Modeling the Hydrogen Atom

New Mexico
Supercomputing Challenge
Final Report
April 2, 2008

Team #83
Sandia Preparatory School

Team Members
Alessandra Anderson
Kaitlyn Hughes

Teacher
Neil McBeth

Project Mentor
Robert Anderson

.

Executive Summary:

Our project is modeling a hydrogen atom. Because no one knows the exact shape or properties of an atom, we used the most accepted atomic theories in our model. These two theories include the Quantum Mechanics theory and the planetary theory otherwise known as Bohr's Atomic Model. The planetary model represents a hydrogen atom as a mini solar system. The proton and neutron are at the center while the electron circles it at a steady orbit of .0593 nanometers. The Quantum Mechanics theory states that the electron acts both as a particle and a wave and doesn't exist in a particular space at a given time. Instead it jumps, and although mostly stays at a certain radius, can move in and out thus creating a probability cloud on where it can be.

The first model we created was solely based on the planetary model. We scaled the electron to be 1/1836 of the proton and made it rotate around the 3-d proton. This was our introduction into C#, the XNA framework, and matrix multiplication. From there we explored the Quantum model. By doing some research we were able to find a probability curve for the radius of the electron. Although the electron stays mostly at a radius of .0593 nanometers it jumps to different levels of energy or an orbital. The probability curve represents how often an electron can be found at a certain radius. By learning how to integrate by parts and testing various points using MatLab we were able to implement the curve. After adding in the law of conservation of angular momentum to the electron we finished our model. We now can represent both a basic Bohr model as well as show an example of quantum mechanics.

Statement of Problem:

To model a hydrogen atom in accordance with both the Bohr Atomic Model and Quantum Mechanics.

Description of method:

Our project started by researching atomic models. We found the two main atomic models, Bohr's Atomic Model and the Quantum model. Since C# is a new language for both of us we started out with the more simplistic Bohr's model. This model only involved creating two objects, one stationary sphere in the center to serve as the proton and neutron, and another to rotate around the center sphere. To do this we used C#'s XNA framework. Also we created our own time so that we could control the speed of the electron by pressing either "F" to make it speed up or "S" to make it slow down. Furthermore we added two more key commands allowing us to zoom in by pressing "Z" and to zoom out by pressing "X". Since we wanted to make our model 3-D we had to learn how to multiply matrices, so after we had one object rotating around the other we began to play with the radius, multiplying it by sin waves and seeing what patterns we could get. This simply served as a base for our understanding, for the math involved was new to us. From there, our team did more research. We discovered the actual size of the electron compared to the proton in nanometers is 1/1836. Also we discovered that in the Bohr's model the electron orbits steadily at a radius of 0.0529 nanometers. We implemented both of these into our program.

We then moved to the quantum model. In order to this it also required more research. We realized almost right away that some of the math involved was beyond our capabilities, so we focused more on the representation. At first we experimented with a random number generator to

predict where the electron ought to be, but then we decided that we would like it to be more accurate, so through research we found the probability curve for the electrons radius which is :

$$\langle r \rangle = \int_0^\infty r \frac{dP}{dr} dr = \frac{4}{a_0^3} \int_0^\infty r^3 e^{-2r/a_0} dr$$

To solve this equation we had to learn how to do integration in parts, but with help we were able to pick up on it quickly. After we had learned the math, we set about implementing it. First off we changed our electron orbit to once again only rotate about the x-axis instead of multiple ones. We then made it so we could cancel the draw update routine by hitting the letter "N" in our program so we could visualize a "probability cloud." From there we used MatLab to test points on the curve and to make sure we could implement it properly. After we had tested it, we implemented it in our program. Next we had to implement the law of conservation of angular momentum so that our electron would speed up and slow down depending how large or small its radius was. Last we rotated our axes around one another causing the electron to travel in a spherical shape forming the proper electron cloud. In order to do this we went back to our Matrix multiplication.

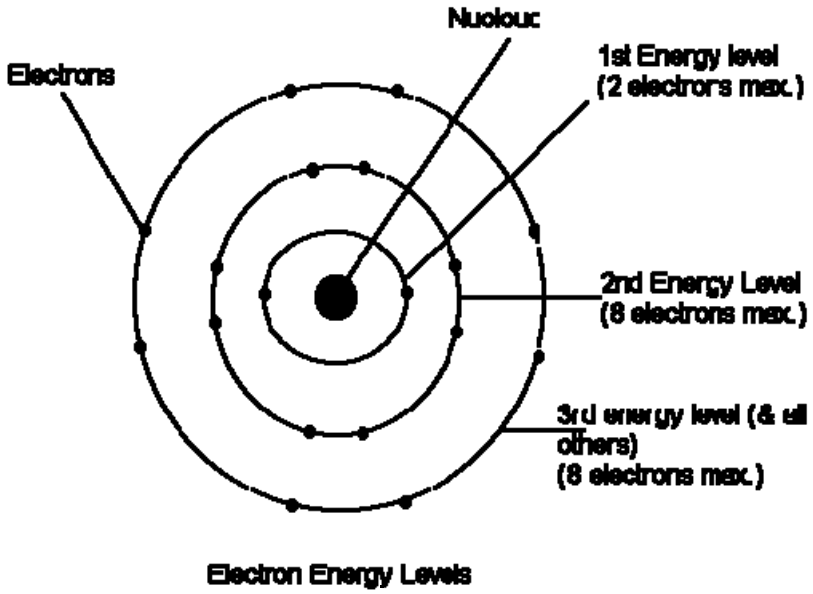
Results and Conclusion of study:

