# A Heat Conduction iPhone and iPad App for Engineering Education

**Dr. Jason M. Keith, Mississippi State University**

Dr. Jason M. Keith is a professor within and the director of the Dave C. Swalm School of Chemical Engineering at Mississippi State University. He is also holder of the Earnest W. Deavenport, Jr. Chair. Prior to joining Mississippi State University, Dr. Keith was employed at Michigan Technological University. Dr. Keith has received numerous teaching and research awards, most notably the Raymond W. Fahien Award from the Chemical Engineering Division of the American Society for Engineering Education. Dr. Keith has taught courses in transport phenomena, separations, fuel cells, and hydrogen energy fundamentals during his academic career.

**Mr. Gerald C Nelson, Mississippi State University**

Gerald Nelson is director of the Office of Entrepreneurship and Technology Transfer. He currently manages OETT for Mississippi State University to protect and maximize value of IP on MSU campus. He also manages MSU Entrepreneurship Center, and the Hatcher Engineering Entrepreneurship Certificate Program. He is director of the Thad Cochran Endowment for Entrepreneurship at Mississippi State University. Since March 2006, he has mentored and funded campus-based start ups at MSU. He manages the endowment budget to accomplish the formation of an entrepreneurial culture at Mississippi State University. Nelson is also the director of the Entrepreneurship Program in the College of Engineering at Mississippi State University. He implemented the Entrepreneurship program at MSU in March 2001 to establish relations and invite entrepreneurial leaders and experts to Seminar Series. He mentors students involved in the program, plans and executes Project Teams, and markets the program to students. He advises the Entrepreneurship Club and the Engineering Toastmasters Club. From July 1999 to March 2001, Nelson was the chief operating officer at Deka Medical Inc. in Columbus, Miss. In this role, he was responsible for manufacturing operations in U.S. and the Dominican Republic. Nelson was also a corporate executive with Mills & Partners from 1993 to November 1998.

**Mrs. Abby Lammons Thompson, Mississippi State University**

Abby Thompson is the Entrepreneurship program coordinator in the Office of Entrepreneurship and Technology Transfer at Mississippi State University. Through her current role at the university, Thompson works to cultivate a culture of entrepreneurship and innovation among faculty, staff, and students. Thompson assists faculty and student start-up companies with developing strategic business and marketing plans. Thompson manages the MSU Business Incubator and the MSU Entrepreneurship Center. Thompson also manages the Jack Hatcher Engineering Entrepreneurship Certificate program in the Bagley College of Engineering at MSU to enhance business skills in engineering students. Thompson received her Bachelors Degree in Industrial and Systems Engineering and a M.B.A. from Mississippi State University.

**Mr. John Louis Gazzini, Nimbus Mobile LLC**
**Read Sprabery, Computer Engineering at Mississippi State University**

# A Heat Conduction iPhone and iPad App for Engineering Education

Abstract

Studies have shown that most students learn by doing. This is increasingly becoming true in today's generation which is full of texting, tweeting, and emails. Students are always on their phones, so why not use them in the classroom? In this paper, we report on the design of a heat conduction app to help students visualize the solution of an unsteady-state heat conduction problem. The mathematical solution to this problem requires some advanced mathematics, and students usually get bogged down in the details of the solution, without understanding the impacts of the mathematical assumptions and/or model parameters on the resulting solution. The app allows students to easily see the temperature profile in the object as a function of time for heating or cooling, and the impact of the system parameters such as the thermal conductivity, heat capacity, density, length, and heat transfer coefficient on the resulting solution.

Introduction

The use of consumer products often follows a process known as the "diffusion of innovation". That is, when a new product, technology, or policy is developed there are innovators, early adopters, the majority of users, and laggards - those who are slow in adopting a new technology. Such a process was conceived by Rogers[1]. In the past, this could be applied to concepts such as recycling or products such as portable CD players, and is currently being applied to concepts like sustainability or products such as smart phones.

