Numerical Modeling of Ellipsoidal and Spherical Monomers: Coagulation and Proof of Concept for Next Generation Nano-Magnets

Robert D. Fisher, Lorin S. Matthews, Jonathan Perry

Abstract—**Numerical modeling of the creation of nanocomposite magnets allows for an in depth study into most efficient ways to create a nanomagnet. Since these magnets utilize two separate materials, it is ideal to create a model made of spheres and ellipsoids, each with two separate physical characteristics, as a proof of concept for next generation nanomagnetic materials. This paper examines the coagulation of spherical and ellipsoidal monomers as a means of later producing a nanocomposite magnet made of two separate materials.**

Index Terms—**Energy Product, Isotropic**

I. INTRODUCTION

Nanocomposite magnets represent a likely candidate in the development of next generation magnets. Current generation high-performance magnets are most commonly made from neodymium, a rare-earth element that is mainly found in China. As a result of reduced Chinese exports, new efforts are being made to create a more nonconventional nanomagnet [1].

One way scientists measure the efficacy in magnets is to study their energy product, or the amount of energy stored per unit area in a magnet. Current Neodymium magnets have a maximum theoretical energy product of only about 64 millions of gauss oersteds (MGOe) or about 509,296 J/m^3, and experimental results have produced results that are around 92% of this value [1]. Consequentially, not much improvement can be made to existing neodymium magnets.

Numerous methods have been made to produce these magnets [2,3]. The basic principle behind their creation and structure is virtually the same in all techniques. Physicist and material scientists start with two fundamental building blocks: a magnetically hard substance and a strongly magnetized substance. The hard substance is a permanent magnet. Its crystalline structure is isotropic, so its magnetic moment can easily line up with a magnetic field. The strongly magnetized substance is a material that is not naturally magnetic, but can be easily magnetized in the presence of a magnetic field. Once scientists have their two materials, they align them and mix them together. Once combined, they can apply heat and pressure to compact the material into a new composite magnet. This nanostructuring from the bottom up approach has been shown in the assembly of Sm-Co and Fe composite magnets [4].

The energy product of nanocomposite magnets has been shown to be much more powerful than what normally exists in plain neodymium magnets. Researchers have shown that an iron-cobalt mixture stabilized with a samarium-iron-nitrogen alloy could produce an energy product of up to 137 MGOe [1]. The applications that nanocomposite magnets can be used in depend on the sizes, shapes, and chemical compositions of the grains used in their production [5].

II. METHODS

In order to model the formation of a nanocomposite magnet, the numerical modeling simulation called Aggregate Builder is run. Aggregate Builder works by building an aggregate from the bottom up. Earlier versions of the code created an aggregate made up of either just spheres or ellipsoids. Since the numerical model was never designed to create an aggregate made up of both spheres and ellipsoids, Aggregate Builder had to be modified.

In order to simulate the build up of an aggregate from a bottom up approach, the current numerical model starts off by creating a single seed monomer, which can either be a sphere or an ellipsoid, depending on what the randomization function built into the model chooses. From there, Aggregate Builder chooses another monomer, which can also be either a sphere or an ellipsoid. The new monomer is trajected towards the seed particle from a random direction. If the seed particle and the new monomer overlap at the points of collision, Aggregate Builder detects a collision and a new aggregate is formed. If not, then the program detects a miss. This process is repeated until a desired *N* number of monomers have been used within the aggregate building process.

Aggregates can be formed under a variety of conditions, each affecting the final outcome of the aggregate to be built. Variables such as mass and charge will have different ways of affecting the evolution and structure of the final aggregate. An example of an aggregate

Manuscript received August 2012. This research was supported by the National Science Foundation: Research Experience for Undergraduates: Grant No. PHY-1002637

made of both spheres and oblate ellipsoids is shown in Figure 1.

