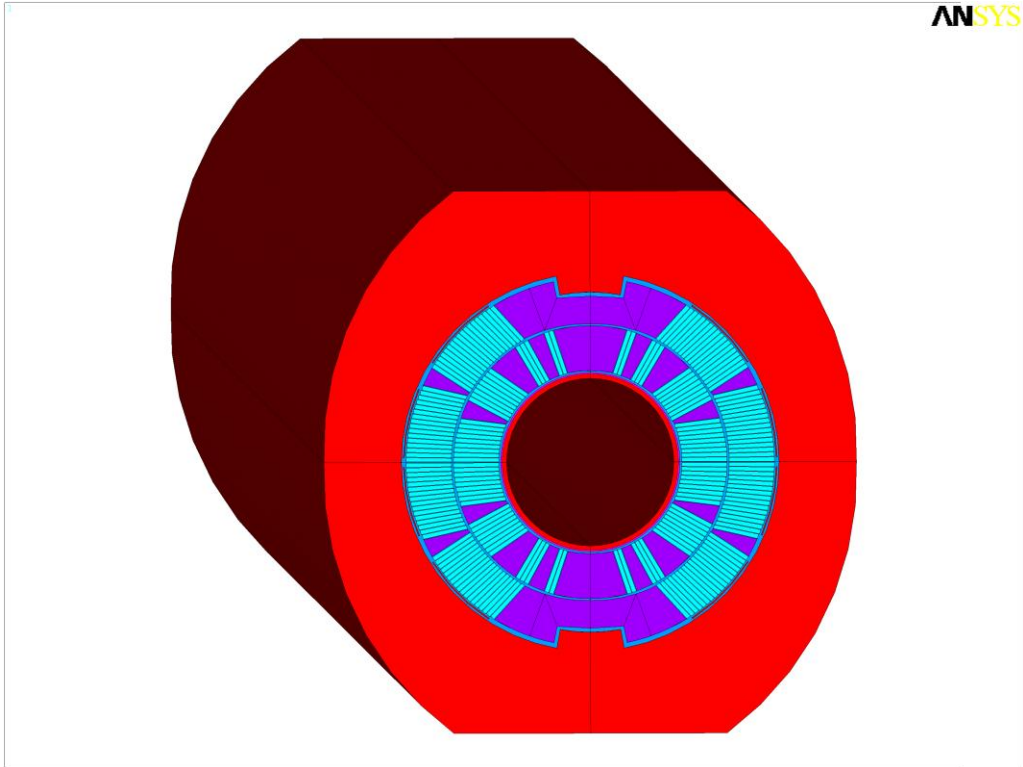


Figure 3: The Full Geometry

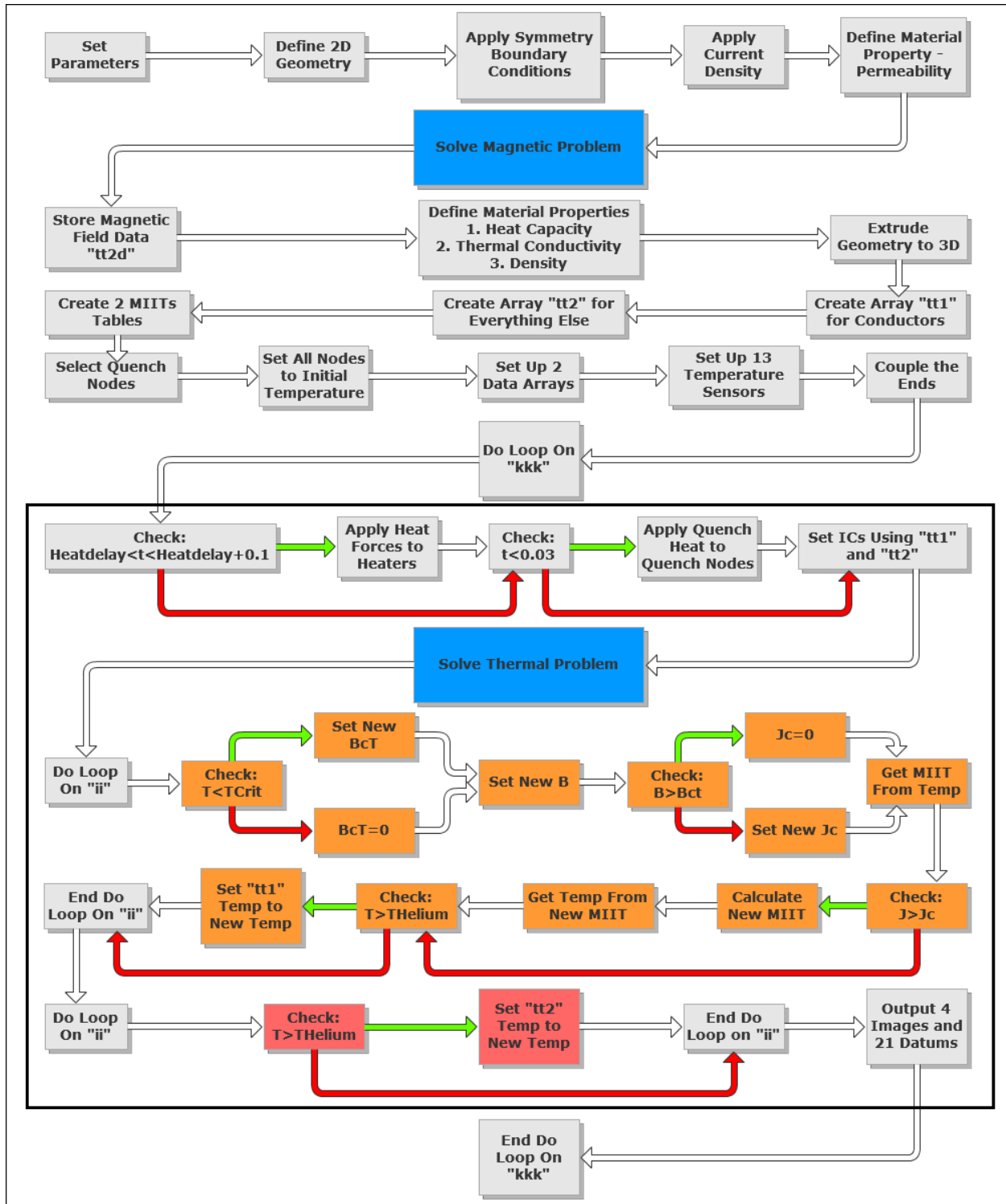