Smart phone usage is especially rampant on college campuses – a place full of early adapters and a younger generation of consumers who live in the present and almost demand information instantaneously. Thus, the smart phone presents an opportunity for educators, particularly in engineering, to develop applications, or "apps" to help students learn challenging topics.

In undergraduate heat transfer or transport phenomena courses, students are typically introduced to the solution of partial differential equations. Such problems also arise in graduate core courses in applied mathematics or in many of the engineering disciplines. In this work, we discuss development of an iPhone / iPad app to provide students with a visual understanding of heat and mass transfer dynamics that are not available in traditional textbooks. Educators and instructors can use these tools to enhance student understanding of the fundamental concepts of transient diffusion. In addition, the app is also interactive so that students can make adjustments to key parameters and see their impact on the physical process - which could be a significant factor in a student's success.

Mathematical Description

Heat conduction and mass diffusion are critical topics in the transport and unit operations course. Students usually quickly learn that heat conduction within an object is due to a difference in temperature $T$ between two points separated by a distance $\Delta x$. Students can also can quickly grasp the concept of thermal conductivity $k$ and how is impacts heat flux by Fourier's law of conduction. That is, heat flux $q_c$ in units of W/m$^2$ is given by:

$$q_c = -k\frac{\partial T}{\partial x} \tag{1}$$

The convective heat transfer rate $q_h$ external to an object is also given in units of W/m$^2$ and is governed by the magnitude of the heat transfer coefficient $h$ and the temperature difference between the surface temperature $T$ and the temperature of the fluid $T_f$.

$$q_h = h(T - T_f) \tag{2}$$

Extrapolating the concept to transient conduction leads to significant complications for the student. In addition to the thermal conductivity, the material density and heat capacity are important parameters. Visualizing the impact of the solution with changes in different parameters is also challenging. The most common system is a large flat plate of thickness $2L$, with the system illustrated in figure 1 below.
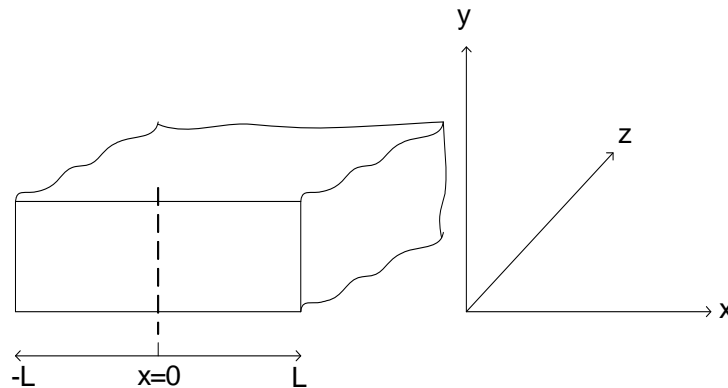


Figure 1. Schematic of unsteady-state conduction in a large flat plate

Thus, the governing partial differential equation is:

$$\rho C_p \frac{\partial T}{\partial t} = k\frac{\partial^2 T}{\partial x^2} \tag{3}$$

The system spans the range from $x = -L$ to $x = L$. Due to symmetry, there is no flux at

x = 0, such that $\partial T / \partial x = 0$. At the surface, where $x = L$, the boundary condition balances internal conduction in the solid and external convection such that $-k\partial T / \partial x = h(T - T_f)$. Furthermore, the initial temperature at time $t = 0$ is a constant of $T = T_0$.

The equation can be solved by the separation-of-variables technique with a solution given by:

$$T = T_f + (T_0 - T_f)\sum_{n=1}^{\infty} A_n \cos\left(\frac{\lambda_n x}{L}\right)\exp\left(-\frac{\lambda_n^2 kt}{\rho C_p L^2}\right) \qquad (3)$$

where the coefficient $\lambda_n$ is given by the transcendental relationship:

$$\lambda_n \tan \lambda_n = Bi \qquad (4)$$

where $Bi$ is the Biot number, given by $Bi = hL/k$.

