MATH 198 Project Preproposal

Professor George Francis
Brian Campbell-Deem

October 5$^{\text{th}}$, 2015

1 Intro

Being a physics major, it is natural that I have an intent interest in physics. I signed up for this course hoping to be able to learn methods of exploring this interest in a way that would not only better my rudimentary knowledge of CS, but also that of the subject matter itself. To this effect, I want to create a project that revolves entirely around physics; physics is inherently both mathematical and physical, so it should be extremely straightforward to find a balance between the math and graphical requisites.

2 (Pre)-Proposal

I am still unsure of one area to focus on, so my current idea ignores that: like the Illyes Turtle project, I plan to make a "hub" of various physics simulations, the contents of which I will go into detail below. I like the idea of doing a lot of smaller 'projects' amalgamated into one big "catch-all physics program", but this idea has other advantages; because it is an unspecified number of different programs put into one, it will be easy to add or subtract parts should I have extra time or not enough time. Additionally, each planned piece is simple on a basic level but allows deep exploration (as many problems do) which means I can go from many basic programs to a few deeper ones, and possibly even settle on just one that ends up interesting me the most; in this sense, my project proposal is dynamic.
One thing I want to stress in my project is user-interactivity and input, because the best way to understand physics is to work with it. To this end, I want every subprogram to have some form of input or manipulation which would dynamically evolve the system and allow the user to gain intuition on how it affects the output.

3 Constituent Programs

3.1 Wavefunctions

I am deeply interested in Quantum Mechanics (QM) and am currently enrolled in PHYS 486. The wavefunction of a particle describes everything there is to know about the system, and QM itself is even by physics' standard very mathematical. There are plenty of different ways to approach this: I could make a program that graphs various wavefunctions at different energy levels in an infinite square well in 1-D, and then extrapolate later to 3-D; I could follow the same approach for other various common potentials, such as the harmonic oscillator and the finite square well (in 1, 2, or 3 dimensions depending on difficulty). With the wavefunctions calculated, I can also give various statistics: $< \hat{x} >$, $< \hat{p} >$, $< \hat{H} >$ and so on, which will become especially interesting should I allow the user to superimpose multiple stationary states onto each other (creating much more dynamic systems).

Once understood, the math would be fairly straightforward once the user enters values and I can avoid an extremely complex program while still retaining and outward look of complexity.

3.2 Chaos

The double-pendulum is a classic example we've already seen, however there are many types of chaotic systems that have simple analyses; for example, I was shown in PHYS 194 various recursive functions that give chaotic behavior, which we used in MATLAB to simulate growth of trees; similar ideas could follow for this class, obvioously adapted to a different programming language. An analysis of the double-pendulum would be straightforward with the Lagrangian approach of solving a system's behavior, and I could evolve it into a triple-pendulum, quadruple pendulum, and so on until I can simulate an n-pendula system. The lengths and masses of the pendula would (according to plan) be entered by the user.

3.3 Thermal and Statistical Mechanics

I'm currently enrolled in PHYS 427 and the systems we are studying are rich in possibility for graphical modelling. As we've seen in class, modeling an atom's collisions and then adding another, and another, and so on until we reach N atoms is a possbility. Possibilities for the user input would be temperature, volume, and so on which we can display with the ideal gas law, creating a very dynamic display. We can also run semi-random/probabilistic situations such as magnetization of an object at temperature T immersed in a magnetic field B. Using the canonical ensemble (Boltzmann statistics) we can show how various arrangments of $1..N$ magnetic moments would behave given T and B, which would also be able to be entered by the user.

3.4 Other

Of course, there are an infinite number of possible systems we could model, so we have freedom for what the other subproject pieces could be. The three areas I've listed above are the ones I've considered the most, however basic mechanical systems such as 2-D kinematics motion, various spring and oscillation systems, and so on are possible. They can become more complex when we consider situations I've encountered in my PHYS 325/326 classes, such as drag forces, coupled oscillators, and so on whose behaviors are not only more interesting, but more realistic. I've also not touched the E&M side of things, such as displayed electric fields and potentials for various charge configurations, circuit analysis, and so on. The "other" section is entirely dependent on how deep I go with the first three and how well I'm able to implement them.

4 Conclusion

As stated in the intro, my current idea is extremely flexible; at some point, I will obviously have to refine it more so as to be able to fulfill my contract, but as it stands there is a lot of room for how complex or simple the program is as a whole, which allows me many directions depending on if I want many simple pieces, a few complex pieces, and so on. I believe this current approach will be fruitful any way I end up doing it exactly because of its dynamicity, and it will be able to very powerfully display the mathematics in a visual manner, especially if it is as user-controlled as I am planning it to be.