Turbulent Transport of Dust in the Protoplanetary Disk

Ward S. Howard, Lorin S. Matthews, and Peter J. Fager

***Abstract*— Dust and gas revolve around a young star in what is known as the protoplanetary disk. Dust grain collisions result in the sticking of the grains into loose, "fluffy aggregates." These aggregates, thought to be the fundamental building blocks of planets, grow through further collisions and sticking into larger aggregates. There exist multiple barriers to sticking that slow the growth rate. One such barrier results from the plasma environment of the disk. Dust grains gain a negative charge due to collisions with electrons in the plasma, causing the grains to repel each other when they interact, rather than colliding. This decreases the likelihood of collisional sticking and aggregation.**

Gas in the disk experiences turbulent flow. This turbulence affects the velocities of the grains in turbulent eddies, potentially giving them great enough relative velocities to overcome electric repulsion and collide. We explore the effect of disk turbulence on the sticking rate of these dust collisions by employing a simulated turbulent gas flow into a numerical model of dust interactions in the disk. The model of the gas drag forces allows smooth integration of particle position and velocity, successfully modeling grain collisions under turbulence. Grain velocities demonstrate dependence on grain radius. Further work is needed to resolve scaling problems in the model and to observe the effect of turbulence on the sticking rate.

***Index Terms*— protoplanetary disk, fractal aggregate, planet formation, coagulation, turbulence, tree code, gas drag**

I. Introduction

ASTRONOMICAL observations of young T Tauri stars and theoretical modeling of developing solar systems reveals the presence of a swirling cloud of gas and dust orbiting the stars in the protoplanetary disk [1]. It is thought that planet formation begins with the creation of aggregates out of micron-scale dust grains [2]. Because aggregates result from sticking during collisions of dust grains, they develop fractal-like shapes with low density. These "fluffy aggregates" then compact and grow with further collisions, leading to larger and larger objects up to planetesimals [3][4]. Time constraints on the formation of planetesimals make it necessary to consider barriers to aggregate growth, as well as processes aiding growth [1]. One such barrier is grain charging. Atoms in the disk are often dissociated due to the radiative environment of space [4], leading to a charged plasma environment. In the plasma environment of the protoplanetary disk, dust grains gain a negative charge through collisions with electrons, which are more frequent than collisions with slow-moving ions [4]. This charging of the grains slows the aggregate growth rate, as many grains that would otherwise have collided with each other no longer have the necessary relative velocities to overcome grain-grain electric repulsion [1].

Gas and dust in the disk experience turbulence due to Magneto-Rotational Instability, which has an unknown effect on the aggregate growth rate. MRI describes the process by which weak magnetic fields in the disk exert a Lorentz force on gas rotating differentially in the radial direction from the central star, creating significant turbulence [5]. Collisions of dust in such turbulence might lead to better grain sticking, increased rates of aggregate shattering, or a general slowing of the aggregate process [1].

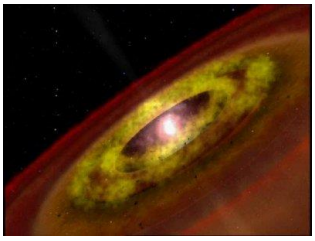

Fig. 1. Artist's conception of a protoplanetary disk of gas and dust orbiting a T Tauri star. Image courtesy of NASA.

II. Theory

In order to accurately model dust grain velocities in turbulence, both a description of turbulent gas flow in the protoplanetary disk and the frictional gas drag exerted on dust grains within that flow are necessary.

A. Disk Turbulence

Turbulent gas flow exists on a variety of scales, since the input energy of a turbulent system gets passed down through smaller and smaller turbulent eddies all the way down to the molecular level, where the molecular viscosity damps out the remaining turbulence. The scale at which this damping occurs is known as the Kolmogorov scale. While the Kolmogorov scale defines the smallest turbulent eddies, the integral scale defines the largest eddies. All turbulent flows are described by these related parameters [6], including the turbulence of the protoplanetary disk.

