Efficiency of Planar Disk Packings

Rohit Banerjee Jacob Krol Supervisor: Ali Mohajer
University of Illinois at Chicago

Mathematical Computing Laboratory

Summary

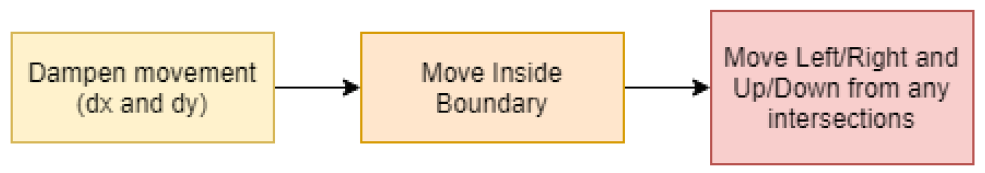
The maximum packing density for disks of the same radius in the plane is well-understood, and is equal to $\pi/\sqrt{12}(\simeq 0.9069)$. Less is known about the maximum density of packings of unequal disks in the plane. This projects objective is to investigate the shape of the curve describing the optimal density of a randomly-generated two-species packing of disks the plane. We do this by generating random packings of disks of two radii. The parameters we vary are the ratio of the two radii, and the relative number of disks of each size.

Background

The highest density packing of two-dimensional disks on an infinite plane is well known. However the upper bound on the density of a two species packing, given proportion and radii is unknown. The findings in this project are a continuation of a previous project done in the Spring of 2018 which resulted in the gaseous physics engine and voronoï implementation used here. It was found in the previous project that the gaseous physics (as opposed to random placement or gravity) produced the best results for our purposes.

Simulation and Computation

Our *Physics Engine* was written in C++ and computes dynamic interactions between disks in a 2D plane while JavaScript was used for visualizing processes and results. The physics operated under the following rules.



Computations. The largest parts of our project were our computational programs functioned to create and fill a 2D plane, or "palette," of disks, arrange them according to physics algorithms, automate the testing process, test saturation of each palette, and report the final results into compact files. All written in C++, our *Physics Engine* and, using the CGAL libraries, *Saturate Process* were nested within our much larger *Data Collector*.

Visualizations. The ability to visualize our results, as well as our processes, became fundamental in the progress of the project for checking the effectiveness of our computational engines, as well as displaying the results. Our visualization programs were based in JavaScript, with the most powerful being the *Voronoi Playground*.

Computational Complexity

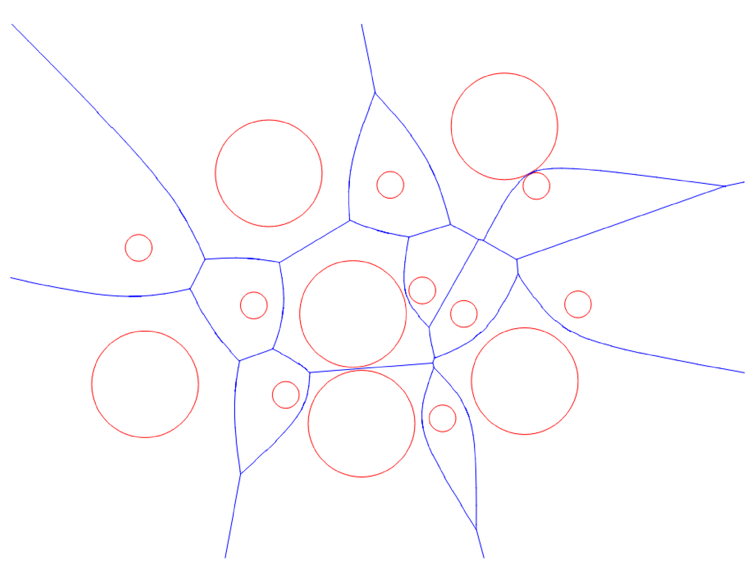
Within the *Physics Engine*, the effect of hard boundaries became a problem in computing reliable and consistent density data. Additionally, to check for collisions, every disk's position is checked against every other disk which runs in $\mathcal{O}(n^2)$ complexity. To shorten run-time, a condition was added, wherein the nested loop would skip any calculations for disks further apart than the sum of their radii. Though this improved run-time significantly, complexity remained unchanged.

Further attempts included partitioning the palette into a grid of cells, thus only necessitating that a check be ran on disks within the same or adjacent cells. The implementation of spatial hashing is still underway.

