CSUMS Final Report, Spring 2011

Peter H. Jumper
Undergraduate Student
Dept. of Physics
UMass Dartmouth
Dartmouth MA 02747
Email: pjumper@umassd.edu

Contents

Abstract

This paper discusses the work I conducted in relation to the brown dwarf desert problem through the spring 2011 semester in the MTH 499 CSUMS class. It first provides background by describing the problem and the techniques used to address this problem, including the methodology of the code I am using and the strategies being used to debug the code. Then it discusses my efforts throughout the semester to solve this problem, which unfortunately have been unsuccessful due to an unresolved bug in the code. The paper then discusses what I have learned in astrophysics and programming during the semester and provides constructive criticism of the class, suggesting allowing more time for research during class time.

1 Description of the Problem

This semester, I worked on the problem of the brown dwarf desert. Brown dwarfs in binary systems are almost never found within 5 AU (5 times the distance between the Earth and the Sun) from the central star of the system. The mechanisms by which this absence, or desert, of brown dwarfs results present an unsolved question in astrophysics. Understanding the formation of the brown dwarf desert would give us insight on the role of angular momentum in star formation processes and would also help us understand brown dwarf formation processes. While some past researchers have proposed a migration model to explain the desert, I have been working on researching the desert through a gravitational fragmentation model with turbulence under my research advisor, Dr. Robert Fisher, and later with a graduate student, Deivid Riberio. We plan to test this model and analyze the results to gain insight about the brown dwarf desert problem. However, we have encountered major errors in our code, so we have been engaged in debugging to allow us to correct this and progress with the primary research.

2 Techniques Being Used to Address Problem

2.1 Code Methodology

2.1.1 Mass Generation and Description

The code first randomly generates two masses from the initial mass function (IMF), which describes the frequencies at which bodies of particular masses form. These become the masses of the two members of the binary system in the simulation. Then, using a star formation efficiency parameter, the mass of the original core that condensed to form the binary is calculated. In this model, only a fraction of the core mass ends up in the binary; the remaining mass is driven from the protostellar disk during the formation process. This is described by the equation

$$M_{core} = \frac{m_1 + m_2}{\epsilon_*}$$

, where M_{core} is the mass of the core, m_1 and m_2 are the masses of the two members of the resultant binary system, and ϵ_* is the star formation efficiency [3]. We then assume in the model that the core may be derived by a Bonnor-Ebert sphere, which involves a gas pressure balance with gravity. The solution is truncated at a certain density cutoff, which lets us describe the estimated boundaries of the core and it density distribution [4].

2.1.2 Angular Momentum Generation and Description

Next, we have the program generate an angular momentum distribution by producing three Gaussian perturbation spectra cubes. Each cube corresponds to one velocity component and has the velocity values in each of the cells in their 128 x 128 x 128 array set according to a Gaussian random field, which has uncorrelated modes. The perturbation cubes are then superimposed over the Bonnor-Ebert sphere description, with only the region of intersection between the cube and sphere considered. This assigns the random velocity components to only the cells that fall within the Bonnor-Ebert sphere description of the core. Using the density distribution of the Bonnor-Ebert sphere and the velocities assigned by the perturbation cubes, the angular momentum at each cell in the code's representation of the core is found; from this a net core angular momentum is calculated. Using a angular momentum efficiency parameter, the angular momentum of the binary system is calculated [4] This calculation uses a similar formula to the star formation efficiency formula: $J = J_{core}\epsilon_J$, were J is the angular momentum of binary system, J_core is the angular momentum of the core, and $epsilon_J$ is the angular momentum efficiency [3].

2.1.3 Simulation Results

Orbital eccentricities are also drawn, allowing the orbital characteristics of the binary to be determined, including the period and the semimajor axis. These results are calculated in accordance to the following equations:

$$P = (\frac{2\pi}{G^2})(\frac{J^3}{M^5})\frac{1}{(1-e^2)^{\frac{3}{2}}}\frac{(1+q)^6}{q^3}$$

and

$$a = \frac{1}{G}(\frac{J}{M})^2\frac{1}{(1-e^2)}\frac{1}{M}\frac{(1+q)^4}{q^2}$$

, where P is the periods, G is the universal gravitational constant, J is specific angular momentum, M is the mass, e is the eccentricity, and q is the mass ratio of the companions, defined as a number less than or equal to 1 [3].

Each simulation sweeps through possible combinations of efficiency parameters and generates numerous systems for each combination (typically at least 200). A researcher can then transmit the output file for a particular combination to a program that generates graphs and statistics [4].

