Summary

We use a genetic algorithm (GA) to explore the shape of an optimal brownie pan. The rectangle is a common shape for a baking pan, however, the shape is inferior in terms of temperature distribution because the heat tends to build up around the corners. Circular pans, which do not have this drawback, cannot be as efficiently packed in an oven, and so may not be optimal, either. Thus, our goal is to find a shape that has the best of both worlds: heat distribution and spatial efficiency. To determine the heat distribution on the pan and brownie, we solve the diffusion equation numerically using the finite difference method. An algorithm to compute the effective area of a shape determines the spatial efficiency. We evaluate shapes for fitness according to these criteria, and use the most successful to form the next generation of shapes.

GA is well-suited for the problem, since our solution space consists of an enormous number of possible shapes. Two chaotic area-preserving maps, Arnold's Cat map and Chirikov Standard map, play a crucial role in our GA. We use the chaotic transformations to produce various shapes in an almost random manner, a process analogous to generating pseudo-random numbers. Furthermore, since the particular chaotic transformations are area-preserving, shapes generated by this process, no matter how different they may appear, have the same area. Thus, our GA and the chaotic transformations constitute an effective method for sampling many different shapes, including ones that cannot be easily generated by straightforward methods. The process has the potential to find entirely new shapes that outperform conventional shapes in both heat distribution and spatial efficiency.

Our finite difference PDE solver, implemented in SciPy, successfully replicates heat build-up around the corners of a rectangular pan. The data obtained by varying thermal conductivity of the pan between 0.05 and 215 indicate that as long as the conductivity of pan material is above a moderately low threshold, an increase in conductivity does little to alter the heat dynamics inside the pan. We find no statistically significant difference, for example, between conductivity 1.05 and 215, which approximate water and hot aluminum, however we find major differences between pans with conductivity 0.05 and 1.05. This suggests that the heat distribution in the pan is much more sensitive to changes in the lower range of conductivity: in general, one can expect only a small improvement on the distribution of heat by using a material with higher thermal conductivity. We also find that the variances in batter temperature, which we take as a measure of heat distribution, differ insignificantly between aluminum and glass pans but greatly between glass and our theoretical "low conductivity" pan.

Perfect Brownies from Chaos: Finite Difference, Chaotic Maps, and Genetic Algorithms

MCM Contest Question A

Team # 22555

February 4, 2013

Contents

List of Figures

1 Introduction

This paper is concerned with modeling and optimization of brownie baking in a standard rectangular oven. Pans that are commonly used for baking are rectangular, however, the

rectangle is not an ideal shape when considering the distribution of heat. Heat tends to build up around the four corner, and the brownie would not be cooked evenly. To resolve this issue, one may use a circular pan instead. This shape, however, has limitations, as well. Since ovens are usually rectangular, circular pans cannot fill the space efficiently, and the amount of goods that can be baked at the same time is reduced by approximately 75%. Our goal is to find a pan shape that excel in both heat distribution and spatial efficiency. Specifically, our model attempts to:

- Determine the optimal base shape with regard to the distribution of heat.
- Determine the optimal thermal conductivity of the pan with regard to the distribution of heat.
- Determine the optimal base shape with regard to the number of pans that can fit in the oven.

We attempt to implement a model that employs chaotic transformations to generate batches of equal-area base shapes in a two-dimensional plane, which are linearly extended into three dimensional 'pans'. Pans are evaluated using numerical solutions of the heat equation by the finite difference method, then culled and recombined by a genetic algorithm in the hope of producing an optimal population. An analytic solution to the rectangular problem is also explored.

The paper is structures as follows. First, we discuss the overall picture of our method in the next section. Next, we present assumptions and their justifications, variables, and constants that are incorporated in our model. Then, we provide backgrounds of the methods used to implement the model. Finally, we analyze the model and the data obtained from it.

2 Our Approach: Genetic Algorithm

Our model employs the following three methods:

1. Genetic Algorithm (GA)
2. Numerically Solving the Dissipation Equation using the Finite Difference Method
3. Generation of Random Shapes by Area-Preserving Chaotic Maps

In this section, we discuss GA, the core of our approach, in some detail, and the other two only briefly. The other two are discussed more extensively in later sections.

GA is an optimization heuristic that mimics the process of natural evolution (Fig 1). GA assigns each pan a *fitness* value based on how close its design is to achieving a set of aims enumerated above (Fig 2).

The more fit pans are more likely to survive and recombined to form a new pan, which is used in the next iteration of the algorithm. After many iterations, we would obtain

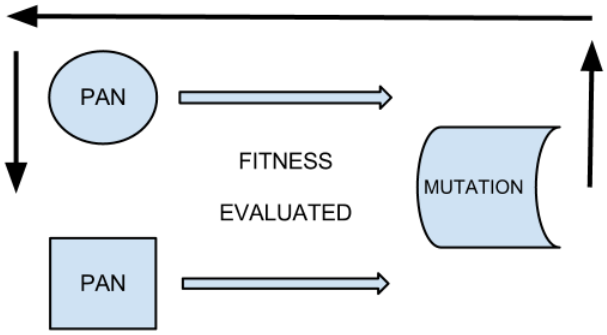

Figure 1: A flow chart of the genetic algorithm.

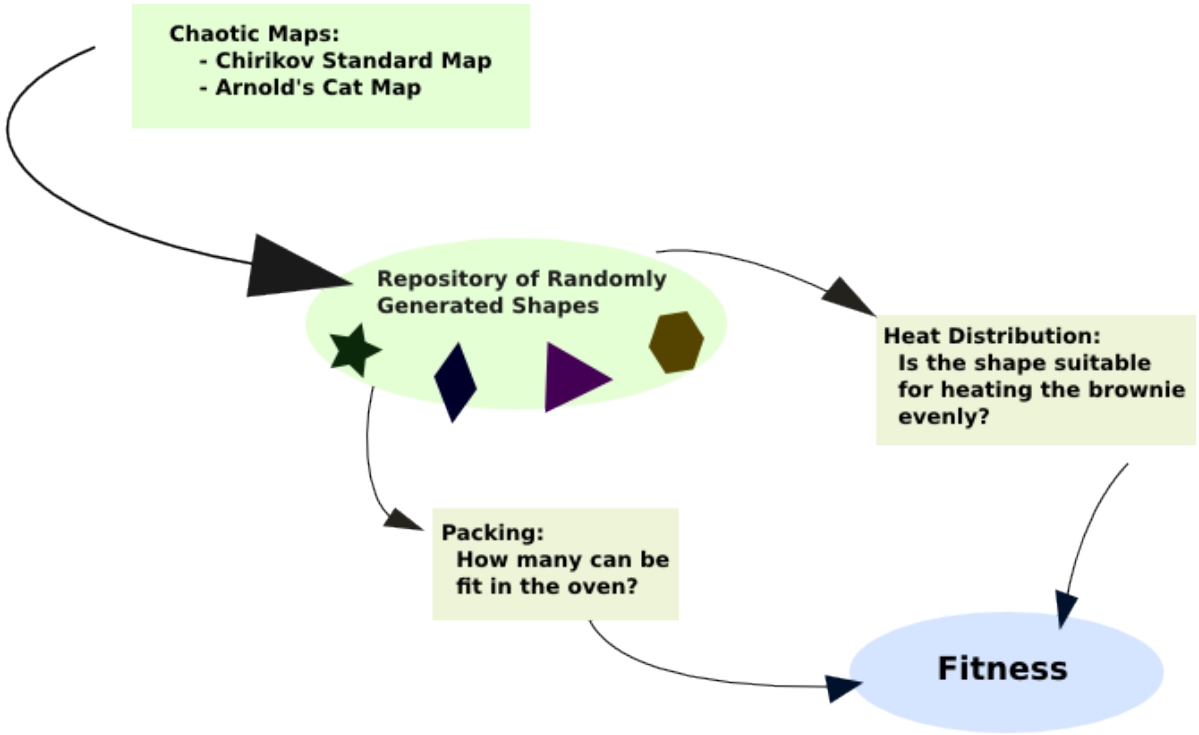

Figure 2: A flow chart of the generation of shape to the fitness evaluation of a shape.