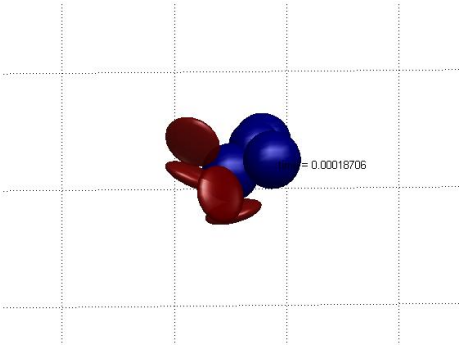

Figure 1

III. AGGREGTE BUILDING

Previous versions of Aggregate Builder could only model aggregates made up of either just spheres or just ellipsoids. In order to create a more accurate model of the creation of a nanocomposite magnet made from a bottom up approach, aggregate builder had to be modified in order to create a magnet made up of both spheres and ellipsoids. To do this, aggregate builder needed to have two separate sets of parameters for each shape, thus allowing the simulation of two different materials coagulating. For the spheres, not only could the density and magnetization be adjusted, but also the radius, which in turn affects the dynamics of rotation within the aggregate and also the mass of each sphere. Each ellipsoid also has its own density and magnetization, but unlike the sphere the separate axes have to be taken into account in order to model the spin and moment of inertia of the entire aggregate once the monomers had collided. Since the ellipsoid's semi-major and semi-minor axis could be changed, we could set the shape of the ellipsoid to have either a prolate shape or an oblate shape as shown in Figure 2. From the given parameters, Aggregate Builder could calculate the volume and mass from the parameters associated with each spherical or ellipsoidal monomer:

$$V_{Sphere} = \frac{4}{3}\pi r^3 \qquad V_{Ellipsoid} = \frac{4}{3}\pi abc$$

$$M_{Sphere} = \rho_{Sphere} \times \frac{4}{3}\pi r^3 \qquad M_{Ellipsoid} = \rho_{Ellipsoid} \times \frac{4}{3}\pi abc$$

Connecting monomers into a single aggregate in Aggregate Builder involves what are called "Collision Points" on the surface of the monomers and aggregates. These points are evenly spaced on the surface, as shown in Figure 3. With spheres, it would be enough to test a collision by setting up a vector that goes from the center of one spherical monomer to the center of another spherical monomer. If the magnitude of the vector is less than or equal to the radius of one monomer plus the radius of another monomer, then the two particles have collided. Of course, since we are dealing with a sphere and an ellipsoid colliding, this method cannot be used since the "radius" of an ellipsoid varies. Instead, we