The largest eddy scale, the integral scale, is given by the

Manuscript received August 4, 2014. This paper is second in series on dust behavior in turbulent protoplanetary disks relating to planet formation. See P. J. Fager, "The Effects of Turbulence on Dust Particle Coagulation in Planetary Formation." This work is supported by the National Science Foundation under Grant No. 1262031.

W. S. Howard is an undergraduate math and physics double-major at Union University, Jackson, TN 38305 USA (e-mail: graphos20@gmail.com).

L. S. Matthews, PhD. is associate director of CASPER at Baylor University, Waco, TX 76798 USA (e-mail: Lorin_Matthews@baylor.edu).

P. J. Fager graduated in 2014 with a B.S. in Mechanical Engineering, Baylor University, Waco, TX 76798 USA (e-mail: peter_fager@baylor.edu).

length L, gas velocity v_L, and turnover time t_L, the time required for the gas to cycle around the eddy one time.

$$L = \frac{\sqrt{\alpha}\, c_g}{\Omega_K} = v_L t_L \quad (1)$$
$$v_L = \sqrt{\alpha}\, c_g$$
$$t_L = \frac{1}{\Omega_K}$$

Here Ω_K is the Keplerian orbital frequency, α is the turbulence strength parameter, and c_g is the speed of sound in the gas, given by

$$c_g = \sqrt{\frac{\gamma k_B T}{\mu m_H}} \quad (2)$$

where γ is a degree-of-freedom parameter of the gas known as the heat capacity ratio, k_B is Boltzmann's constant, T is gas temperature, m_H is the mass of a hydrogen atom, and μ is the mean molecular weight, accounting for the helium and other gasses in the disk [1][7].

The smallest scale, the Kolmogorov scale, is given by the length η, turnover time t_η, and gas velocity v_η.

$$\eta = \left(\frac{1}{Re}\right)^{3/4} L \quad (3)$$
$$v_\eta = \sqrt[4]{\frac{1}{Re}}\, v_L$$
$$t_\eta = \sqrt{\frac{1}{Re}}\, t_L$$

where Re is the Reynolds number, and is given by

$$Re = \frac{\alpha {c_g}^2}{\nu \Omega_K} \quad (4)$$

and the molecular viscosity is defined as

$$\nu = \sqrt{\frac{2}{\pi}} \frac{c_g \mu m_H}{\rho_{gas} \sigma_{mol}} \quad (5)$$

where σ_{mol} is the collisional cross-section of the gas molecules [1].

B. Stokes Gas Drag

The dust grains entrained in a turbulent gas eddy will experience a frictional drag force and eventually come to rest with respect to the gas. The amount of time it takes for this to occur is known as the frictional time, or stopping time, t_s

$$t_s = \frac{3 m_d}{4 c_g \rho_g \sigma_d} \quad (6)$$

where m_d is the mass of the dust grain, σ_d is the collision cross-section area of the grain, and ρ_g is the gas density [8].

Note that for a spherical grain of constant density, the stopping time is proportional to the grain radius. Larger grains are less well-coupled to the gas drag due to the difference in stopping times, leading to a difference in relative velocities between grains of different radii, causing greater compaction to occur in aggregate collisions. It is unknown whether this decoupling trend of larger particles from the gas increases coagulation or fragmentation of aggregates [3][8].

The frictional drag on very small particles is proportional to the dust velocity with respect to the gas, and is described by Stokes Law [9].

$$\boldsymbol{F}_{Stokes} = -6\pi\eta r_{grain}(\boldsymbol{v}_{grain} - \boldsymbol{v}_{gas}) \quad (7)$$

where r_{grain} is the radius of the grain, $\boldsymbol{v}_{grain}$ is the velocity of the grain, and $\boldsymbol{v}_{gas}$ is the turbulent gas velocity. The absolute viscosity is $\eta = \rho_g \nu$ [1].