shapes that satisfy our requirements. GA is especially suited for searching through a large solution space, which may be computationally infeasible for global optimization algorithms. This is exactly the case for our problem, since there are practically infinite number of shapes that we can adopt. Consider the following process that produces a polygon of area A:

1. Pick a random integer $N \geq 3$.
2. Pick N points on $\mathbb{R}^2$.
3. Take the polygon formed by the N points as our shape.
4. Compute the area of the polygon, and scale it so that the area would become A.

This process alone produces an enormous number of different shapes. Also, it is possible that an optimal pan shape is curvilinear, such as S- or spiral-shaped. Since GA has the potential to find an optimal solution just by searching through a fraction of the vast solution space, the method allows us to explore shapes that would not be considered in algorithms that require the search space to be reasonably-sized. Implementations of GA and their benefits as well as limitations are discussed in Mitchell [1998].

The other major components of our model are area-preserving chaotic maps, and numerical solutions of the diffusion equation. In short, area-preserving chaotic maps and numerical solutions to the diffusion equation provides our GA means of supplying random shapes and evaluating the fitness of shapes, respectively (Fig 2). Later sections discuss the methods in more detail.

3 The Model

3.1 Assumptions and Justifications

- The oven is preheated before the brownies and pan are placed inside, and the temperature of the oven is kept constant by a thermostat. The batter and pan start at equal temperature.
- We test a pan one at a time. Even in the case where more than one pan is placed in the oven, our model assumes the sides of our pans are held at constant temperature. This assumption becomes questionable in a closely packed oven, where the outer pans have more surface area exposed to heat.
- The walls of our pans will be flat.

3.2 Variables and Constants

- $n \times m$: grid size

- H: pan's wall height
- A: pan's area (Section 5)
- A_{eff}: the effective area of a shape (Section 5)
- T_{oven}: the oven's internal temperature
- T_{room}: the room temperature
- Fit_{total}: the overall fitness of the shape of a pan (Section 4)
- Fit_{pack}: the fitness of the shape of a pan measured by the number of the same shapes that can be packed in a oven (Section 4)
- Fit_{heat}: the fitness of the shape of a pan measured by reduction in edge overcooking and even heat distribution. (Section 4)
- K_{pan}: the conductivity constant of the pan
- K_{br}: the conductivity constant of the brownie

4 Genetic Algorithm

As we mentioned in the introduction, GA is a crucial piece of our solution. GA is used to explore an enormous solution space that is made possible by random shape generation by chaotic maps. Since we have already discussed the overall mechanism of our GA, in this section we define relevant mathematical terms. These are the three constants that we employ to describe the performance of a shape:

1. Fit_{total}: the overall fitness of the shape of a pan
2. Fit_{pack}: the fitness of the shape of a pan measured by the number of the same shapes that can be packed in a oven
3. Fit_{heat}: the fitness of the shape of a pan measured by reduction in edge overcooking and even heat distribution

Our area estimation algorithm (see Section 5) computes Fit_{pack}, and simulation of the heat equation determines Fit_{heat}. Fit_{total} is computed from Fit_{pack} and Fit_{heat} by

$$Fit_{total} = w \cdot Fit_{heat} + (1 - w) \cdot Fit_{pack},$$

where w is the weight on the requirement of heat distribution, as the problem statement requires. Thus, when w is large, our GA tends to let shapes with better abilities to distribute heat survive, regardless of their spacial efficiency.

5 Shapes

The area is fixed as A. To determine the optimal shape, we use $A = 1$. The effects of varying A are explored separately.

5.1 Mutation, Area-Preserving Map, and Chaos

In our GA, mutation serves to introduce new kinds of potential base shapes, which are then marked for success or failure by the culinary process of evolution. We consider mutation as a transformation of a two-dimensional region. Since pans must be comparable in aptness of packing, we need to mutate shapes in a way that preserves area.
An area-preserving map is a measure-preserving map in a two-dimensional sense, i.e. $m(L^{-1}) = m(L)$, where m is the Lebesgue measure. For example, suppose L is a linear transformation. If $\det(L) = 1$ then it is an area-preserving map; the class of such maps correspond to the special linear group $SP(2, R)$. A typical member of $SP(2, R)$, however, does not bring about a radical change to a shape. For example, the matrix

$$\begin{pmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{pmatrix}$$

as a linear map corresponds to the rotation by θ. Although this characteristic is suitable as a way of introducing slight modifications to a pre-existing shape, applications of such maps would not create an entirely new shape. This observation motivates us to employ chaotic maps instead. We can think of chaotic maps as having a higher mutation rate than non-chaotic maps. These transformations are deterministic, yet highly unpredictable. Repeating the application of the chaotic maps a random number of times is, in effect, very similar to producing random shapes of a fixed area. Our process of generating random shapes is similar to a process that generates pseudo-random numbers. Hence, we can use chaotic maps as a way of creating a random shape, which would correspond to generating a random rational number in a usual GA. Area-preserving chaotic maps, which mutate shapes in such a drastic manner, allow our GA to explore the solution space that would not be easily accessible otherwise. In particular, we employ two chaotic maps: the **cat map** and **kick map**.

5.2 The Cat Map

One of the chaotic maps that we use to mutate shapes is the cat map (commonly referred to as the "Arnold's Cat Map"). Arnold's cat map is a chaotic, area-preserving map on a two-dimensional torus [Hilborn, 1994]. The cat map F is defined as

$$F : (x, y) \mapsto (2x + y, x + y) \pmod 1.$$

The corresponding matrix is

$$A = \begin{pmatrix} 2 & 1 \\ 1 & 1 \end{pmatrix},$$

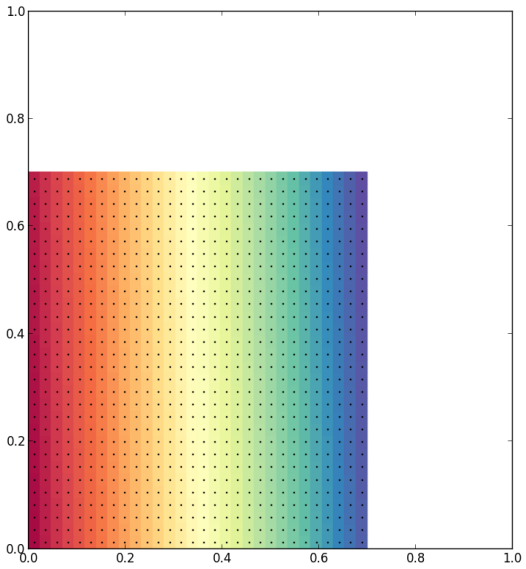

Figure 3: A square $A = 0.49$ (i.e. the length of each edge is 0.7). Each square consists of patches. This square, for instance, consists of 400 patches, each represented with different colors.