2.2 Code Verification and IMF Update

The code was previously prepared by Dr. Fisher for a 2004 paper on the binary period distribution [3]. Its current IMF does not include the brown dwarf portion of the IMF as adequate information on this was unavailable at the time. Although a new, updated IMF including the brown dwarf portion has been prepared, the group is first attempting to reproduce the results of the 2004 paper before making this modification to verify that the code is working as expected. After this verification, the update will be made and the simulations will provide data that allows us to analyze our model and the brown dwarf desert problem.

2.3 Techniques to Isolate Bugs in the Code

A series of errors, including not a number (NaN) outputs were unexpectedly found in our output files and statistics, prompting debugging efforts. In this effort, our primary techniques have been analyzing the behavior of the code when simplifying assumptions are made and module testing. The former involves temporarily commenting out one section of code, inserting a simplifying case in its place, hand calculating expected results, and seeing if these match. The primary example of this technique has been our use of a rigid, rotating body assumption. The latter technique involves looking the original code in small sections and making sure that each section is individually behaving as expected.

3 Process

3.1 Review of Notes from Preliminary Work During the Summer and Winter

Prior to beginning the project in earnest this semester, I had done some preliminary work and reading during the summer and winter. Thus, when the project began, I quickly reviewed the notes that I had taken.

3.2 Initial Project Scope

When the project began, my task was to write a Gaussian random field generator for the code. Although the original version of the code had one, it seemingly did not work on the modern compilers that the group was using. The new generator code that I wrote would replace the old one and restore our ability to use the simulation code. Dr. Fisher believed that this task would be simple for a graduate student but would present an advanced project for an undergraduate such as me. After this was completed, we would progress to our actual research, in which we would try to understand how angular momentum and turbulence in the star-forming gas cloud, the core, influenced the binarity properties in star systems, particularly in regard to the brown dwarf desert phenomenon.

3.3 Writing the Proposal and the Abstract

Early in the class, I was required to write a proposal with citations for CSUMS and an abstract to send to the Undergraduate Research Conference at the University of Massachusetts Amherst. Dr. Fisher required numerous revisions to my abstract over several drafts, as it would be representing both of our work. Thus, writing the abstract was probably more time-consuming than writing the proposal.

3.4 Review of FORTRAN Programming

Although I had worked extensively with FORTRAN programming during the summer CSUMS program, I had not done very much work with FORTRAN during the fall and winter, leaving me out of practice. Since my initial task required writing a FORTRAN code, reviewing FORTRAN was imperative, so I reminded myself of its coding techniques and commands by reading the first-order Godunov solver code that I had written during the summer CSUMS program. I entered my notes on this into my notebook for easy future reference.

3.5 Attempted Planning of Code

After this review, I started to outline my planned structure for the code in my notebook. Almost immediately, I realized that I did not understand enough about the workings of the rest the other codes used in the simulation to write my code properly. I decided to seek help and arranged a meeting with Dr. Fisher to discuss the project. Furthermore, after the meeting I realized that I had fundamentally misunderstood what the Gaussian random field generator was supposed to do, which had contributed to my confusion.

3.6 Meeting with Dr. Fisher about Project

3.6.1 Summary

I met with Dr. Fisher in his office on President's Day, February 21, to discuss the project and clarify my understanding. During this meeting, he reviewed most of the content that I have discussed in the "Code Methodology" subsection. Consequently, I realized that I had previously been under the mistaken impression that the Gaussian random field generator was related to drawing masses from the IMF. Additionally, we discussed the significance of turbulence in our model, experimental evidence for turbulence, plans for restructuring the IMF code when necessary, and the optimization of code [4].

3.6.2 Evidence for and the Importance of Turbulence

Dr. Fisher told me that experimental evidence for turbulence in cores is provided by our observations of emission lines (from ammonia in his example) along our line of sight to the core. All emission lines have a minimum line width, the natural line width, set by the Heisenberg uncertainty principle. The observed line width is increased by both the thermal and non-thermal (bulk) motion of the particles in the core, the latter of which is dominates and is associated with the Doppler Effect. As one looks at the core along the line of sight, the lines tend to red shift, providing evidence of motions within the core. These motions are thought to be turbulent. In turn, core turbulence provides the source of angular momentum in our models: although the smaller turbulent eddies in the core will tend to cancel each other, the large scale eddies ultimately dominate and determine the net angular momentum [4].