C. Numerical Model

The numerical model for the disk environment is the box_tree code [10], an N-body code which uses a tree code to reduce the number of direct force calculations from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$. Box code handles boundary conditions with ghost particles and uses symmetry arguments to allow a representative box of the N-body disk to stand in for the whole, while tree code divides the simulation cell of grain masses into a tree-like structure of sub-volume cells within cells until each mass occupies its own sub-cell. Then a multipole expansion of the force about the centers of mass of small or distant cells is done for each test grain [4][10].

In order to use box_tree to analyze the effects of turbulence on aggregate growth, it was modified to interpolate the turbulent gas velocities at any point inside the simulation using a database of velocities from a numerically simulated turbulent flow. These interpolated gas velocities were then used to calculate the Stokes drag on the grains. The overall simulation cell in which these calculations are run is on the order of ten times the disk Kolmogorov scale, giving a good idea of the size of the turbulent gas flow simulation required.

D. JHU Turbulence Database

Johns Hopkins University maintains a web-accessible database (JHTDB) of a turbulent fluid flow. Velocities, pressures, and gradients are calculated at 1024 regularly spaced points on each Cartesian axis over a box with spatial dimensions (2π, 2π, 2π), with 1024 time-steps, modeling the evolution of gas turbulence features. The database, queried via the web from a MATLAB script employing fast-SOAP protocol for quick data downloads provided by JHU, may be accessed as a whole, or in data cutouts up to 1024^4 grid points in space and time [11]. For this project, only the velocity at each grid point is downloaded and saved into 3D matrices using MATLAB, where each (x, y, z) matrix element holds the velocity at that spatial position. Handling the velocity parametrically makes the vector nature of velocity fit ideally with MATLAB matrices. This algorithm is also easily implemented in C. For each (x, y, z) grid-point, only one component of the velocity is stored as an element in the matrix, requiring three parallel matrices for each velocity data cube at each time step, as Fig. 2 illustrates.

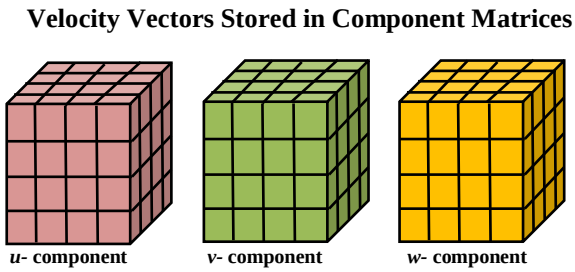

Fig. 2. Velocity vectors split parametrically into three parallel data cubes, where the same (x, y, z) grid-point location holds a vector component. By working with all three simultaneously, 3D velocities are achieved.

E. Finding Velocity with Interpolation

To find the Stokes' drag force acting on the grain at each time step, the gas velocity must be known at that point in space and time. Since each 3D velocity grid gives the value of the gas velocity in space, the $N\times N\times N$ grid-points define $(N-1)\times(N-1)\times(N-1)$ smaller index cubes. The eight vertices of the grid surrounding the location P of a grain give its position one of the index cubes. The gas velocities at these eight grid vertices are used to estimate the velocity of the gas at the interior point P where the grain resides, using trilinear interpolation. This interpolation method of weighting vertex values of a three-dimensional box for any point P inside gives a spatially-accurate value by multiplying the value of each i-th vertex by the ratio of the i-th sub-volume to the total index cube's volume. A diagonal line between a given vertex and P defines the sub-volumes, and weights of each vertex may be found as ratios of box sub-volumes to the total volume. Each vertex's corresponding sub-volume is diagonally opposite it. The diagonal from P to the anti-vertex i defines the sub-volume i, as illustrated in Fig. 3., since a vertex closer to P should receive a larger weight (volume) than one far from P, and that volume corresponds to the anti-vertex rather than the vertex. Thus, for sub-volumes V_1 through V_8, velocity cube vertices v_1 through v_8, and an index cube of total volume V_{tot}

$$\boldsymbol{v}_P = \frac{V_1}{V_{tot}}\boldsymbol{v}_1 + \frac{V_2}{V_{tot}}\boldsymbol{v}_2 + \frac{V_3}{V_{tot}}\boldsymbol{v}_3 + \cdots + \frac{V_8}{V_{tot}}\boldsymbol{v}_8 \qquad (8)$$