and clearly, $\det(A) = 1$. In order to obtain a wider variety of shapes, we parametrize the Arnold's map as follows:

$$F : (x, y) \mapsto (kx + (k-1)y, x + y) \pmod 1,$$

where $k \in \mathbb{R}$, $0 \leq k \leq 10$. Note that the determinant of the corresponding matrix

$$\begin{pmatrix} k & k-1 \\ 1 & 1 \end{pmatrix}$$

is one, and therefore, the parametrized Arnold's map is also area-preserving. The cat map stretches and folds the domain, and as a result, it produces numbers of thin strips. The parameter k determines the thickness of the strips; the higher the value of k, the thinner the strips are (Fig. 4). Although, k can be any real number, parameters k_0 and its negative $-k_0$ result in effectively the same transformation, since the mapped images would be symmetric about the origin, and we take modulus 1 of the points. Thus, we can only use $k \in \mathbb{R}^+$ without constraining our solution space. Also, the cat map tends to stretch the unit square to the x-direction for greater k. We will use $k \leq 10$, because, in a simulation using a limited precision, $k \geq 10$ does not give rise to appreciably different shapes (Fig. 5).

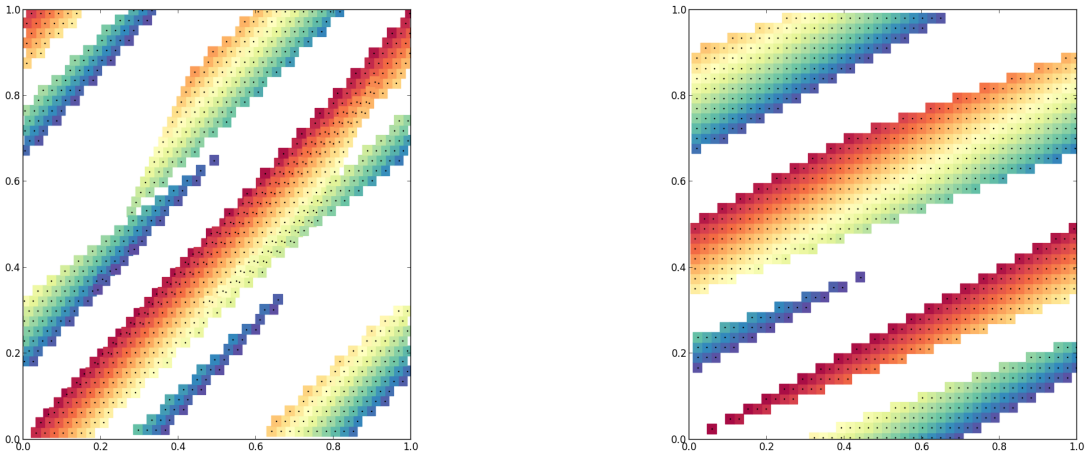

Figure 4: Cat Map. left: $k = 1.5$; right: $k = 3$. After one iteration. Colors of the patches correspond to those in (Fig. 3). The one with greater k tends to stretch the square to the x-direction, and splits the square region into thinner strips.

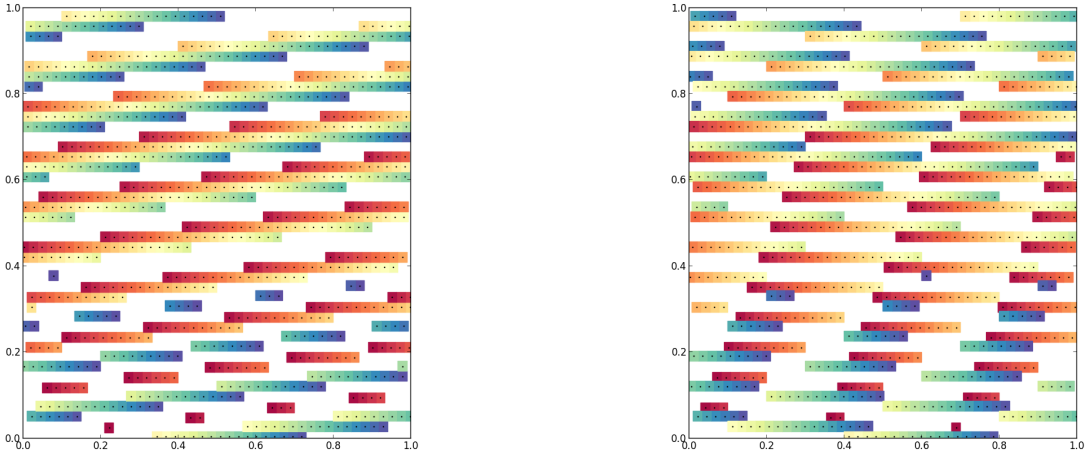

Figure 5: Cat Map. left: $k = 10$; right: $k = 30$. After one iteration. Although the two images are not exactly the same, the differences seem not to be significant from a qualitative point of view.

5.3 The Kick Map

The other area-preserving chaotic map we employ, the *kick map*, is usually called the Chirikov's Standard Map [Ott, 1993]. Qualitatively, the kick map *swirls* a shape, and

complements the cat map, whose primary action is *stretching*. The kick map is defined as:

$$y \mapsto y + k \sin x \pmod{2\pi}$$
$$x \mapsto x + y \pmod{2\pi},$$

where updates of y and x are done asynchronously (y first). The swirling effects of the kick map are stronger for higher value of k (Fig. 6). As k approaches 1, the kick map becomes a chaotic chaotic map. It is known that the chaotic border is $k = 0.971635...$ [Chirikov and Shepelyansky, 2008]. Although we would like to apply the kick map to the unit square, the original kick map is a function from $[0, 2\pi] \times [0, 2\pi]$ to the same region. We use the following version of the kick map whose domain and image are the unit square:

$$y \mapsto \frac{\pi + 2\pi y + k \sin(2\pi x)}{2\pi} \pmod 1$$
$$x \mapsto \frac{2\pi(x + y)}{2\pi} \pmod 1.$$

In plain words, the coordinates of each point in the unit square are scaled by 2π prior to the application of the original kick map, then the result is scaled by $1/2\pi$. Taking the chaotic border ($k \approx 1$ in the original kick map) into account, we take the range of k to be $(0, 2\pi)$.

5.4 Generation of Random Shapes

To generate random shapes, we repeatedly apply the chaotic maps to the unit square (Fig. 3) with parameters $0 \leq k \leq 10$ and $0 \leq k \leq 2\pi$ for the cat map and kick map, respectively. In many cases, random iterations result in a chaotic shape that is apparently not suitable as a shape for the pan. However, the process may arrive at orderly shapes such as those in Fig. 8. We see that the repeated random iteration of the cat map and kick map is able to generate random shapes. We expect that, by this method combined with the genetic algorithm, we may obtain an unforeseen shape that would outperform circular or rectangular pan in terms of both heat distribution and packing efficiency.