3.7 Reading Scientific Papers for Background

After the meeting, Dr. Fisher asked me to read a pair of scientific papers to provide further details to what we had discussed during the meeting. The first of these was "Turbulent Molecular Cloud Cores: Rotational Properties" by Andreas Burkert and Peter Bodenheimer. Many of the points that Dr. Fisher discussed about turbulence were covered in this point, but to a much greater depth [1]. The second was Dr. Fisher's own 2004 paper, "A Turbulent Interstellar Medium Origin of the Binary Period Distribution." This detailed the equations and methodology used in the code and provided Dr. Fisher's previous results [3]. These were greatly helpful for reinforcing what Dr. Fisher had discussed at the meeting.

3.8 Changed Objectives

Around the time of this meeting, a major change was made to my objectives when Dr. Fisher unexpectedly managed to get the original Gaussian random field generator working, rendering the new generator that I was supposed to build unnecessary. Instead, I would now begin by verifying that the code was running properly by reproducing Dr. Fisher's 2004 results. After that, we could add the new initial mass function code and immediately commence the primary research.

3.9 Loading Errors in analysis.m

After clarifying a minor understanding of the exact command sequence I need to enter on the command line, I was able to run the main code and generate an output file. Once I transferred this file to the server that had the statistics and graph-producing script, analysis.m, I attempted to load the file to analysis and execute the script. I received an error message stating that analysis.m only accepted input files named periods.output, something that I confirmed when I reviewed the code. Therefore, I copied the output file to a new file named periods.out and tried again. I got another, different error message. Baffled, I consulted online resources about the load commands to try determining what I was doing wrong. This was unsuccessful, so I decided to conduct a module test of the load command. I copied this command into a new script file, test.m, and then wrote a simple, hypothetical output file with five integer numbers. I named this new output file jumper.output and then changed the load command in test.m to accept input from jumper.output. The five numbers loaded successfully. Since the only change I had made to the load command was the file that it accepted, this demonstrated that I knew how to use load commands correctly and that analysis.m's load command was probably correct. This suggested that something about the numbers in the output file itself were responsible for the error. Since the numbers in the actual periods.output file were written in a scientific notation, I decided to repeat my test with the numbers in my jumper.output test file changed to values in the same form of scientific notation. This time, an error resulted. It appeared that the character used to denote scientific notation, E, was causing the load command to not recognize the numbers in the output file as numeric values. I consulted Dr. Fisher, who suggested that I replace the E in the scientific notation with a D, which was another accepted convention. This change worked; the E character had indeed been the problem. However, since the output file had 1600 values in it, I needed to learn a find and replace command in order to change all instances of E to D efficiently.

3.10 Troubleshooting analysis.m

Although I had succeeded in getting a data file to load to analysis.m, I found to my dismay that there were now other compilation errors. I had not previously encountered these because the program had terminated at the load command before reaching them. I began a systematic approach to correcting these errors, checking the code at the line numbers referenced by the error messages. The first issue was with the arguments of a function called errorbar. After consulting Dr. Fisher, I simplified errorbar's arguments by removing unneeded optional arguments. After this, error began executing. Next, I received warnings that the program was unable to load a file named dmperiods.output. Referring to the code, I realized that there were a group of lines responsible for generating graphs based on the data produced by two scientists to whom Dr. Fisher was comparing his 2004 paper's work. Since this comparison was not part of my project and I did not have the relevant data file, I resolved this issue by commenting out all the relevant lines. After this was done, I was finally able to run the analysis script.

3.11 Installing Ghostscript and Ghostview

When the analysis script ran, it printed the calculated statistics on the command line and created the graphs in postscript files. However, when I tried viewing the graphs, I found that my computer seemed incapable of opening their files correctly. When I contact Dr. Fisher about this problem, he instructed me to download a free program called GSView 4.9 (Ghostview) that would allow me to open and view these files. However, Ghostview required another program called Ghostscript to run, so I needed to download this first. This was a straightforward procedure.

3.12 Establishing SVN Repository

In addition to my work with the code and simulations, Dr. Fisher had asked me to establish a Subversion (SVN) repository, a revision control system for backing up data, in my user files on the server the main code was on. This would allow me to revert to earlier versions if I accidentally made a catastrophic error when working with the code. I had had some slight previous experience with SVN during the summer 2010 CSUMS program, I had found the process extremely confusing and did not truly understand. Cognizant of my weakness with SVN, I looked up several online resources on SVN to help guide me through the process. I tried repeatedly, but I was continually frustrated, particularly during instances in which I had thought that I had made a working SVN repository, only to find that I could not checkout and commit revisions properly. Eventually, I contacted the graduate student Avinash Kumar, who had helped me significantly with SVN during the summer, and asked him for help. In response, Avinash emailed me the set of instructions from which he had learned SVN. These worked and I wrote them down in my research notebook for convenient reference.