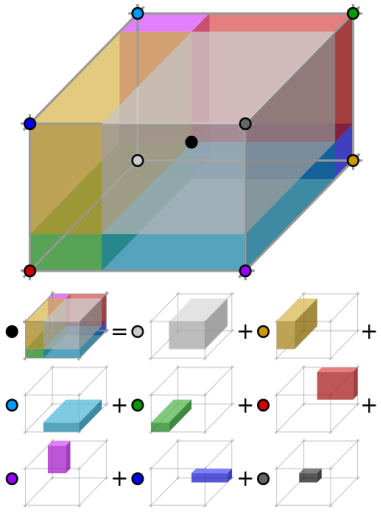

Fig. 3. Trilinear interpolation, in which the weighted average of the 8 vertices at a given P, where the correlated vertices and sub-volumes are color-coded. Courtesy Wikimedia illustrator Cmglee used under CC BY-SA 3.0

In addition to interpolating in space, the model must find the gas velocity at times between the time steps at which the cubes are given by JHU. This is done parametrically for the x, y, z components of the velocity with a simple linear interpolation in time. Using the velocity cubes at the time steps above and below the desired point in time,

$$\boldsymbol{v}_{interp}(t) = \begin{bmatrix} \frac{u_{high}-u_{low}}{\Delta t} \\ \frac{v_{high}-v_{low}}{\Delta t} \\ \frac{w_{high}-w_{low}}{\Delta t} \end{bmatrix}(t - t_{low}) + \begin{bmatrix} u_{low} \\ v_{low} \\ w_{low} \end{bmatrix} \qquad (9)$$

where u, v, w are the parametric components of velocity. The "low" velocity components correspond to the three parallel velocity matrices at the timestep below the interpolated time, while the "high" components are for the timestep above.

F. Scaling the JHTDB to Disk Turbulence

Since the JHTDB data cubes have their own integral and Kolmogorov scales, these must be scaled to match conditions in the disk. Disk values are obtained from JHTDB values by

$$\begin{aligned} \boldsymbol{r}_{disk} &= \frac{L_{disk}}{L_{JHU}} * \boldsymbol{r}_{JHU} \\ t_{disk} &= \frac{t_{L,disk}}{t_{L,JHU}} * t_{JHU} \\ \boldsymbol{v}_{disk} &= \frac{v_{L,disk}}{v_{L,JHU}} * \boldsymbol{v}_{JHU} \end{aligned} \qquad (10)$$

where the length, time and velocity scales for the JHTDB data are $L_{JHU} = 1.376$, $t_{L,\,JHU} = 2.02$, and $v_{L,\,JHU} = 0.6812$ [11][12].

For the trilinear calculations, the distance between grid-points is normalized such that each grid-point is referenced by integer values. For example, an element of velocity matrix A on the second row, third column, and first page of a 3D matrix is $A(2, 3, 1)$, while the normalized location of that grid point is (2, 3, 1). Such a normalized system allows the same values to represent both a matrix element identifier as well as a physical position. Furthermore, for a point P inside an index cube defined by this matrix element and one element above, to the right, and backwards, distances are always less than one, convenient for trilinear interpolation since $V_{tot} = 1$, and V_n the volume given in terms of the component distances from the bottom, front, left vertex, given by the direct distance of the n-th component from the vertex to P, or one minus that distance. For instance, the normalized position P = (1.3, 2.4, 1.2) will always have volumes in terms of some combination of 0.3, 0.4, 0.2, or 0.7, 0.6, 0.2. Since there are 1024 grid-points in each direction, there will be 1023 spaces between points, so converting between these normalized integer units and JHTDB units entails multiplying by $2\pi/1023$ or its inverse [13]. Physical disk units (SI) are then determined with the scale factors in Eq. (10).