5.5 Computing the Boundary

As we saw in the previous sections, a shape generated by chaotic maps applied to a square is not necessarily a single piece (Fig. 9). We need an algorithm that computes the boundary of a given shape in order to run heat equation simulations for these shapes. Without knowing the boundary of a shape, we cannot determine the location of walls, which makes simulation impossible. To compute the boundary of any given shape, we use the *boundary fill algorithm*. With the knowledge of boundary, we can also check the integrity of a shape in the following manner:

1. Pick a point from the collection of patches that constitute a shape.

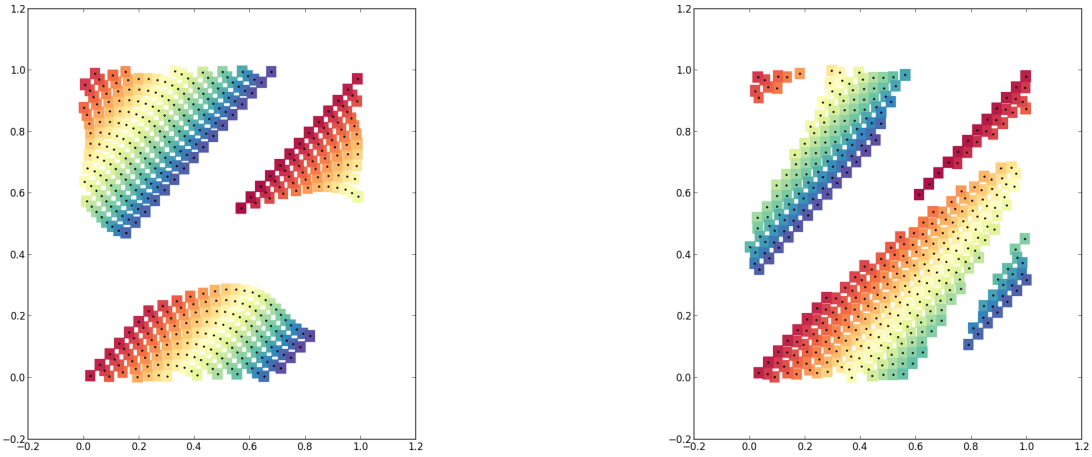

Figure 6: The kick map. left: $k = 0.5$; right: $k = 3$. After one iteration. The kick map with a higher value of k has stronger effects on the domain.

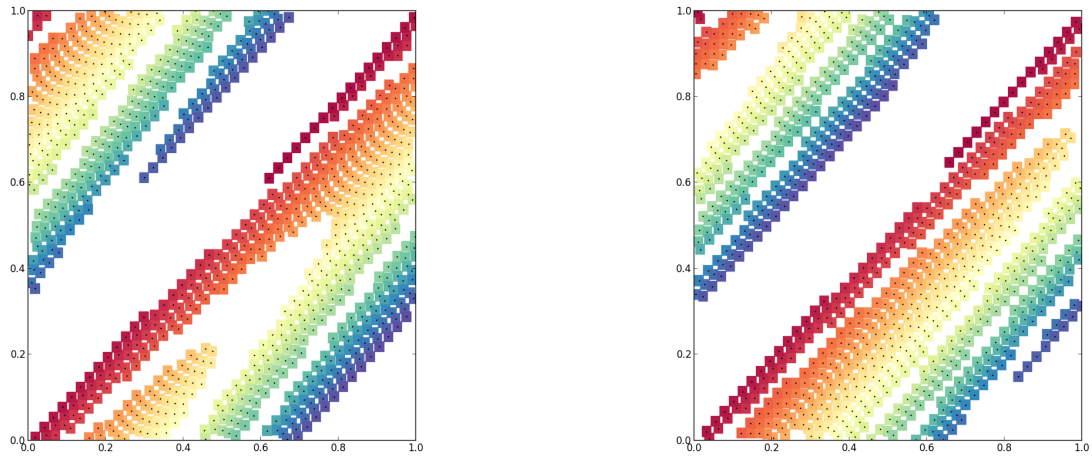

Figure 7: The kick map. left: $k = 10$; right: $k = 30$. After one iteration. Although the two images are not exactly the same, the differences seem not to be significant.

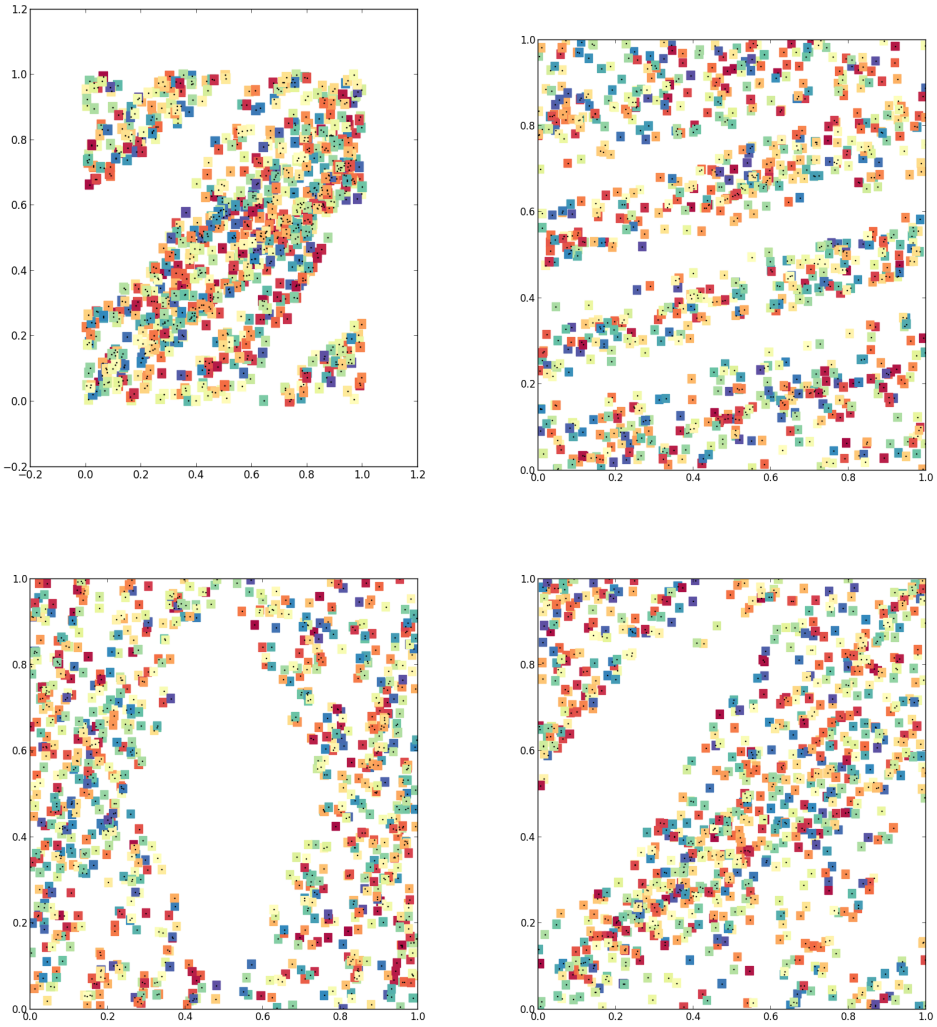

Figure 8: Even after random iterations, a process that seem to create a chaos, orderly figures may suddenly emerge. These images were produced after random iteration of the chaotic maps 2, 128, 256, and 1280 times, respectively.