3.13 Attempts to Reproduce Dr. Fisher's Results

Now that I had eliminated the compilation errors that I had encountered and had established the SVN repository that Dr. Fisher requested, I was free to begin my attempts to reproduce Dr. Fisher's 2004 results as a verification of the code on March 7. Near the end of his paper, a data table listed his results for five combinations of efficiency parameters. I set up the code to run a simulation for one set of these parameters with 200 systems and then executed the program, which took about ten to fifteen minutes to finish. I then transferred the data file to the server with analysis.m on it, executed the analysis script, and wrote down the results. Then I repeated this for the remaining cases.

3.14 NaN Results and Other Anomalies

When I reviewed the results, I was highly disturbed to find that some of my output files, and thus the results from the analysis script, had not a number (NaN) values. Since all the values we were testing for were expected to have positive, real values, getting a NaN result showed that there was some fundamental error. Furthermore, since Dr. Fisher had previously used a similar version of this code to produce his results, which did not have NaN results, something seemed to have happened to the code to introduce these errors; perhaps we had accidentally introduced a catastrophic change to the code. Additionally, several of the real value result did not seem to agree as closely with Dr. Fisher's Results as closely as I would expect.

3.15 Inserting Code to Test Periods

When I contacted Dr. Fisher about the NaN problem, he noted that there were only a small number of NaN results in the output files, which that I had also noticed. Thus, he suggested that perhaps a simple sanity check could be added after the period was calculated to test that the value was real, retrying the simulation if it was not. I inserted this into the code, but it did not seem to stop the problem.

3.16 Reading about CDF of Chabrier IMF

As spring break arrived, I received an email from Dr. Fisher requesting that I read an article that he had posted on our research group's wiki about the revised IMF that we would be using in the project, based on the work of Chabrier [2]. I took extensive notes to help me understand the complicated equations discussed in the article and then resume working on the code.

3.17 Testing the Code Again

I repeated my earlier tests of the simulation to see if the NaN problem that I had previously observed would repeat itself consistently. Additionally, Dr. Fisher had asked me to see what would happen to the results if I deleted the output lines with NaN values and used analysis.m on the remaining lines, so I did this. However, I found that the numbers that results, albeit real, disagreed with Dr. Fisher's results at many points.

3.18 Reading Grether and Lineweaver

During spring break, Dr. Fisher also emailed me the paper, "How Dry is the Brown Dwarf Desert? Quantifying the Relative Number of Planets, Brown Dwarfs, and Stellar Companions Around Nearby Sun-Like Stars" by Grether and Lineweaver. He particularly wanted me to see a good graphical representation of the brown dwarf desert that was included in the paper. I reviewed this figure and then took notes on the rest of the paper [5].

3.19 Instant Messaging Conferences with Deivid over Spring Break

Partway through the project, another student had joined the project, Deivid Riberio, a graduate student. Although we were supposed to coordinate our efforts with each other, our communication prior to spring break had been minimal and ineffectual, partially because we experienced significant lags between sending emails and our partner receiving and responding to them. Then, during spring break, Deivid proposed that we could significantly improve our communications by utilizing an instant messaging program, such as Skype. I did not have Skype, so Deivid provided me instructions for downloading and registering for Skype. Once this was done, I began an instant messaging call to Deivid through Skype, commencing a two-hour conversation.

I quickly learned that although Deivid had heard Dr. Fisher's general explanation of the code, he had never worked with FORTRAN before and was thus having difficulty understanding the code. Therefore, I began by giving a tutorial about the basic commands used in the code and the rules of writing FORTRAN, focusing on parts of the code that Deivid had asked me about. This was an interesting experience, particularly considering that I had only learned FORTRAN during the summer; I realized how far my FORTRAN skills had progressed in that time and was reminded that many commands that I now considered basic could easily confuse a beginner.

The next day, Deivid and I both ran the code, setting the seed value in the random number generator to an agreed upon value. (As the "random" number generator is actually an algorithm for creating pseudorandom number, using the same seed value made sure that Deivid and I would get the same results for our tests). Our results were again inconsistent with Dr. Fisher's prior results and had some occurrences of the NaN problem.