III. Results

Several types of preliminary results were obtained with the turbulent dust model. First, the gas velocities were downloaded and interpolated for a 20×20×20 cube of velocities and their correlated positions in space. The

interpolations do show turbulent behavior, as shown in Fig. 4. Smaller selections of interpolated velocities were graphed alongside actual grid velocities to find that the velocity vectors had similar directions and magnitudes, as shown in Fig 5.

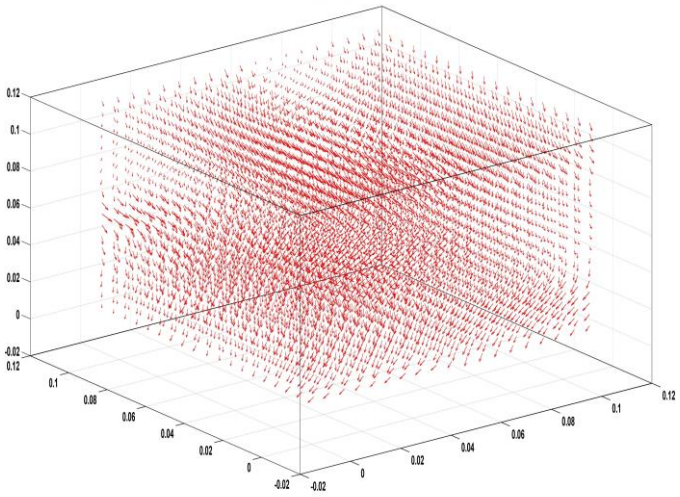

Fig. 4. Turbulent gas flow interpolated once near the mid-point of each index cube, in a 19-by-19-by-19 data cube with 20^3 grid points. All interpolated velocities are shown as red vectors. Notice the swirling eddy pattern of the interpolated velocities.

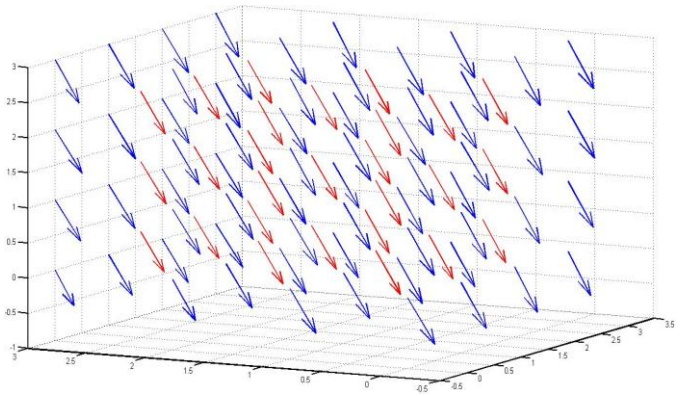

Fig. 5. A subset of interpolated turbulent velocities, near the mid-point of each index cube, in a 3-by-3-by-3 data cube with 4^3 grid points. All interpolated velocities are shown as red vectors, while JHTDB velocities are shown as blue vectors.

In these simulations, the distance from the central star is set to be 1-2 AU, with a star of one solar mass. Assuming a disk gas of primarily H_2, γ = 7/5 [7], μ = 2.34, $\sigma_{mol} = 2.0\times10^{-19}$ m^2 [1, qtd. Chapman and Cowling, 1970], and T = 300 K, it follows that c_g = 1200 m/s, $\nu = 4.7$ m^2/s, and $Re = 4.3\times10^9$. With these values and $\alpha = 0.01$ [1] and $\Omega_K = 7.1 \times 10^{-7}$ s^{-1} [7], the disk integral scale is described by Eq. (1) as $L = 1.7 \times 10^8$ m, $t_L = 1.4 \times 10^6$ s, and v_L = 120 m/s. It follows from Eq. (3) that the disk Kolmogorov scale is given by η = 10 m, t_η = 21 s, and v_η = 0.47 m/s. A stopping time of 0.63 s is obtained by assuming a spherical grain of constant density ρ_d = 3000 kg/m^3 and radius $r_d = 1.0 \times 10^{-6}$ m, with $m_d = 1.3 \times 10^{-14}$ kg, $\sigma_d = 3.1 \times 10^{-12}$ m^2, and $\rho_g = 4.0 \times 10^{-6}$ kg/m^3 [14].