Our model was not intended to solve a mathematical or scientific problem; however, we feel we have accomplished our goal. We have created a scientifically accurate, yet visual and understandable model. The model is scaled to be proportionate to the nanometer equivalent in the atomic world. The electron jumps to different orbital's as it would in a real atom following a probability curve for the radius of hydrogen's electron; also our electron gains and losses velocity due to the law of conservation of angular momentum. By holding down the "N" key one can represent the electron "cloud" as in the quantum mechanics model, and by letting it

alone one can easily explain the Bohr's model. This model overall becomes an easily explainable representation for both the Bohr's atomic model and the quantum mechanics theory model.

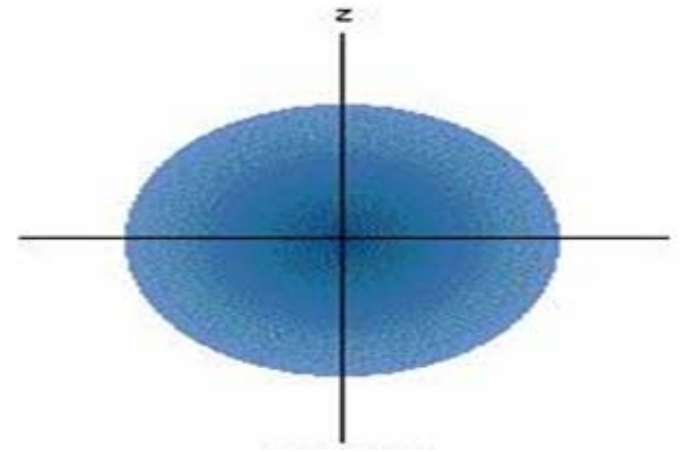
References and Tables:

Bohr's Model



Electron Energy Levels

Quantum Model



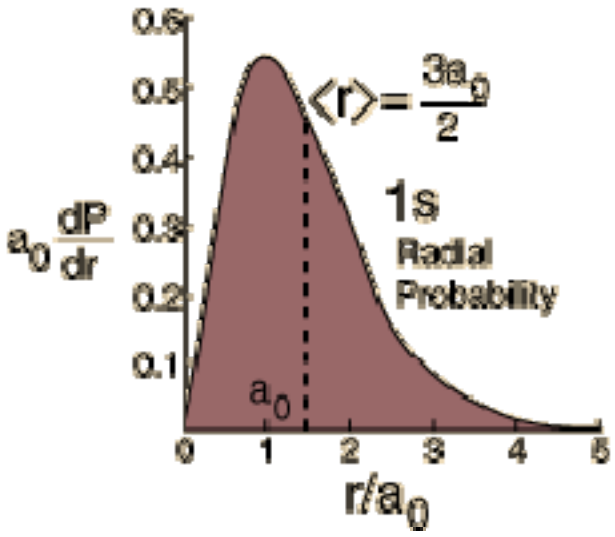
Matrix's

$$\mathbf{A} = \begin{bmatrix} a_{1,1} & a_{1,2} & a_{1,3} & \dots \\ a_{2,1} & a_{2,2} & a_{2,3} & \dots \\ a_{3,1} & a_{3,2} & a_{3,3} & \dots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} = \begin{bmatrix} A_1 \\ A_2 \\ A_3 \\ \vdots \end{bmatrix}$$