The parameter $A_n$ is dependent upon the initial condition. For a constant initial condition of $T_0$, we have:

$$A_n = \frac{\dfrac{\sin \lambda_n}{\lambda_n}}{\dfrac{1}{2} + \dfrac{\sin 2\lambda_n}{4\lambda_n}} \qquad (5)$$

App Development

The development environment for "Heat Transfer for Students" is fairly standard because the target platform is iOS and Apple devices. This dictates that development must be done on a Macintosh computer using software provided directly from Apple for development. We used a 15 inch MacBook Pro for all development including algorithm development and testing in addition to user interface design and optimization. An iPhone 4S running iOS 5.1 was used as a test bed and helped in identifying issues with the user interface. All software was coded in XCode 4.2, a proprietary Integrated Development Environment (IDE) specifically for developing applications for Apple products. Development was done in Objective-C, the programming language of choice for iOS.

Development for "Heat Transfer for Students" started by identifying the core algorithm used to calculate the transient temperature change at various points throughout the object. After reworking the infinite series algorithm that can calculate the temperature at a given point over time in the Objective-C language, we

began working on the user interface. This was very important to us because even a minor flaw in user design, such as placing a button where it is unintuitive to find or press, can greatly reduce the quality of an otherwise good application. Educational apps require a high quality user experience more so than other applications because these apps must convey knowledge in a manner that will be meaningful for a student. "Heat Transfer for Students" tries to ease the learning process of heat transfer theory by graphing the heat transfer as an object changes from its original temperature to the temperature of the surrounding body of mass. We used "Core Plot" for development of the graphs within the app. Core Plot is a graphing library for Objective-C that eases the development of graphs in programs written for OSX and the iPhone. This library allowed us to create a dynamic graph quickly without having to write our own graphing effects in Objective-C by providing a number of function calls from within the code to initialize and modify graphs. This graphing library is licensed under the New BSD License, and is available for free use from its Google project-hosting page. Even with the library, much work had to be done to ensure that the graph displayed properly and effectively conveyed the information. By allowing students to enter variables into fields on the iPhone, we allow for visualization of the overall process and the impact of a single variable change. Quick and easy variable identification and visualization of how any one variable affects heat transfer while keeping others constant is just one way students can use the app to aid in the learning process.

While the software we used had no direct cost (Core Plot is open source and XCode is available for free from Apple), there is a cost with uploading apps to the iPhone app store. In order to be eligible to submit apps to Apple, a developer account must be purchased for a yearly recurring fee of $100. After finishing the user interface, we submitted the application to the iTunes App Store, which requires its own approval process. This process ensures that submitted apps conform to the App Store's terms of service and check for malicious code in an effort to protect end users. Having submitted it to the store, we have seen 776 downloads in under a month. There has been no marketing effort other than posting the app to the App Store, so there is clearly an interest in the visualization of heat transfer data.

Instead of providing a clean, mathematical function to the graphing library, we instead had to provide an array of data points (as x,y coordinates) to visualize on the chart. At any given time, the plot is displaying 201 data-points. There are 100 points to the left of the origin, one data-point at the origin, and 100 points to the right of the origin. For the sake of efficiency, the 100 points to the left of the origin are simply mirrored to the right side, because in our simple 1-dimensional model, the temperature will be symmetric across the y-axis. So, for each image displayed, 101 points have to be calculated. Performing this calculation in an effective manner is necessary for the "Animate Graph" feature, which allows the student to actually watch the heat transfer process taking place.

Before calculating these 101 data-points, a vector of possible lambda-values must be generated. The size of this vector is important, because the more values we use, the

more accurate our final graph will be. This vector is generated in a loop, with each iteration of the loop solving a transcendental equation to generate the next lambda-value. During development, we initially used only 8 lambda-values. This affected the accuracy of our data and increased the effects of Gibbs' Phenomena, which causes oscillations at low time-values. We wrote a small function to detect this "ringing" phenomena in the temperature profile and to smooth the graph if necessary, but the more mathematically accurate solution was to increase our lambda-vector to a size of 50. With the longer vector, each of our data-points was more accurate, and the misleading oscillations were no longer seen in the final graph.