The second category of results obtained involved the stopping time of the dust grains when subjected to the gas drag of turbulent eddies. Simulations with the numerical model demonstrate increasing stopping times and thus weaker coupling for larger grains.

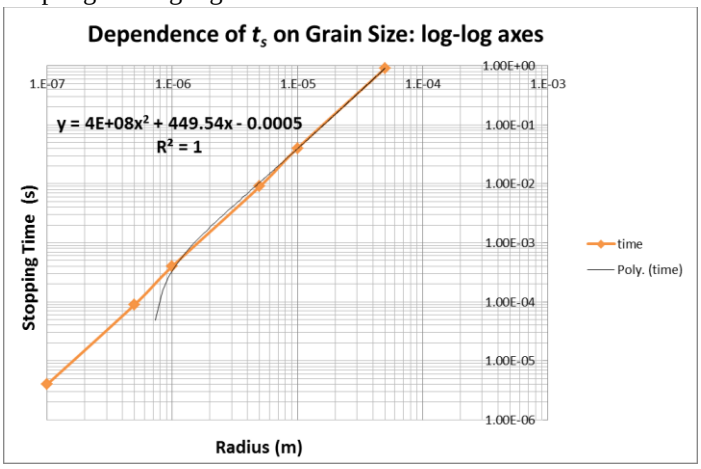

Fig. 6. Grain stopping time as a function of grain radius. Notice the lack of a linear scale, as well as the values of the stopping time, far less than 1 second.

Unfortunately, while Eq. (5) dictates stopping times scale linearly with increasing particle radius, the stopping times found for grain radii 1×10^{-7} m, 1×10^{-6} m, 1×10^{-5} m are, respectively, 5.0×10^{-6} s, 5.0×10^{-4} s, and 5.0×10^{-2} s, indicating error in the code, as numerical stopping time scales as approximately r_d^2 instead of r_d over the radius interval. (In fact, the linear trend of the log-log data plot means the relationship must be given by a power law fit.)

A third result involves the CPU time needed to run the processes related to the turbulent velocity interpolation. These processes include scaling of quantities, opening binary files containing JHU velocity data, reading those data into 3D matrices, performing the interpolation calculations, and closing the binary files. In order to determine how much relative CPU time is required for each of these sub-processes of reading in and interpolating the JHTDB velocities, a test version of the interpolation code was run in MATLAB, outputting the elapsed CPU time of each sub-process, and after five runs of the code to average the elapsed time of each section, these values were divided by an approximate total time. This normalization gives the information necessary to modify the algorithm for maximum CPU efficiency.

TABLE I
DEPENDENCE OF STOPPING TIME ON GRAIN SIZE

Grain Size	Stopping Time
1×10^{-7} m	4×10^{-6} s
5×10^{-7} m	9×10^{-5} s
1×10^{-6} m	4×10^{-4} s
5×10^{-6} m	9×10^{-3} s
1×10^{-5} m	4×10^{-2} s
5×10^{-5} m	9×10^{-1} s

Data obtained in MATLAB version of box_tree dynamics code, observing the time it takes grain velocity to approach gas velocity.