2. Perform the boundary fill algorithm to compute the connected components of the point.

3. Compute the area inside the particular connected component by counting the number of patches in it.

4. If the area is less than $3A/4$ (the area of the original square), throw away the shape.

5. Also, if the area is larger than A, it means that there are holes on the boundary of the connected component. The shape would not function as a pan, so we abandon it.

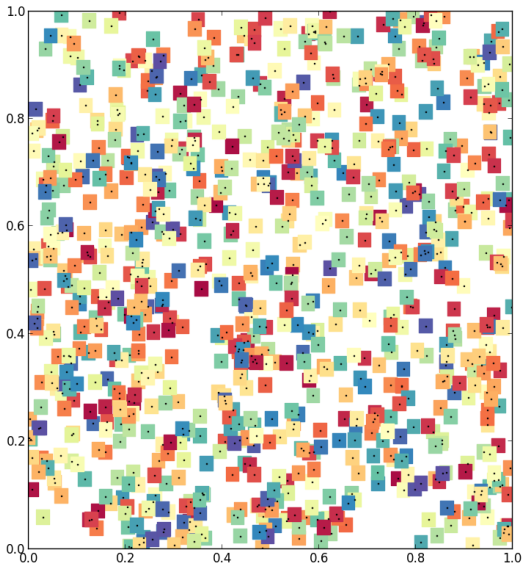

Figure 9: Another example of a square of area 0.64 after 128 random iterations of the cat map and kick map. Most of time, random iterations will create a disorderly figure.

We were not able to include a complete implementation of the boundary fill algorithm in our model. Because of this limitation, we could not perform the heat equation simulation to determine the heat distribution of a shape, which is a crucial procedure in our GA. Unfortunately, we were not able to use our genetic algorithm to find an optimal shape.

6 Packing

An ideal shape should fill the oven without creating a waste of space, maximizing the number of pans that can fit in the oven. Clearly, most of the shapes that our GA generates are not able to form tessellations. As a measure of how well a shape can pack inside the oven, we use the *effective area*. The effective area is the area where a shape occupies and cannot be shared with another shape. For example, if a shape had a hollow circle in the middle, the area would be less than that of the full circle; however, the effective area would be the area of the full circle. Fig. 10 shows how a shape with empty areas in between can have an effective area larger than A. To approximate the effective area of A_{eff}, we use the following algorithm:

1. Compute the center of mass, which equals to the mean coordinates of all points.

2. Consider four quadrants around the mean point.

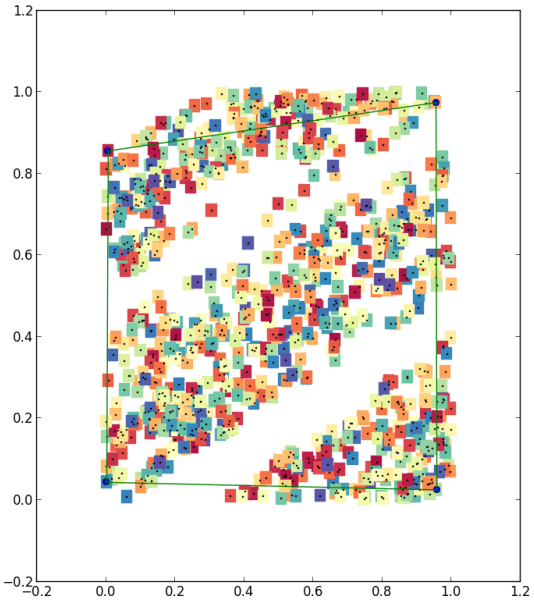

Figure 10: Demonstration of our algorithm to approximate the effective area. The area of the square that we started with is 0.81. The approximated effective area for this shape is 0.876625844048.

3. Within each quadrant, compute the point with the greatest distance from the mean point.
4. The effective area A_{eff} is approximated by a 4-gon with the four vertices.
5. The fitness Fit_{pack} is A/A_{eff}, a rational number less than or equal to 1.

The area of a quadrilateral defined by the four edges of length a, b, c, d is computed using the *Bretschneider's formula*:

$$\frac{1}{4}\sqrt{4p^2q^2 - (b^2 + d^2 - a^2 - c^2)^2}.$$

The fitness Fit_{pack} is expressed as A/A_{eff}. We have $Fit_{pack} \leq 1$, since A_{eff} is at least as large as A. Fit_{pack} will be used to compute the fitness of a shape together with Fit_{heat}.

7 The Heat Equation

Our approach began with an attempt to solve the heat equation for a 3D rectangular brownie pan using separation of variables and Fourier series. Our intent was to first solve

the problem for a rectangular mass of brownie batter with unknown boundary conditions on five sides and the upper face held at constant baking temperature T_b, and then solve the equation again for the pan around the batter, using the pan solution to fill in the boundary conditions for the batter solution.

In the end, we abandoned the analytic solution because it was not making progress toward our GA-based approach to the general problem. We failed to resolve an unknown in the batter solution, which kept us from attempting the second stage of the problem, the pan. The mathematical issues may have been worsened by our definition of the problem: our boundary conditions held the top of the batter at T_b, but the pan and interior brownie batter were given initial temperature T_i. This meant the edges of the batter touching the top of the pan were being held at two unequal temperatures simultaneously, a physical impossibility which at the least seems like a recipe for inaccurate temperature readings around the edges.

After we recognized the analytic method produced little result we switched to the method of finite differences, which proved more successful. Section 1 of what follows contains a partial analytic solution to the heat equation, and Section 2 is a discussion of our finite difference solution to the heat problem.

7.1 Analytic Solution

Consider a rectangular mass of batter flush with the first octant and having width W, height H, and depth l. Let the width of the baking pan be D. Let K be the conduction constant for brownie batter. Let T_i be the initial temperature of the batter and let T_b be the baking temperature. Let $f(x,y,z,t)$ be the heat function for the brownie pan, let $r = (x,y,z)$ and let $s = (x+D, y+D, z+D)$ be an 'adjustment factor', for later use, so that the pan problem may also be solved in the first octant. We wish to find a function T satisfying

$$\partial T\, t = K\nabla^2 T$$

subject to the initial condition

$$T(x,y,z,0) = T_i$$

for points in the batter and boundary conditions

$$\begin{aligned} T(0,y,z,t) = T(W,y,z,t) &= f(s,t) \\ T(x,0,z,t) = T(x,L,z,t) &= f(s,t) \\ T(x,y,0,t) &= f(s,t) \\ T(x,y,H,t) &= T_b \end{aligned}$$

where s modifies the coordinates on the term to the left. To simplify the following computations, we define

$$u(r,t) = T(r,t) - f(s,t)$$

The equation becomes

$$\frac{\partial u}{\partial t} K\nabla^2 u$$

with initial condition

$$u(r,0) = T_i - f(s,t)$$

and boundary conditions

$$\begin{aligned} T(0,y,z,t) = T(W,y,z,t) &= 0 \\ T(x,0,z,t) = T(x,L,z,t) &= 0 \\ T(x,y,0,t) &= 0 \\ T(x,y,H,t) &= T_b \end{aligned}$$

with the knowledge that $f(x+D, y+D, H+D, t) = 0$, since those points are not part of the pan.