The final process to produce the graph for the student is actually quite involved. Each time the student changes a variable on the screen, 50 new lambda-values are instantly generated. Then, 101 new data-points are calculated, each requiring 50 iterations (for the lambda-vector). These 101 data-points are placed into an array, and then 100 of them are mirrored and appended to the array as well, for a total of 201 data-points. These points are then provided to the graphing library, which eventually plots the graph to the screen. Using an iPad 2 running iOS 5.1, this can be animated at approximately 10 frames per second, though it is worth noting that the graphing library is by far the largest bottleneck in the overall process, and the lambda-vector is actually shared among all of the data-points and only has to be calculated once for the entire animation.

The final project contained 10 files for the Objective-C code. Of that code, approximately 120 lines were used for the actual calculations, and approximately 1200 lines of code were required for the user interface. This included error-checking and validation of user-input, certain animation sequences, and general interactions (such as linking button-presses to actions). The user interface also consists of 2 "Storyboard" files and 18 image files, to support a variety of screen resolutions. The total development time for the app was 26 hours. Sample images of the process are seen in the figures 2-3 below.

Functions and features of the App

Screen captures of the app "Heat Transfer for Students" is shown in figure 4 for the iPhone and in figure 5 for the iPad. There is an area that allows the user to enter the parameters for the slab. Default material properties are for butter, based upon Example 5.3-2 in Geankoplis[2].