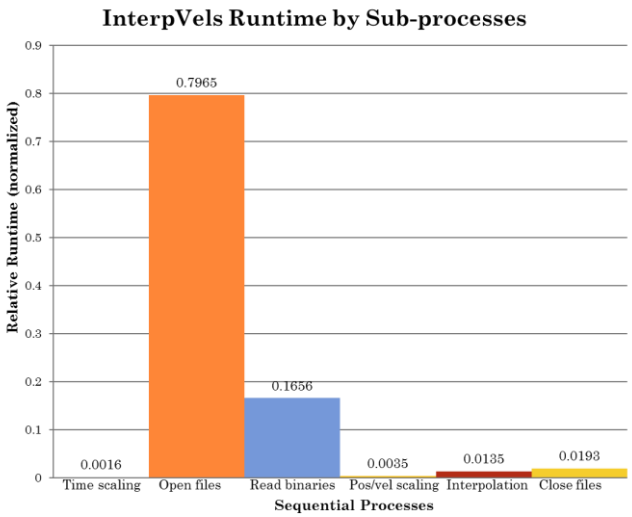

Fig. 7. Relative CPU time required for each sub-process in the box_tree JHTDB binary read-in and interpolation of velocity data. Relative CPU usage data are given both as normalized bars and decimal ratios.

IV. Discussion

The preliminary results obtained show promise for implementing turbulent gas drag functionality for box_tree. The power law relationship of grain size to stopping time, as shown in Fig. 6., rather than the theoretical linear correlation, as well as the unphysical stopping times for a test grain under turbulent gas drag, indicate that possible causes of the 10^3 order of magnitude difference in t_s include bad values of absolute viscosity, a problem with the fourth-order Runga-Kutta (RK4) method used to integrate the grain trajectory, or a bad assumption somewhere in the code. Viscosity does not seem to be the culprit, as the calculated value for absolute viscosity agrees with that of Moudens *et al* [15]. Recent investigations of the first few time-steps in the integration have turned up anomalous behavior such as small fluctuations in what should be a smooth change in velocity, as well as a large jump in the velocity at the second or third time-step. This implies that at least part of the problem is with the RK4 method. Further work is needed to resolve this issue.

V. Ongoing and Future Work

A. Debugging and Maximizing CPU Efficiency

Anomalous stopping time behavior requires further debugging. The RK4 method needs to be monitored at every step of the process to find where the non-smooth velocity integration is occurring. If this does not solve the issue, more detailed investigation of all steps of the interpolation and gas drag calculations will likely prove necessary.

At present, the C version of the box_tree code running in a Linux environment is not successfully reading the potentials file containing the gas drag functions, though it compiles without error. This is also proving difficult to debug. Both issues need to be resolved in order to have confidence in our results, the second is more important, as the *N*-grain calculations are run using C in a Linux environment.

Also, the information gathered on the relative run-times of each part of the interpolation process revealed a need for minimizing the number of times files are opened, close, and read. For now, this is done every time-step, but with many time-steps occurring for the same high and low time-step binary files, this is unnecessary. Following a line of thought developed by Fager, an algorithm employing occasional loading into static arrays is under development [13]. The basic process involves loading the low and high time-step files on the first function call, saving data into "static double" arrays. Then close the first file. When the global time variable finishes cycling through all particles and begins the next cycle, a check is performed to see if time is within the range of the two static arrays. If so, then they are not updated. If not, the next high time-step binary file loads, writing over the array values of the second file, which in turn overwrite the array values of the first file.

B. Application to N Uncharged Grains

Once the gas drag code is behaving as expected, *N*-body grain simulations need to be carried out over time. Because space is a near vacuum, the grain number density in the simulation cell must be artificially scaled up to cause grain collisions on a time scale the CPU can run. A correction factor will be applied after the simulation completes to account for the density change. The effect of turbulence on coagulation of non-charged grains will be determined to give a control for the dusty plasma case.

C. Application to N Charged Grains

After the non-charged grains are run, coagulation tests with charged grains in turbulence will be run to see if turbulent eddies provide a means to overcome the grain-charging barrier to aggregate formation described in Okuzumi, *et al*. [1].

VI. Conclusion

In conclusion, the import of a simulated turbulent flow into box_tree, for use in interpolating gas velocities throughout the simulation cell at all times, and finding gas drag forces on dust grains within it, shows promise for answering the research question. Whether turbulence brings about greater sticking or breakup of the aggregates, or slows down the overall rate of coagulation, is a question well on the way to having an answer. Preliminary tests, while problematic, show a dependence of stopping time on grain size, as well as successful interpolation of turbulent gas velocities. Isn't science fun?