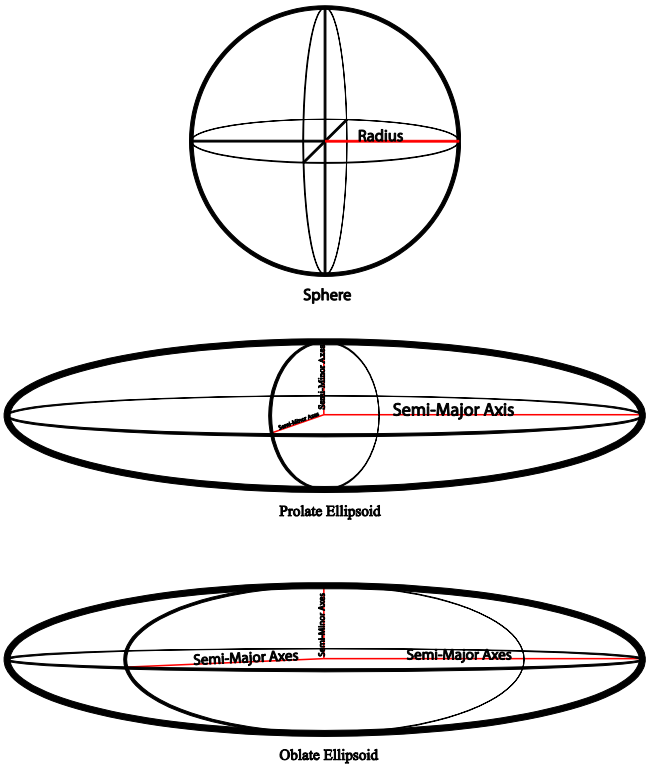

Figure 2

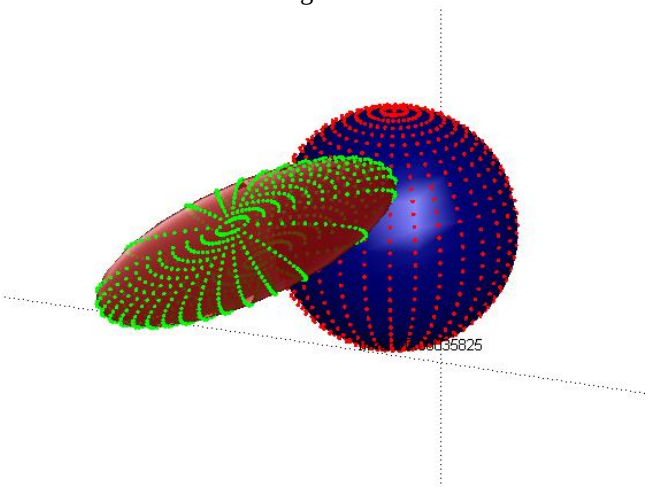

Figure 3

have to use a method that treats the spherical monomers like ellipsoids by setting each axis of the sphere equal to the radius of the sphere. From there, Aggregate Builder sets a variable equal to the three axes of the seed monomer and incoming monomer. So, the axis of the seed particle is saved separately from the axes of the incoming monomer. Aggregate Builder then uses the equation,

$$V = \left(\frac{x}{a}\right)^2 + \left(\frac{y}{b}\right)^2 + \left(\frac{z}{c}\right)$$

to test whether or not the sphere and ellipsoid has overlapped or connected. It does this by finding the *x*, *y*, and *z* positions of the collision points on the surface of the seed and incoming monomer. Then it finds the axes of the sphere and ellipsoid monomer. Aggregate Builder runs two tests, one for each monomer. The first test uses the equation from above and inputs the values of the axes for the seed particle into the *a*, *b*, and *c* variables. It then takes the *x*, *y*, and *z* positions of the incoming monomer's collision points. If the equation is less than or preferably equal to one, then the particles have collided. The second test utilizes a very similar method, but instead it uses the values of the axes for the incoming monomer as the *a*, *b*, and *c* variables, and uses the *x*, *y*, and *z* positions of the seed particle's collision points as the *a*, *b*, and *c* variables. If the parameters satisfy the equation above, then the particles have collided and a new aggregate structure is created. If the equation is not satisfied, then the incoming particle is destroyed, and a new incoming monomer is trajected towards the seed aggregate.

Once the particles have collided, Aggregate Builder can work on combining all of the characteristics of each monomer into a single aggregate. When combining two monomers together, Aggregate Builder easily finds the total mass and charge of the aggregate by summing all of the masses and charges of the connected particles together. However, the orientation of each monomer within the aggregate must be within the same set of axes in order to calculated calculate the spin of the entire aggregate. An illustration of this is shown in Figure 4.

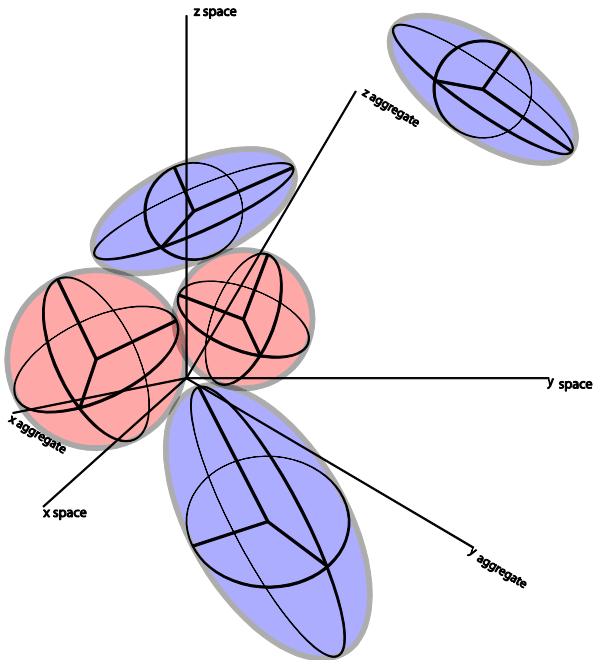

Figure 4

In order to have all of the monomers within the same axes, a rotation matrix (*R*) is used to rotate the incoming monomer's axes (*T*) into the aggregate's axes (T'),

$$T' = RT$$

The rotation matrix is then applied again in order to get the orientation of the entire aggregate into a uniform space axes. Each axes is centered at the center of mass of the monomer/aggregate. Since spheres have symmetry across allow diagonals through the center of mass, they do not have to be rotated.