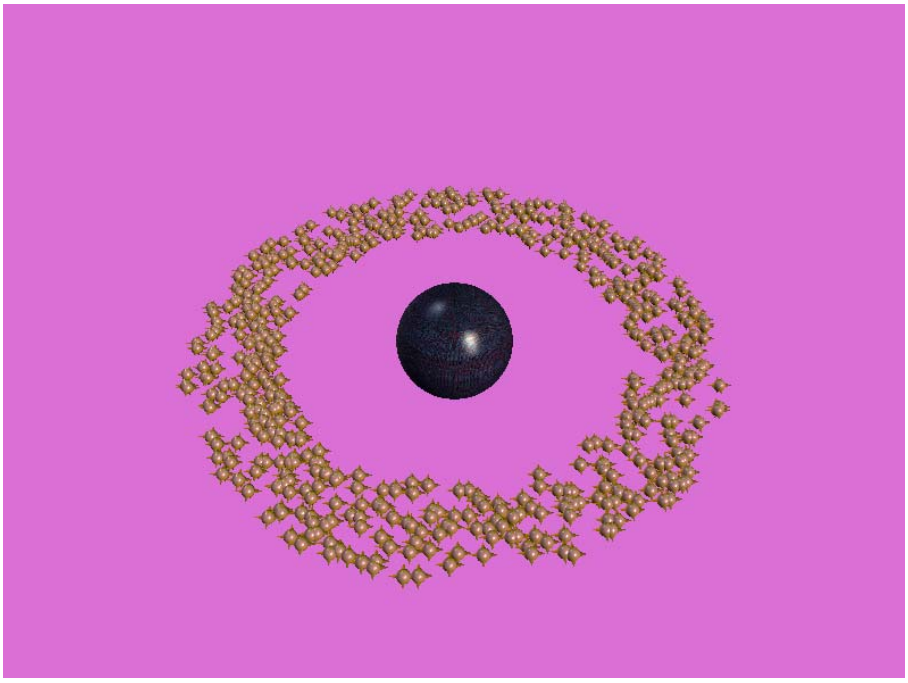
Integration by Parts

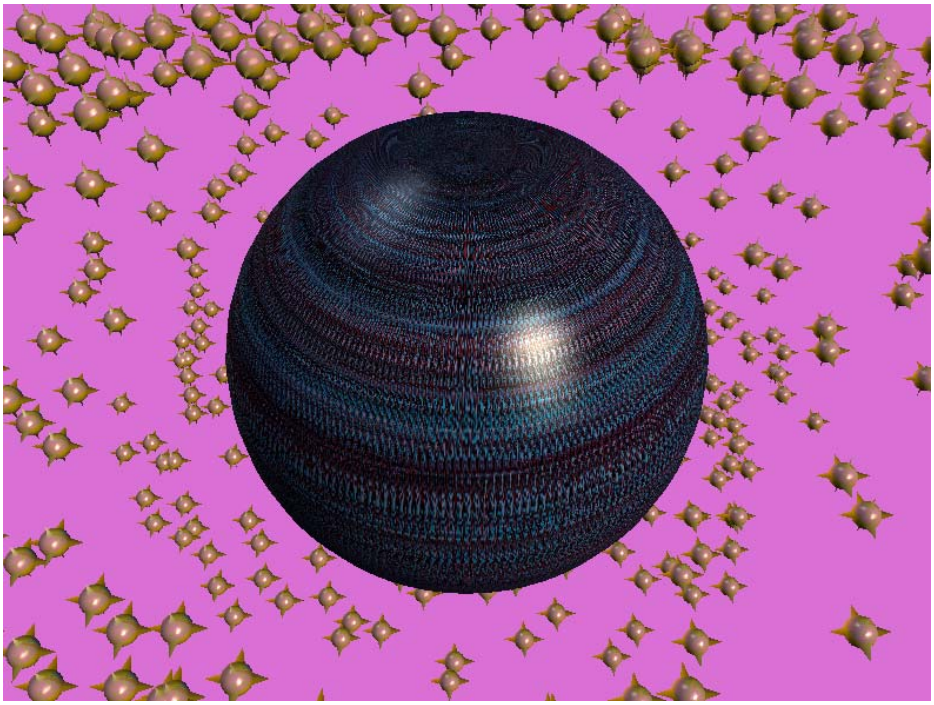
$$\langle r \rangle = \frac{4}{a_0^3}\left[e^{-2r/a_0}\left(\frac{-a_0 r^3}{2} - \frac{3a_0^2 r^2}{4} - \frac{3a_0^3 r}{4} - \frac{3a_0^4}{8}\right)\right]_{r=0}^{r\to\infty}$$

Probability graph



Examples of XNA and Electron Cloud





Most significant achievement:

Our entire project was both new, challenging, and considered are significant achievement. Not only did we manage to program an accurate hydrogen atom using the Quantum mechanics theory as well as Bohr's Model, we also learned how to program with a new language, C#, work with the XNA framework, and challenge our mathematical intellect by using matrix multiplication, random number generators, probability curves, and integration by parts. Overall we put a lot of work into this project and learned how difficult something so basic can be. Therefore we find this entire project was a significant achievement.

Acknowledgment:

Mr. Anderson, thank you for all the help. We could not have gotten through either the programming or the mathematics without you. You helped keep our project alive, and were always willing to help. Thanks again

Mr. McBeth, thank you for sponsoring this great program. This is one of the most challenging and rewarding experiences we have participated in during high school.

Ms Jones, thanks for all this information you provided us. It was very helpful and led us to the models we have created.

Mr. Suding, thanks for your help in pointing our group in the right direction and giving us a good place to start with references and suggestions.