Acknowledgments

To begin, the insight and guidance of Dr. Lorin Matthews on this project are extremely appreciated, though several others deserve thanks as well. These include Dr. Charles Meneveau of Johns Hopkins University, who helped with the disk-to-JHTDB scaling via email correspondence, Peter Fager, who through email correspondence and his paper reviewing past work on this project, helped me think through several physics and coding issues. My fellow REU students' examples, drives, and critiques also helped my thinking about the project and made summer research at CASPER a joy. Special thanks

especially are due to Jeremy Smallwood and Amy Anderson. I also wish to thank Dr. Truell Hyde, CASPER director, for his mentorship, oversight, and provision of research funding.

In addition to the individuals above, I wish to thank the National Science Foundation for research grant No. 1262031, without which this project would not be possible.

REFERENCES

[1] S. Okuzumi *et al*., "Electrostatic barrier against dust growth in protoplanetary disks. II. Measuring the size of the 'frozen' zone," *The Astrophysical Journal*, vol. 731, no. 96, 14pp, Apr. 2011.

[2] H. Mizuno et al., "Formation of the Giant Planets," Progress of Theoretical Physics, vol. 64, no. 2, pp. 544-557, Aug. 1980.

[3] S. J. Weidenschilling and J. N. Cuzzi, "Formation of planetesimals in the solar nebula," *Protostars and Planets III*, (A93-42937 17-90), pp. 1031-1060. 1993.

[4] L. S. Matthews *et al*., "Modeling agglomeration of dust particles in plasma," Center for Astrophysics, Space Science, and Engineering Research, Baylor University, Waco TX 76798-7310, Oct. 2011.

[5] S. A. Balbus, J. F. Hawley, "A powerful local shear instability in weakly magnetized disks. I - Linear analysis. II - Nonlinear evolution," *The Astrophysical Journal*, vol. 376, pp. 214-233, Jul. 1991.

[6] P. A. McMurtry, "Turbulence," class notes for ME 7960, Dept. of Mechanical Engineering, University of Utah. Available: http://www.eng.utah.edu/~mcmurtry/Turbulence/turbulence.html

[7] S. J. Weidenschilling, "Evolution of grains in a turbulent solar nebula," *Icarus*, vol. 60, pp. 553-567, 1984.

[8] C. W. Ormel and J. N. Cuzzi, "Closed-form expressions for particle relative velocities induced by turbulence (Research note)," *Astronomy and Astrophysics*, vol. 466, pp. 41-420, 2007.

[9] J. R. Taylor, "Projectiles and Charged Particles," in *Classical Mechanics*, USA: Edwards Brothers, Inc., 2005, p 72.

[10] D. C. Richardson, "Planetesimal dynamics," Ph.D. dissertation, Clare Hall Inst. Of Astro., University of Cambridge, Cambridge, UK, 1993.

[11] Y. Li *et al*., "A public turbulence database cluster and applications to study lagrangian evolution of velocity increments in turbulence," *Journal of Turbulence*, vol. 9, N31, 2008.

[12] C. Meneveau. (2014, Jun. 26). *Scalability of JHU Isotropic Forced Turbulence Database* [Online]. Available e-mail: RE_Ward_Howard@baylor.edu Message: Re: Scalability of JHU Isotropic Forced Turbulence Database.

[13] P. J. Fager *et al*., "The effects of turbulence on dust particle coagulation in planetary formation," Center for Astrophysics, Space Science, and Engineering Research, Baylor University, Waco TX 76798-7310, Aug. 2013. Available: http://www.baylor.edu/content/services/document.php/208051.pdf

[14] S. P. Ruden and J. B. Pollack, "The dynamical evolution of the protosolar nebula," *The Astrophysical Journal*, vol. 375, pp. 740-760, Jul. 1991.

[15] A. Moudens *et al*., Photophoretic transport of hot minerals in the solar nebula," *Astronomy and Astrophysics*, vol. 531, A106, 11 pp, May 2011.