Figure 4: A flow chart detailing exactly how the program works

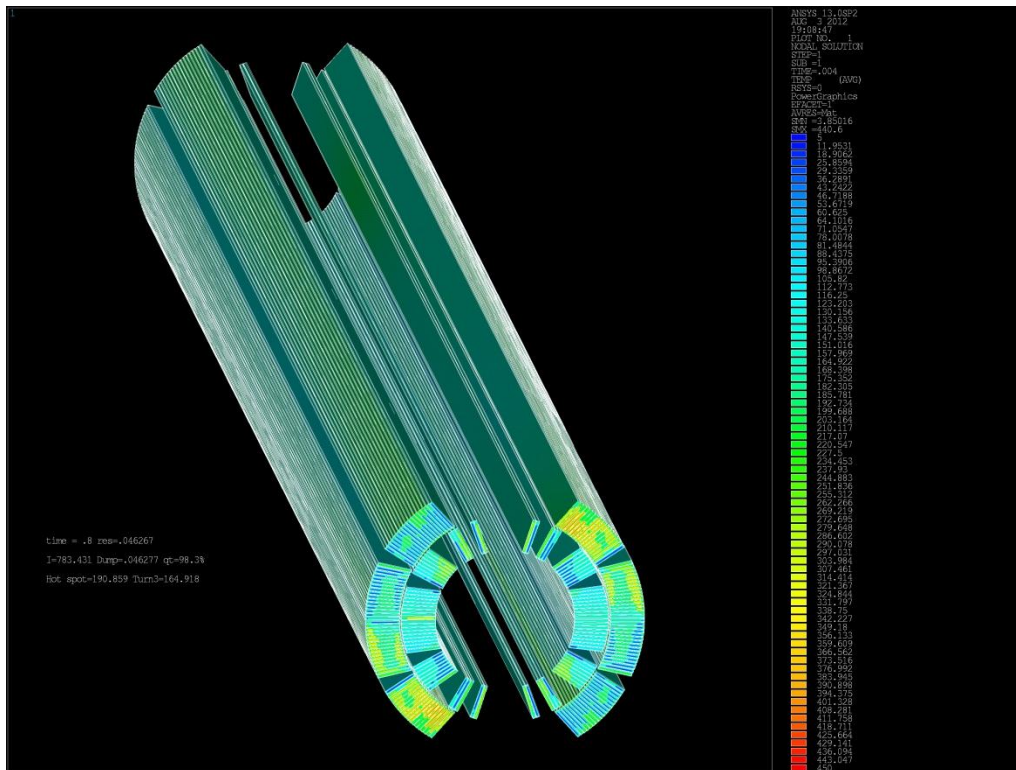


Figure 5: Sample Picture 1 (From