Separating variables, suppose

$$u(r,t) = XYZG$$

where X, Y, Z, and G are functions of x, y, z, and t. We quickly see that

$$\begin{aligned} u_t &= XYZG' \\ u_{xx} &= X''YZG \\ u_{yy} &= XY''ZG \\ u_{zz} &= XYZ''G \end{aligned}$$

Plugging these into the heat equation and dividing by $KXYZG$, we find

$$\begin{aligned} XYZG' &= K(X''YZG + XY'ZG + XYZ''G) \\ \implies \frac{G'}{KG} &= \frac{X''}{X} + \frac{Y''}{Y} + \frac{Z''}{Z} \end{aligned}$$

note that the left hand side of this equation is constant, so the right must be as well. Introducing λ^2, we write

$$\frac{G'}{KG} = -\lambda^2 \qquad \frac{X''}{X} + \frac{Y''}{Y} + \frac{Z''}{Z} = -\lambda^2$$

From the first equation, we immediately get the ODE

$$G' - G\lambda^2 = 0$$

For the others, let

$$\frac{X''}{X} = -\frac{Y''}{Y} - \frac{Z''}{Z} - \lambda^2 = -\mu^2$$

and from there let

$$\frac{Y''}{Y} = -\frac{Z''}{Z} - \lambda^2 + \mu^2 = -\nu^2$$

and

$$\frac{Z''}{Z} = -\lambda^2 + \mu^2 + \nu^2 = -\rho^2$$

We can now derive the following equations:

$$X'' + \mu^2 X = 0$$
$$Y'' + \nu^2 Y = 0$$
$$Z'' + \rho^2 Z = 0$$
$$\mu^2 + \nu^2 + \rho^2 = \lambda^2$$

and the one from before:

$$G' - G\lambda^2 = 0$$

The general solution G is $G = Ae^{-\lambda^2 Kt}$ for some constant A. The general solution for $X'' + \mu^2 X = 0$ is

$$X = c_1 \cos \mu x + c_2 \sin \mu x$$

Considering the boundary condition $X(0) = 0$, note that $c_1 = 0$. Considering $X(W) = 0$, we must have $c_2 \sin \mu W = 0$. Since $c_2 = 0$ is a trivial case, we focus on cases where $\sin \mu W = 0$. This happens when μW is an integer multiple of π. We note therefore that

$$\mu_m = \frac{m\pi}{W}$$

for positive integers m. Substituting these findings into the equation for X, we find solutions

$$X_m(x) = \sin(\frac{m\pi x}{W})$$

the same method leads to solutions for Y:

$$Y_n(y) = \sin(\frac{n\pi y}{L})$$

for for positive integers n.

For Z, the boundary conditions are $Z(0) = 0$ and $Z(H) = T_b$, so we let

$$Z = \alpha_l e^{\rho_l z} + \beta_l e^{-\rho_l z}$$

the boundary condition $Z(0) = 0$ gives us $\alpha_l = -\beta_l$. Choosing $\alpha_l = \frac{1}{2}$ we find

$$Z_l(z) = \sinh \rho_l z$$

the separated solution for u is thus

$$u_{mnl}(r,t) = A_{mnl}\sin(\mu_m x)\sin(\nu_n y)\sinh(\rho_l z)e^{-\lambda_{mnl}^2 Kt}$$

with

$$\lambda_{mnl}^2 = \mu_m^2 + \nu_n^2 + \rho_l^2 = \left(\frac{m\pi}{W}\right)^2 + \left(\frac{n\pi}{L}\right)^2 + \left(\frac{l\pi}{H}\right)^2$$

We can also form a linear superposition:

$$u(r,t) = \sum_{m=1}^{\infty}\sum_{n=1}^{\infty}\sum_{l=1}^{\infty} A_{mnl}\sin(\mu_m x)\sin(\nu_n y)\sinh(\rho_l z)e^{-\lambda_{mnl}^2 Kt}$$

with constants A_{mnl}. This general solution satisfies the boundary conditions on all sides except the top. To determine the values of the constants A_{mnl}, note that at $t = 0$ and therefore

$$T_i - f(s,0) = \sum_{m,=1}^{\infty}\sum_{n=1}^{\infty}\sum_{l=1}^{\infty} A_{mnl}\sin(\mu_m x)\sin(\nu_n y)\sinh(\rho_l z)$$

Let

$$b_m(y,z) = \sum_{n=1}^{\infty}\sum_{l=1}^{\infty} A_{mnl}\sin(\nu_n y)\sinh(\rho_l z)$$

and consider the resulting Fourier series:

$$T_i - f(s,0) = \sum_{m=1}^{\infty} b_m(y,z)\sin(\mu_m x)$$

Employing the usual method, we multiply both sides by $\sin(\mu_k x)$ and integrate:

$$\int_0^W (T_i - f(s,0))\sin(\mu_k x)dx = \sum_{m=1}^{\infty}\int_0^W b_m(y,z)\sin(\mu_m x)\sin(\mu_k x)dx$$

Take note that on the right hand side, the sines are orthogonal unless $k = m$, leaving

$$\begin{aligned}\int_0^W (T_i - f(s,0))\sin(\mu_m x)dx &= \int_0^W b_m(y,z)\sin(\mu_m x)\sin(\mu_k x)\\ &= \frac{b_m(y,z)W}{2}\end{aligned}$$

and so we have

$$b_m(y,z) = \frac{2}{W}\int_0^W (T_i - f(s,0))\sin(\mu_m x)dx$$

The same technique will produce the analogous

$$c_n(x,z) = \frac{2}{L}\int_0^L (T_i - f(s,0))\sin(\mu_n y)dy$$

for the opposite side.

We were eventually stumped by the vertical direction. We searched for a Fourier series solution, but were unable to use orthogonality relations on the sin multiple in the product solution, and ultimately decided to move on to other methods.

7.2 Numerical Solution

We implemented the method of finite differences to model the diffusion of heat through the batter and pan using the forward Euler method. We started in the two-dimensional case, by building a model for a vertical cross section of a rectangular pan, as follows:

Let U^t and C be arrays of dimension $M \times N$. Let c_a, c_p and c_b be the thermal conductivity constants of air, the pan material, and brownie batter. Let T_i stand for the initial temperature of the batter and pan, and let T_b be the baking temperature, which we assume to be constant. Finally, let $\Delta s = \frac{1}{N}$, and let w be the thickness of the pan material in units.