When taking into account the rotational dynamics of the aggregate, the moment of inertia tensor is taken into used. For spheres, the tensor is the same across the diagonal:

$$I_{Sphere} = \begin{bmatrix} \frac{2}{5}MR^2 & 0 & 0 \\ 0 & \frac{2}{5}MR^2 & 0 \\ 0 & 0 & \frac{2}{5}MR^2 \end{bmatrix}$$

For ellipsoids, the tensor is:

$$I_{Ellipsoid} = \begin{bmatrix} \frac{1}{5}M(b^2+c^2) & 0 & 0 \\ 0 & \frac{1}{5}M(a^2+c^2) & 0 \\ 0 & 0 & \frac{1}{5}M(a^2+b^2) \end{bmatrix}$$

Since the monomers are connecting, the moment of inertia tensor also needs to be rotated in order to be within the entire aggregate body's axes. In a similar process to the axes rotation, the rotation tensor is used make the transformation from the monomer body to the aggregate body axes:

$$I' = RIR^T$$

Since the rotation is being applied to a tensor, the transposed matrix of the rotation tensor must be multiplied on the left side in order to make a proper rotation. This process is similar to the outcome of the diagonalization of a matrix. In order for the moment of inertia tensor to be physically possible, the inertia tensor in all cases must be Hermitian.

IV. CODING TECHNIQUES

A randomization function was added to the code that would initialize a seed and incoming monomer of either a sphere or an ellipsoid.

Since the Aggregate Builder model was only designed to plot just spheres or just ellipsoids, the new Aggregate Builder code had to be designed in order to take into account the combination of both shapes. To do this, the aggregate shapes were set up to correspond to an array of arbitrary but consistent numbers, in this case, zeroes and ones:

[0 1 0 0 0 1 1 1 0 0 0 1 0 1 0 1 0 1 1 1]

In order for Aggregate Builder to properly run a spherical or ellipsoidal model, a loop had to be added that would only act on spheres if a 0 was hit, while every 1 corresponding to an ellipsoid was ignored. The same is true for ellipsoids corresponding to a 1. As a result the plotting and rotation operations could be coded into the model, while at the same time taking into account the separate shapes of each monomer.

V. FUTURE WORK

Although much of the coding is complete, an actual nanocomposite magnet model still needs to be run. Ideally, a Samarium-Cobalt-Iron nanomagnet is to be created based off of experimental results [2]. Once run to the appropriate size, aggregate builder should be able to detect results such as the number of hits and misses between monomers along with the total charge on the final aggregate.

Along with running a simulation, additional coding could be implemented that would implement the composite magnet traveling through a fluid field, thus simulating the effects of real life drag on an object. The model would require Aggregate Builder to calculate the flux of the field around the monomer. As mentioned before, the orientation of the spherical monomers would not matter, but the orientation of the ellipsoidal monomers would, thus producing a drastic change in the way the monomer and entire aggregate evolves.

VI. ACKNOWLEDGEMENTS

Special thanks to Dr. Lorin S. Matthews and Jonathan Perry for helping me with the coding. Thank you to the numerical modeling group for helpful discussions on Aggregate Builder. Finally, thank you to Baylor University, CASPER, and the NSF for helping fund this project.

III. REFERENCES

[1] G. Hadjipanayis, A. Gabay, "The Incredible Pull of Nanocomposite Magnets," *IEEE* Spectr., Aug. 2011.

[2] M. Huang, L. Zhang, B. Ma, Y. Zheng, J. Elbicki, W. Wallace, S. Sankar, "Metal-bonded Sm_2Fe_{17}-N-type magnets," Journal of Applied Physics, vol. 70, pp. 6027, 1991.

[3] G. Hadjipanayis, L. Withanawasan, R. Krause, "Nanaocomposite $R_2Fe_{14}B/\alpha$-Fe Permanent Magnets," IEEE Trans. Magn., vol. 31, no. 6, pp. 3596-3601, Nov. 1995.

[4] M. Sachan, S. Majetich, "Combustion-Driven Compaction of Nanostructured Sm-Co and Fe Mixtures," IEEE Trans. Magn., vol. 41, no. 10, pp. 3874-3876, Oct. 2005.

[5] A. G. Roca, R. Costo, A. F. Rebolledo, S. Veintemillas-Verdaguer, P. Tartaj, T. González-Carren□o, M P Morales and C J Serna "Progress in the preperation of nanoparticles for applications in biomedicine," *J. Phys. D: Appl. Phys.*, vol. 42, pp. 11, 2009.

Robert D. Fisher is a senior at Texas A&M University–Kingsville. He is currently working on his Bachelor of Science in Physics with a minor in Mathematics and plans on graduating in May 2013.