Left with Both Layers)

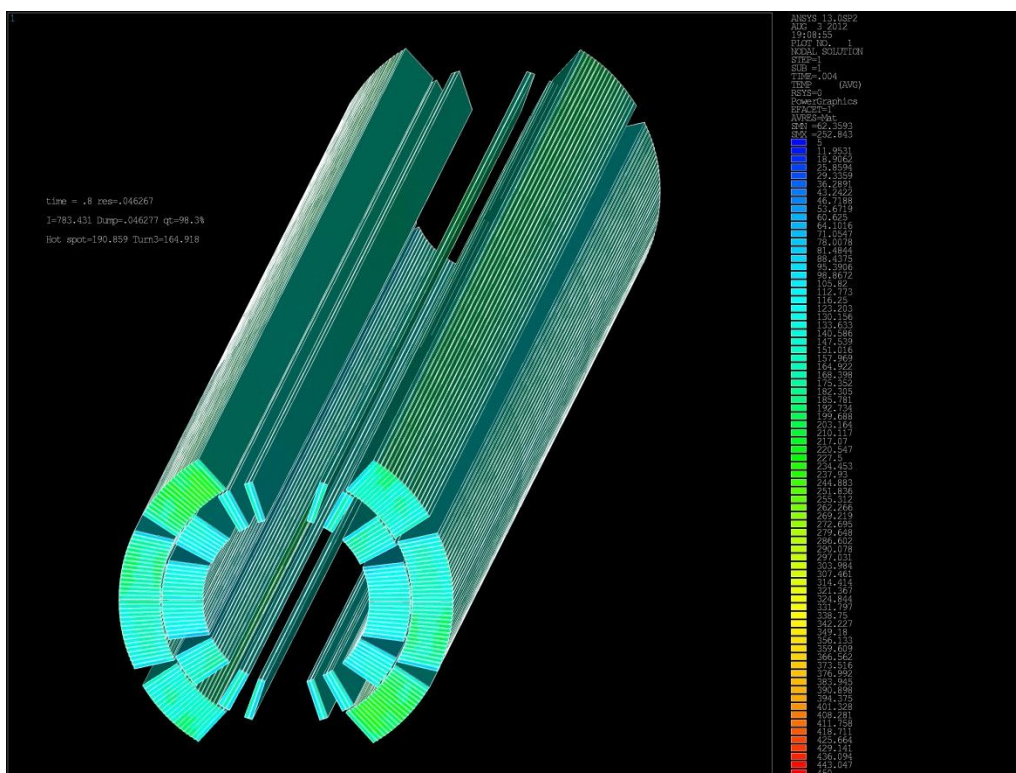


Figure 6: Sample Picture 2 (From Right with Both Layers)