Schematically, we intend $U^t_{i,j}$ to hold the temperature of point (i,j) at time t and $C_{i,j}$ to hold the corresponding thermal conductivity constant. For an example of initial conditions, consider the 6×6 case with $w = 1$:

$$U^0 = \begin{pmatrix} T_b & T_b & T_b & T_b & T_b & T_b \\ T_b & T_i & T_i & T_i & T_i & T_b \\ T_b & T_i & T_i & T_i & T_i & T_b \\ T_b & T_i & T_i & T_i & T_i & T_b \\ T_b & T_i & T_i & T_i & T_i & T_b \\ T_b & T_b & T_b & T_b & T_b & T_b \end{pmatrix} \qquad C = \begin{pmatrix} c_a & c_a & c_a & c_a & c_a & c_a \\ c_a & c_p & c_b & c_b & c_p & c_a \\ c_a & c_p & c_b & c_b & c_p & c_a \\ c_a & c_p & c_b & c_b & c_p & c_a \\ c_a & c_p & c_p & c_p & c_p & c_a \\ c_a & c_a & c_a & c_a & c_a & c_a \end{pmatrix}$$

To track the evolution of U through time, we define the following partial derivatives for points (i,j,τ), with time-coordinate τ:

$$u_t \approx \frac{u^{\tau+1}_{i,j} - u^{\tau}_{i,j}}{\Delta t}$$
$$u_{xx} \approx \frac{u^{\tau}_{i+1,j} - 2u^{\tau} i,j + u^{\tau}_{i-1,j}}{(\Delta s)^2}$$
$$u_{yy} \approx \frac{u^{\tau}_{i,j+1} - 2u^{\tau}_{i,j} + u^{\tau}_{i,j-1}}{(\Delta s)^2}$$

Given these, the heat equation

$$u_t = c(u_{xx} + u_{yy})$$

becomes

$$\frac{u_{i,j}^{\tau+1} - u_{i,j}^{\tau}}{\Delta t} = c\left(\frac{u_{i+1,j}^{\tau} - 2u_{i,j}^{\tau} + u_{i-1,j}^{\tau}}{(\Delta s)^2} + \frac{u_{i,j+1}^{\tau} - 2u_{i,j}^{\tau} + u_{i,j-1}^{\tau}}{(\Delta s)^2}\right)$$

from which we can derive

$$u_{i,j}^{\tau+1} = u_{i,j}^{\tau} + c \cdot \frac{\Delta t}{(\Delta s)^2}\left(u_{i+1,j}^{\tau} + u_{i-1,j}^{\tau} - 4u_{i,j}^{\tau} + u_{i,j+1}^{\tau} + u_{i,j-1}^{\tau}\right)$$

Δt, the length of a time step, can be chosen arbitrarily but must obey the relation

$$\Delta t \leq \frac{(\Delta s)^2}{2c} \quad \text{[Horak, 2005]}$$

for $c = c_a, c_p, c_b$ for a stable solution. Our next step was to design an analogous scheme for the three-dimensional problem. Using the partial derivatives

$$\begin{aligned}
u_t &\approx \frac{u_{i,j,k}^{q+1} - u_{i,j,k}^{q}}{\Delta t} \\
u_{xx} &\approx \frac{u_{i+1,j,k}^{q} - 2u^q i,j,k + u_{i-1,j,k}^{q}}{(\Delta s)^2} \\
u_{yy} &\approx \frac{u_{i,j+1,k}^{q} - 2u_{i,j,k}^{q} + u_{i,j-1,k}^{q}}{(\Delta s)^2} \\
u_{zz} &\approx \frac{u_{i,j,k+1}^{q} - 2u_{i,j,k}^{q} + u_{i,j,k-1}^{q}}{(\Delta s)^2}
\end{aligned}$$

we arrived at the 'update function'

$$u_{i,j,k}^{q+1} = u_{i,j,k}^{q} + c\frac{\Delta t}{(\Delta s)^2}\left(u_{i+1,j,k}^{q} + u_{i,j+1,l}^{q} + u_{i,j,k+1}^{q} + u_{i-1,j,k}^{q} + u_{i,j-1,k}^{q} + u_{i,j,k-1}^{q} - 6u_{i,j,k}^{q}\right)$$

and successfully replicated heat buildup around the edges and corners of the brownie pan, however the three-dimensional solution was too computationally unwieldy to generate the quantity of data we would have liked.

8 Testing and Results

8.1 Parameter Fitting

Using values $m = 10, n = 10, T_b = 176, T_i = 15, \Delta t = 1/50000, c_a = .024$, and $c_b = .58$, we varied the conduction coefficient for the pan to attempt to determine the best pan material

for even heating around the edges. Our reasoning in these choices was that $176°C \approx 350°F$ and $15°C \approx 60°F$, representing reasonable estimates for baking temperature and room temperature. We chose $c_a = .024$ because this is the approximate conduction coefficient of air, although conductivity increases with temperature, so this may be an underestimate in an oven setting. $c_b = .58$ is approximately the conductivity of water and potatoes, which we estimated to be reasonable approximations for brownies.

8.2 Optimal Thermal Conductivity

We first compared glass ($c_p = 1.05$) to a much less conductive material ($c_p = 0.05$) and aluminum ($c_p = 215$) in a 10x10 grid, and reached the following states in the lower left hand corners after 10000 time steps (on next page for visual comparison):

$$c_p = 0.05: \begin{pmatrix} 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \\ 176 & 149.04 & 152.22 & 153.46 & 153.88 & 154.88 & 153.46 & 152.22 & 149.05 & 176 \\ 176 & 130.22 & 132.89 & 133.93 & 134.26 & 134.26 & 133.94 & 132.90 & 130.23 & 176 \\ 176 & 119.73 & 119.45 & 119.08 & 118.82 & 118.82 & 119.08 & 119.45 & 119.73 & 176 \\ 176 & 114.59 & 111.31 & 109.25 & 108.24 & 108.24 & 109.25 & 111.31 & 114.59 & 176 \\ 176 & 112.76 & 107.61 & 104.24 & 102.57 & 102.58 & 104.24 & 107.61 & 112.76 & 176 \\ 176 & 113.96 & 107.97 & 103.68 & 101.55 & 101.55 & 103.68 & 107.98 & 113.97 & 176 \\ 176 & 120.22 & 112.47 & 107.11 & 104.67 & 104.67 & 107.12 & 112.47 & 120.23 & 176 \\ 176 & 137.07 & 120.13 & 113.46 & 111.28 & 111.28 & 113.46 & 120.13 & 137.07 & 176 \\ 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \end{pmatrix}$$

$$c_p = 1.05: \begin{pmatrix} 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \\ 176 & 174.52 & 173.06 & 171.97 & 171.39 & 171.39 & 171.97 & 173.06 & 174.52 & 176 \\ 176 & 173.22 & 170.47 & 168.43 & 167.34 & 167.34 & 168.43 & 170.48 & 173.22 & 176 \\ 176 & 173.22 & 170.47 & 168.43 & 167.34 & 167.34 & 168.43 & 170.48 & 173.22 & 176 \\ 176 & 172.26 & 168.57 & 165.83 & 164.36 & 164.36 & 165.83 & 168.58 & 172.26 & 176 \\ 176 & 171.76 & 167.59 & 164.48 & 162.82 & 162.82 & 164.48 & 167.29 & 171.77 & 176 \\ 176 & 171.76 & 167.64 & 164.55 & 162.90 & 162.90 & 164.55 & 167.59 & 171.79 & 176 \\ 176 & 172.33 & 168.72 & 166.03 & 164.59 & 164.60 & 166.03 & 168.72 & 172.33 & 176 \\ 176 & 173.32 & 170.70 & 168.74 & 167.69 & 167.70 & 168.74 & 170.70 & 173.32 & 176 \\ 176 & 174.61 & 173.32 & 172.34 & 171.82 & 171.82 & 172.34 & 173.32 & 174.61 & 176 \\ 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \end{pmatrix}$$