Upon scrolling down, the user will see that they are presented with options to animate the graph, clear out all variables, or auto-fill variables with the default values. As a test case, the properties from Example 5.3-2 in Geankoplis[2] are entered. The initial temperature $T_0$ = 277.6 K, the fluid temperature $T_f$ = 297.1 K. The convection coefficient is $h$ = 8.5 W/m$^2$-K, the thermal conductivity is $k$ = 0.2 W/m-K, the product of the density and heat capacity is 2300 kJ/m$^3$-K, and the half thickness is $L$ = 4.6 mm.
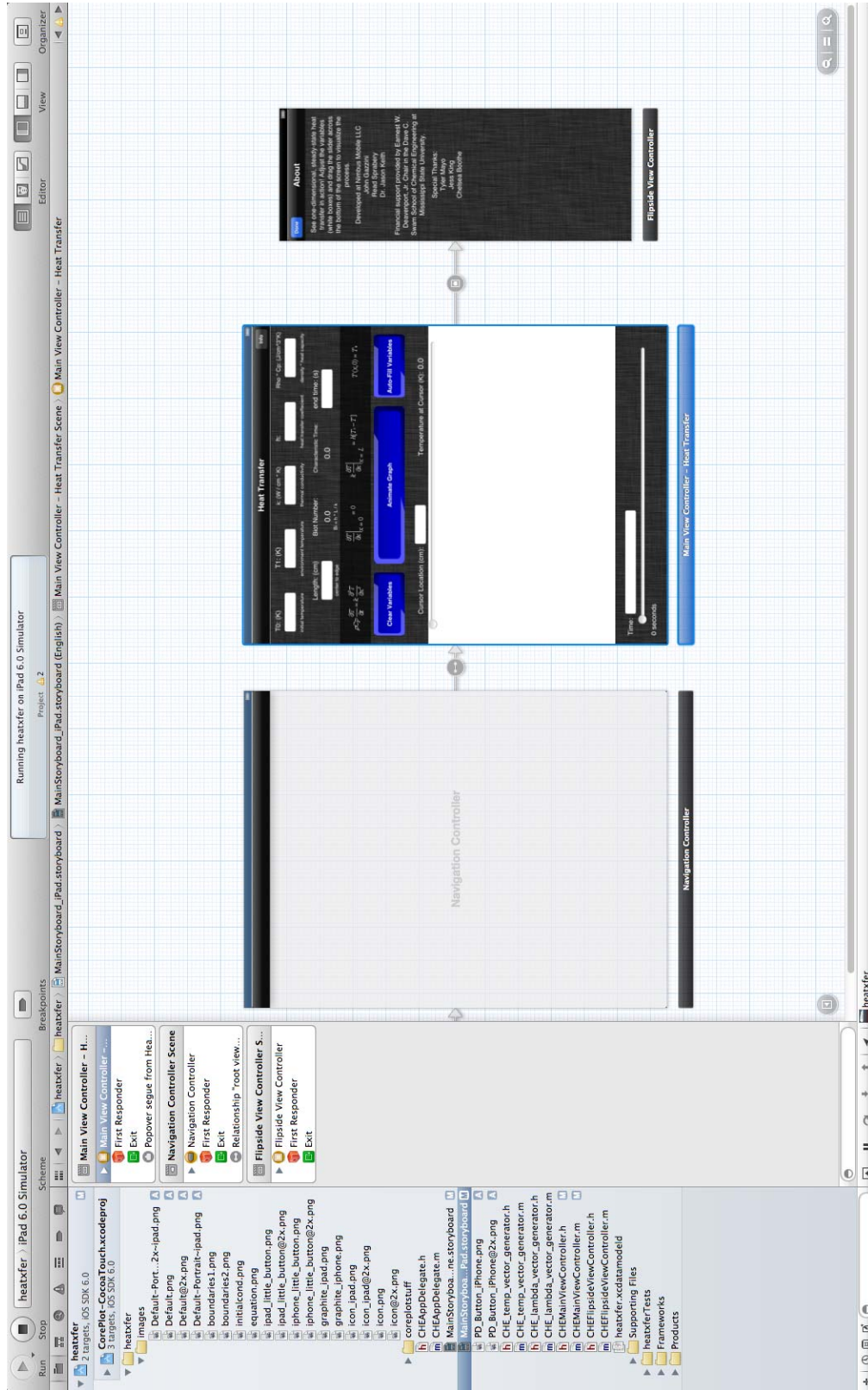
Figure 2. One of the two "Storyboard" files. This one defines the layout for the iPad.
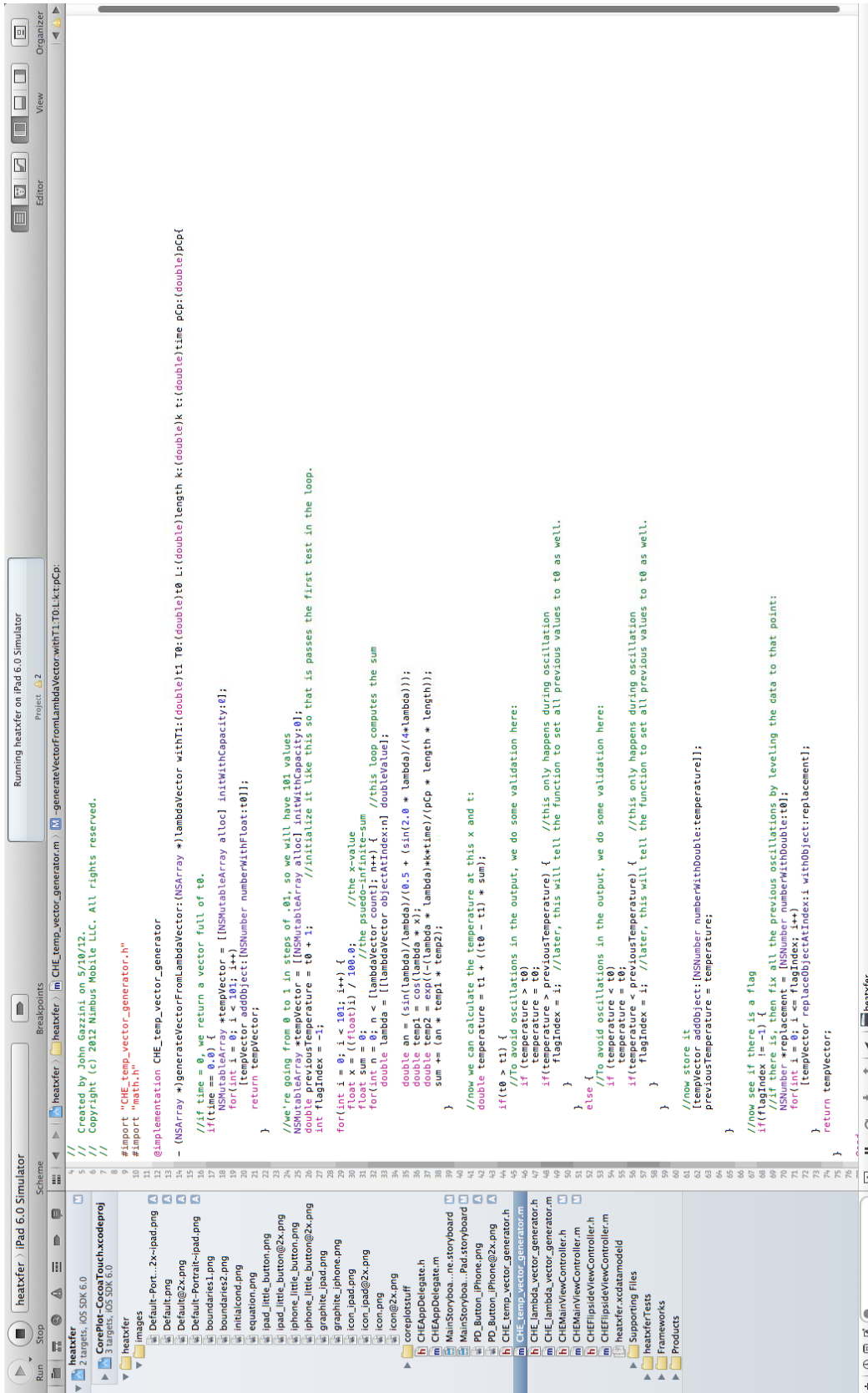
Figure 3. A short Objective-C file seen in Xcode. This class generates the temperature vector.
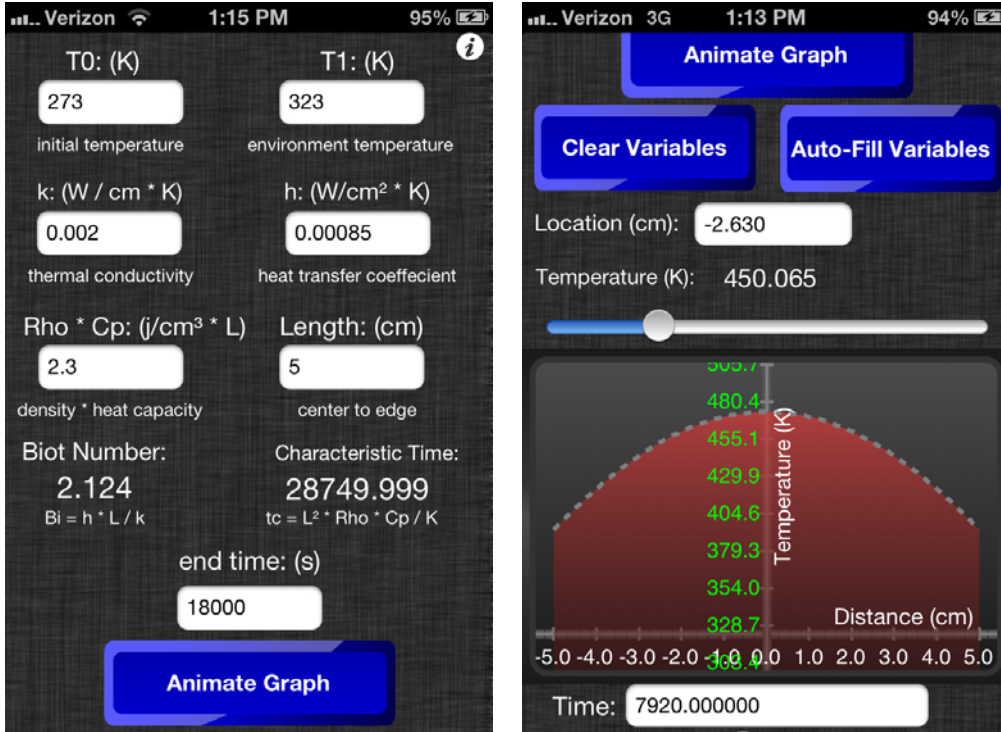
Figure 4. App Screenshots for iPhone. The left image shows the parameter input area. Scrolling down shows the right image, a snapshot of a cooling process.
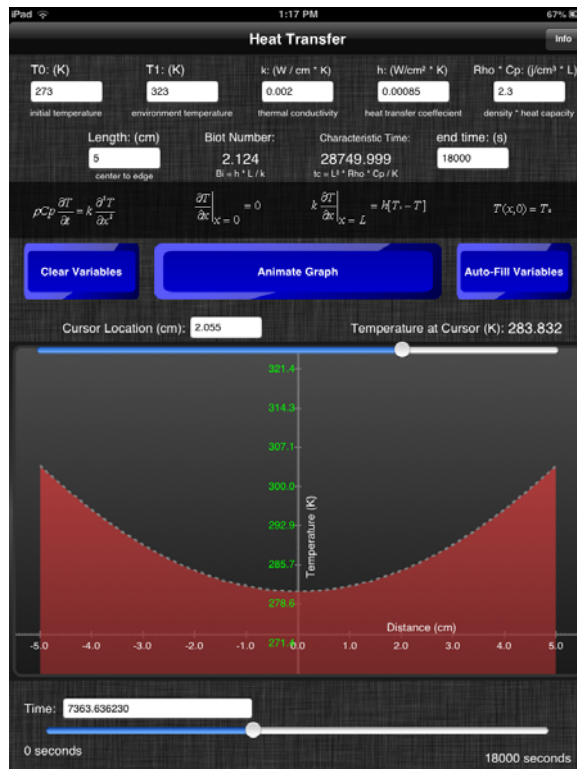