3.20 Attempts to Create a Shared SVN Repository

Dr. Fisher had also asked Deivid and I to attempt creating a shared SVN repository that would allow both of us to work from and conduct revision control with the same base of code. I found some instructions online, but was left even more confused than when I attempted to create my personal SVN repository. Deivid was similarly unsuccessful, so we decided to postpone work on this task until later. However, we still have not returned to this issue.

3.21 Producing a Raster Plot

Toward the end of spring break, Dr. Fisher suggested that generating a Raster plot could be helpful in attempting to analyze the error in the data. This plot would show the values on a two-dimensional grid by color-coding the cells according to their volume. In this case, I would take a two-dimensional cross-section of the simulated core angular momentum from a data file provided by the program and use the Raster plot to help demonstrate the behavior of the random field generator. I looked up the function and argument structure for producing a Raster plot in Octave and understood the instructions clearly. However, when I located the data file that I needed, I soon realized that the values in the data file were not arranged in a manner useful for making the Raster plot. Although I knew in principle how to rearrange the data set into a useable form, the process would take prohibitively long as the file had millions of values (as it was storing information on a 128 x 128 x 128 cube). I informed Dr. Fisher of the problem, carefully explaining how the data set was not properly formatted for making the desired Raster plot and asking if he knew of any commands that could allow for an easy reformatting. He responded by giving me instructions detailing how to use the reshape function, which would reformat a data set according to the dimensions that I specified, allowing me to put the data in a useful form for the Raster plot function. Thus, I was able to generate the Raster plot, although it did not seem to provide much insight on what was going wrong in the code.

3.22 Preparing for Conference Presentations

As spring break ended and April approached, I needed to begin preparing for the four conferences that I would be presenting at during that month: the Society of Physics Students (SPS) Zone 1 Meeting at the University of Massachusetts Dartmouth, the American Physics Society and American Association of Physics Teachers (APS-AAPT) Joint Meeting at the University of Massachusetts Lowell, the Sigma Xi Conference at the University of Massachusetts Dartmouth, and the Massachusetts Statewide Undergraduate Research Conference at the University of Massachusetts Amherst. Reflecting on what I had done and learned during the project, I assembled a PowerPoint presentation and a talk. I submitted the former to Dr. Fisher for criticism and ultimately included more images on his advice. A few weeks later this PowerPoint was used as the basis for my Sigma Xi poster, built on a PowerPoint template that Dr. Zarrillo provided to me upon my request.

3.23 Developing a Hand Calculation for Specific Angular Momentum

While I had been preoccupied with the conferences, Deivid had prepared a test of the code. Dr. Fisher had suggested that we insert a special case scenario into the code in which we made the simplifying assumption that the core was a rigid, uniformly rotating object of uniform density, as this model was so simple that we would be able to calculate the expectation value of the specific angular momentum by hand and then determine if the code produced an answer consistent with this. In the meantime, the turbulence would be commented out of the simulation. If the answer was consistent with our expectations, this would let us significantly narrow down the source of the error. Deivid had already made the necessary code changes and derived a formula for our expectation value by the time I resumed work on the research, so I quickly derived the same formula independently to double check his work. This equation was

$$\frac{J}{M} = \frac{2}{5}\sqrt{3GM\beta R}$$

. G is the universal gravitational constant, M is the core mass, β is the ratio of the core's kinetic energy to gravitational potential energy, and R is the core radius.

After this, I contacted Deivid and informed him that I agreed with his derivation. Deivid then indicated that he was almost ready to conduct the test, but needed to find the values of some of the variables from the code before proceeding, particularly the radius. Collaborating through another instant messaging conference, I looked through the code with Deivid and found the information that we needed. We then both independently calculated the expectation value from the formula and agreed upon the result.

3.24 Testing J/M with Deivid

Deivid then ran the simulation and announced his result by an instant message: the simulation had failed to produce a value and three of the eight columns in the output file, the period, the semimajor axis, and the specific angular momentum, were filled with NaN results. This was very disturbing as even the simplified case of the simulation was producing physically impossible results.

3.25 Module Testing of the Code

Afterwards, Deivid and I agreed that we should focus our efforts on module testing the code, particularly the sections that were producing NaN results. We also decided to initially work on this task independently so that each part of the code would be reviewed more times, hopefully increasing the chances of finding an error. I started my efforts in this module testing by verifying that the final equations for the semimajor axis and the period were the same in the code as listed in Dr. Fisher's paper.