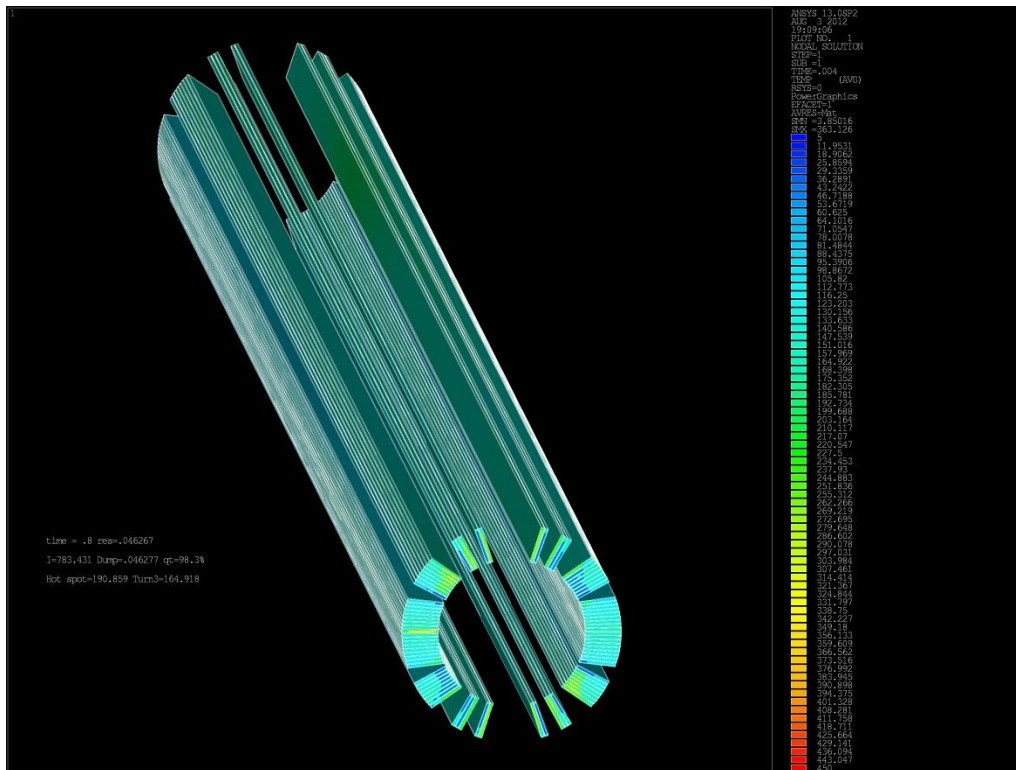


Figure 7: Sample Picture 3 (From Left with Inner Layer Only)