Figure 5. App Screenshot for iPad

Entering these parameters and running the simulation allows the user to see the process of heat conduction in the solid material. There are two key parameters that are output: the Biot number = 1.95 and the characteristic heat conduction time $t_c = 23{,}000$ s. After running the simulation, user can also adjust parameters and re-run the simulation to see, for example what happens when Bi or $t_c$ is increased or reduced.

After a time of 5 h, the surface temperature can be determined by adjusting the upper slider bar to give the surface temperature of 292.4 K (compared with the text solution of 292.2 K) and an internal temperature at a distance 0.208 cm from the center of 288.4 K (compared with the text solution of 287.4 K).

The engineering education literature[3-5] has shown that students exhibit different learning styles. Moreover, many of today's students are visual learners and benefit greatly from seeing the physical processes taking place.

The download history as of the submission of this paper is summarized in the table below.

Table 1. Download history for Heat Transfer for Students App
**Total Downloads: 2651**
USA: 1165
UK: 146
Canada: 101
Thailand: 135
India: 63
Mexico: 67
Turkey: 69
Saudi Arabia: 60
Italy: 51
Brazil: 54
"Others": 740

Conclusions

This paper demonstrates an iPhone and iPad app called "Heat Transfer for Students" to allow students to visualize heat transfer. It is available for free on the app store. Educators and instructors can use this app for homework assignments, an in-class demonstration, or an in-class problem.

Bibliography

1. E. M. Rogers, <u>Diffusion of Innovations</u>, Glencoe Free Press (1962).
2. C. J. Geankoplis, *Transport Processes and Unit Operations, Fourth Edition*, Englewood Cliffs, NJ: Prentice Hall (2003).
3. R. M. Felder, "Matters of Style," *ASEE Prism*, **6(4)**, 18 (1996).
4. D. A. Kolb, *Experiential learning: Experience as the source of learning and development*, Englewood Cliffs, NJ: Prentice Hall (1984).
5. J. E. Stice, "Using Kolb's Learning Cycle to Improve Student Learning," *Engr. Education*, **77**, 291 (1987).