3.26 Exercise: Estimating Angular Momentum in the Solar System

Later, at an astrophysics research group meeting, Dr. Fisher proposed that I attempt estimating the angular momentum of the solar system and the specific angular momentum of its planets as an exercise to help me better understand the magnitudes of the angular momenta that I was working with in the research. I derived a specific angular momentum approximation on the blackboard during the meeting and then looked up the orbital properties of the planets and their masses to calculate approximate values. In found that the planetary specific angular momenta ranged from between about 1/100 and 1/10 of the typical angular momenta that I observed in the binary system formation modeled by the Dr. Fisher's simulations. I also estimated the Sun's total angular momentum was only a few percent of the total angular momentum in the solar system. This suggested that much of a core's angular momentum ends up in companions, such as planets, during stellar-companion system formation, a point that Dr. Fisher wanted me to convince myself of.

3.27 Attempts to Use a Debugger

To assist my efforts to module test the code, Dr. Fisher advised me to use a debugger. Although I had previous experience with a debugger, the debugger I had used was a built-in feature in Visual Studio 2008 with a graphical user interface. In contrast, the debugger that I would now be using operated on the command line; thus I had no experience with this form of debugger. When I initially tried to operate the debugger, I failed, so I consulted online resources and asked other students for help during the CSUMS class. Nonetheless, I have remained unsuccessful. I plan to ask Dr. Fisher about operating the debugger so that I may start using it effectively.

4 Progress

Unfortunately, I did not achieve results on the actual research problem this semester. While I identified that there was some problem in the simulation code, I have not yet succeeded in identifying the source of the error. However, I have significantly improved my knowledge of the research problem and have had an opportunity to practice my programming and debugging skills.

5 Reflection on Learning

During the project, I learned more about astrophysics, particularly in regards to star formation and angular momentum. My discussions with Dr. Fisher through the project provided me one-on-one instruction, including stimulating thought exercises, and were valued experiences. Additionally, the scientific papers that I read were very interesting and informative, both enhancing my understanding of the project and exposing me to additional ideas that I would not have directly used in the project.

Furthermore, I increased my programming skills during the project. In addition to practicing my existing techniques, I learned several new commands that may be helpful in the future. Additionally, I began to learn how to teach others FORTRAN when I had to teach Deivid its basics.

6 Comment and Criticisms about CSUMS Class

6.1 Best Feature

The best feature of the CSUMS class was the focus that provided to my research. While I had been productive during my summer research experience, I found that I lost focus during the fall semester, in which I was not actively involved in the CSUMS program. When I started the CSUMS class, I made my research a priority again and once again became productive. While research time still needed to compete with my other studies, I became highly motivated to set aside time for my research. Although unfortunately I did not accomplish much this semester, my drive has been renewed and I am excited to continue my research into the summer.

6.2 Worst Feature

The worst feature of the CSUMS class was that most class days were largely dedicated to listening to presentations. While I understand that this was an important component of the class and that conducting research was homework, I would have liked a greater opportunity to conduct research during class. I feel that this would have significantly improved my productivity by allowing me use of a highly regular, dedicated research time, in contrast to having to find time outside of class to work on research in addition to my other homework. Additionally, I would have a greater opportunity to help and be helped by other students in class, which was hampered by the number of presentations.

7 Acknowledgements

I would like to extend my sincerest thanks to Dr. Fisher for his crucial support as my research advisor. I also want to thank Dr. Gottlieb and Dr. Kim for their efforts in running the CSUMS class, Deivid Riberio for his collaboration with me, and Dr. Zarrillo for providing a template for and arranging the printing of my poster.

References

[1] Burkert, Andreas and Peter Bodenheimer. *Turbulent Molecular Cloud Cores: Rotational Properties.* The Astrophysical Journal, 543:822-830, 10 November 2000. Print.

[2] Chabrier, Gilles. *Galatic Stellar and Substellar Initial Mass Function*, The Astronomical Society of the Pacific, 115:793-95, July 2003. Print.

[3] Fisher, Robert T. *A Turbulent Interstellar Medium Origin of the Binary Period Distribution.* The Astrophysical Journal, 600:769-780, 10 Jan 2004. Print.

[4] Fisher, Robert T. *Personal Interview*, 21 Feb. 2011.

[5] Grether, Daniel and Charles H. Lineweaver. *How Dry is the Brown Dwarf Desert? Quantifying the Relative Number of Planets, Brown Dwarfs, and Stellar Companions Around Neabry Sun-Like Stars/* The Astrophysical Journal, 640:1051-1062, 1 April 2006. Print.