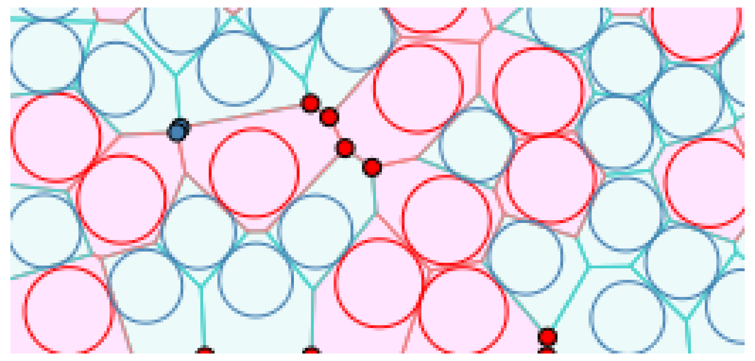
Additively Weighted Voronoï Decompositions

Once we have filled a palette with disks, one of the most crucial steps is determining whether it is sufficiently "saturated." We define the palette to be saturated if for any point in our palette, there is not space to add another disk at that point without it intersecting at least one of the already existing neighbors or the boundaries of the palette. It would be exhaustive to naively check every point and run an intersection calculation. Instead, we incorporated Voronoï decompositions into our analysis.

Additvely Weighted Voronoï decompositions are a set of lines drawn on the plane such that any point on the line lies as far as possible from all adjacent disks, or sites. An arrangement may look something like the following:

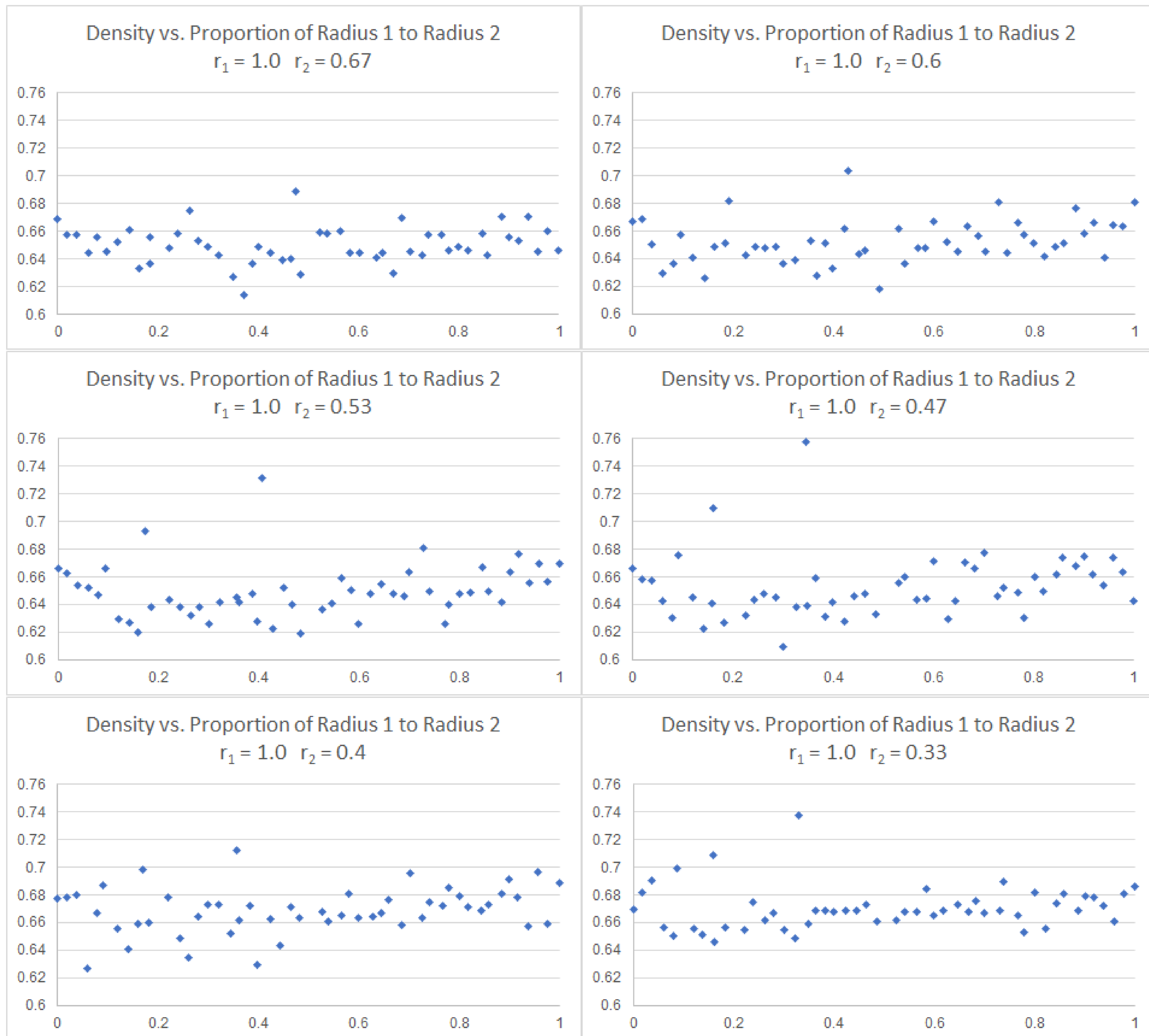


From this, we can extract the vertices of these line segments, and this would grant us a finite set of points to check for any open space. The use of the CGAL Voronoï library to extract vertices was contributed by Dr. David Dumas over email (October 2018). While this runs within our *Data Collector* program to determine saturation, our JavaScript adaptation also allows for visualizations of the process. Essentially, we filter these vertices to determine which have available space, and which of the two species could fit within that space. This can be seen below.



Data

The x-axes record the proportion of the palette filled with disks of radius 1 and the y-axes record the density of the final saturated packing and have a much smaller range than 0 to 1 to more clearly present our results. After determining the closest values to approximating an infinite plane that our programs could handle, we applied multiple runs of our *Data Collector* at various ratios near this limit. Our collected densities for these runs were consistent across trials, and a single set of representatives for each of six ratios are provided below.



While the radii values are scaled above such that the larger radius is 1.0 for all runs, it is valuable to note for future projects that the actual scale, *larger radius* : *side length of finite boundary*, was approximately 1:52.

Identifying Trends

Greater difference in radii size led to higher average densities. As the relative size of the smaller radius shrank, the average densities of the packings increased. Though this shift was not drastic, it was consistent, signifying the validity. This is expected, however, since a smaller second disk should allow for more efficient packings.

Certain proportions are more optimal than others. Less obvious, however, was the appearance of the same outliers in every graph. There is a noticeable upward trend in density across the same 3 data points that rises above the trend line. These proportions are approximately 0.09, 0.17, and 0.35. This may be an indication that, while most proportions of radii have little effect on the density, particular ratios produce more ideal packings. A more improved physics engine will either confirm or deny this.

Future Direction

Despite many challenges throughout the progression of the project, there are still many areas which can be sharpened in future research to improve upon the results collected herein.

Larger palettes. Because we are limited to finite palettes, our larger radius would only be capable of reaching a density around 0.8356 even if placed in an ideal hexagonal packing. This means that more precise data can be collected if the palettes used are grown relative to the disks to better approximate this infinite plane.

Optimization of algorithms. This project was limited to the scale of disks used due to complexity. In order to compute significant results in a reasonable amount of time, a primary objective should be to reduce the complexity physics algorithm. The current solution is spatial hashing which will split the palette into cells that can be represented as vectors in the C++ Data Collector which allow for dynamic allocation of disks within the cells.

Results and Conclusions

The project approximated average and maximal densities given a proportion and ratio of radii in a two-species randomly-generated packing. The data would suggest a density upper bound of density around 0.75 for our testing data, although the expected outcome was a density exceeding that of a single species. It is likely that due to the current saturation algorithm an effect like heat expansion places the disks farther apart and prevents higher densities from occurring. Obtaining a more precise average is possible using more data sets with a larger disk-to-palette ratio in order to avoid the problems that arise with strict boundary conditions.

References

Hale, Thomas C. *Cannonballs and Honeycombs* Notices of the AMS **Volume 47, Number 4**.(April 2000), 440-449.

Heppes, Aladar. *Some Densest Two-Size Disc Packings in the Plane*. Discrete and Computational Geometry **30:241262**. (2003), 241-262.

Karavelas, Menelaos and Yvinec, Mariette. 2D Apollonius Graphs (Delaunay Graphs of Disks). In *CGAL User and Reference Manual*. CGAL Editorial Board **4.13** edition. 2018.

Lagarias, Jeffrey C. *The Kepler Conjecture and Its Proof*. Godfrey Tampach. (1611), 3-26.

Kcnarf, d3-weighted-voronoi, (2018), GitHub repository, `https://github.com/Kcnarf/d3-weighted-voronoi`

Banerjee, Krol *Efficency of Planar Disk Packings* Mathematical Computing Laboratory, University of Illinois at Chicago. Spring 2018.