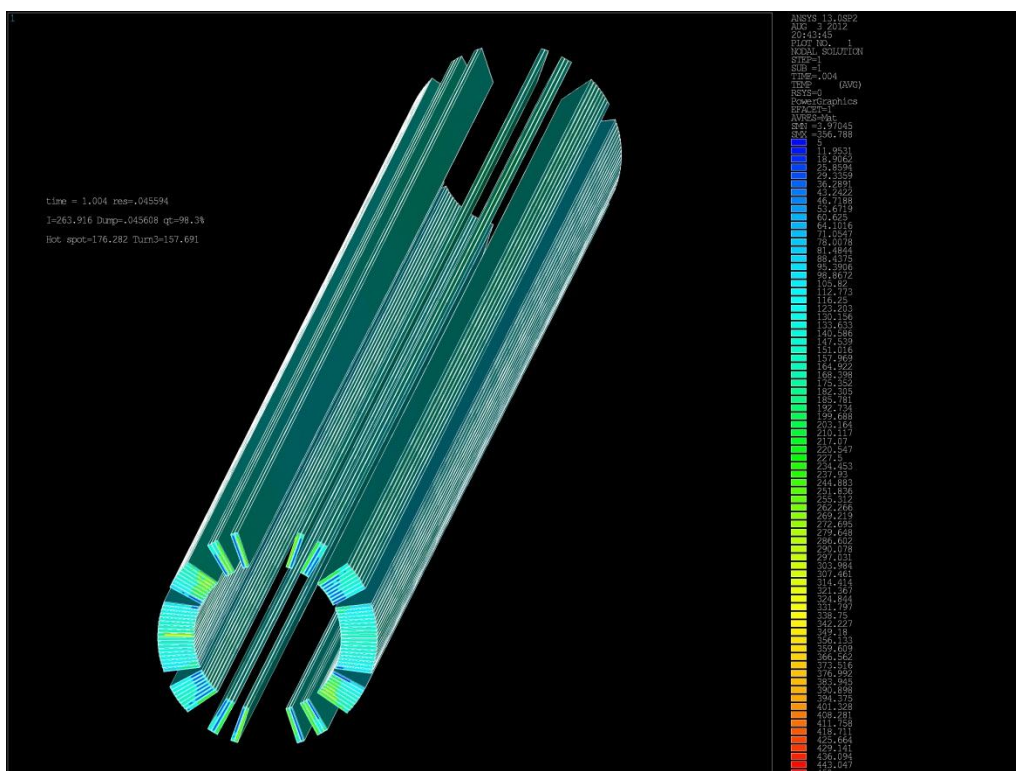


Figure 8: Sample Picture 4 (From Right with Inner Layer Only)

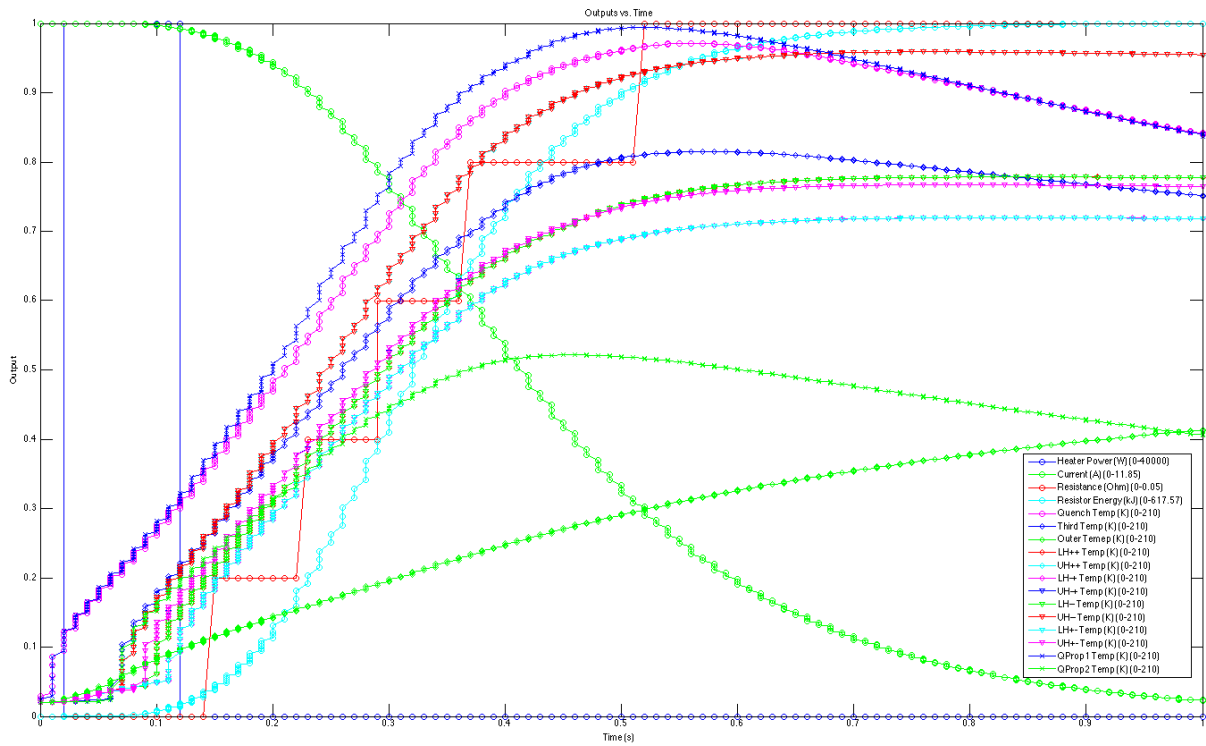


Figure 9: Sample Graph of Output