$$c_p = 215: \begin{pmatrix} 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \\ 176 & 174.84 & 173.52 & 172.54 & 172.01 & 172.02 & 172.54 & 173.53 & 174.84 & 176 \\ 176 & 173.82 & 171.36 & 169.52 & 168.53 & 168.53 & 169.52 & 171.36 & 173.82 & 176 \\ 176 & 173.08 & 168.78 & 167.31 & 165.99 & 166 & 167.32 & 169.79 & 173.08 & 176 \\ 176 & 172.7 & 168.99 & 166.21 & 164.72 & 164.72 & 166.21 & 169 & 172.71 & 176 \\ 176 & 172.74 & 169.08 & 166.33 & 164.86 & 164.87 & 166.34 & 169.08 & 172.75 & 176 \\ 176 & 173.19 & 170.04 & 167.68 & 166.41 & 166.42 & 167.68 & 170.04 & 173.19 & 176 \\ 176 & 173.98 & 171.74 & 170.07 & 169.17 & 169.17 & 170.07 & 171.75 & 173.98 & 176 \\ 176 & 174.99 & 173.99 & 173.21 & 172.79 & 172.8 & 173.21 & 173.99 & 174.99 & 176 \\ 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 & 176 \end{pmatrix}$$

In each of these cases, the open top of the pan is oriented upward. Observing results of the simulation, we were surprised by the relatively small difference between glass and aluminum, despite the apparently large difference in conductivity. We took the variances of the batter in the three pans above and found them to be 103 and 8.71, and 6.76, respectively. The non-conductive material clearly performed badly in preventing heat buildup, however, an F-test ($d_1 = d_2 = 36$) found the difference between the glass and aluminum variances to be insignificant with a p-value of 0.23. From this we concluded

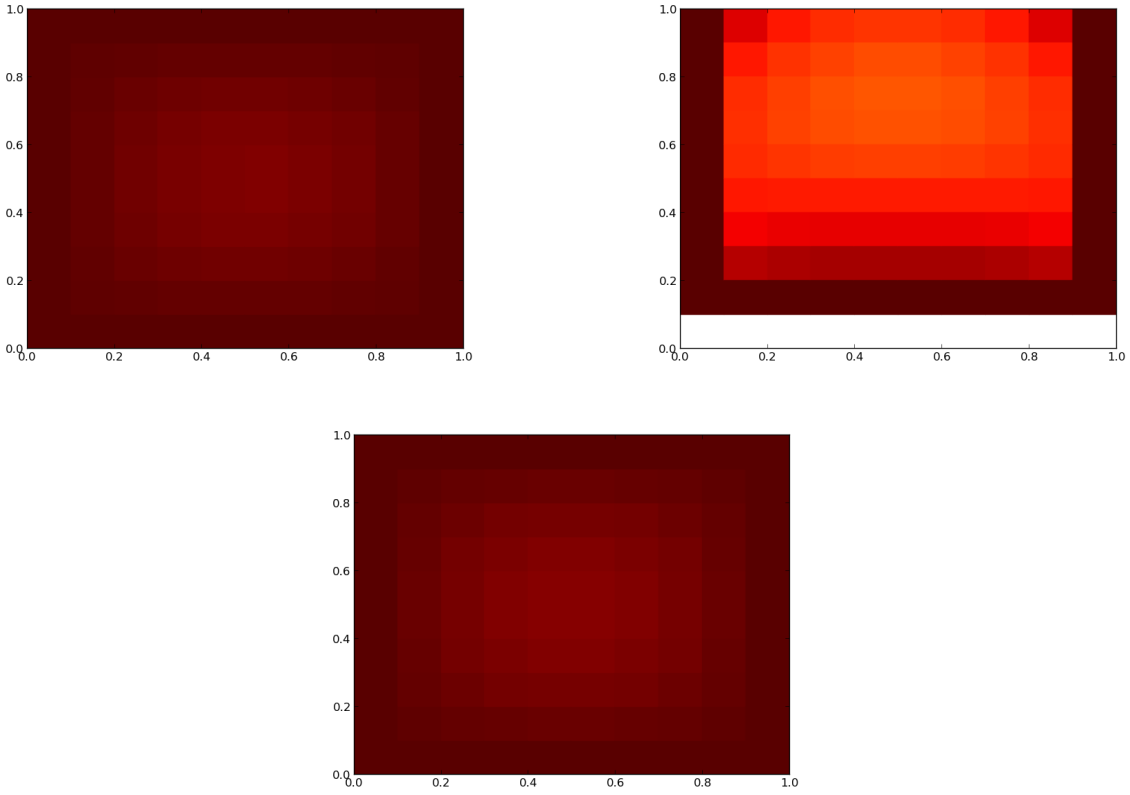

Figure 11: Comparison of different K: clockwise from top left: 215 (Aluminum), 0.05 (low conductivity), 1.05 (Water)

from this that, assuming a minimal level of conductivity (on par with glass), an increase in conductivity does little to increase or allay heat buildup around the edges of the pan.

9 Conclusion

Our essential plan was to use Arnold's cat map to generate batches of random shape boundaries, then extend the boundaries upward at a linear slope to form three dimensional 'pans' to be evaluated for effectiveness using finite differences. We would have then used a genetic algorithm to cull the most ineffective pans from the population. We ran into unexpected difficulty when faced with the problem, given an arbitrary and possibly disjoint boundary, of deciding whether points in the oven lie within the boundary or outside it. This undermined our ability to differentiate between batter and air in our model oven. The next step would have been to generalize the finite difference scheme for arbitrary shapes, but due to time and practical constraints we were unable to implement our final plans.

10 Letter to the Magazine:

To the Venerable Editors of Brownie Gourmet:
Dear friends,
We write to you as a group of mathematics students who hold a long-time interest in your magazine. Believe it or not, we gather frequently to brainstorm ways to improve the science of brownie baking, and thought we would share with you the fruits of our latest session.

It occurred to us that industry manufacturers of brownie pans may have overlooked some unorthodox, but potentially useful pan designs, such as those which could be randomly generated by chaotic maps. Though the vast majority of chaotic brownie pans will be ineffective for optimizing heat distribution within a pan, and also difficult to pack efficiently within the oven, it is our belief that generic algorithms could be applied to populations of chaotic pans in order to reach an optimally suited population. Accordingly, we have designed for your magazine a system by which batches of chaotic pans can be evaluated for fitness using finite difference solutions to the heat equation in combination with packing constraints, then selected for fitness and recombined for the next generation of better fit pans.

Though some minor kinks in our method remain, it is our hope that you will expand on our work to produce and market your own line of pans, so that you may capitalize on this previously unexplored market niche. We provide this framework to you, with no expectation of recognition or remuneration, out of friendly concern for BG's long-term financial sustainability and admiration of your work. As long-time subscribers who look *forward* to paying the monthly fee to support your enterprise, we could not possibly ask for so much as a free magazine.

In essence, our method uses Arnold's cat map and the Chirikov Standard map to generate populations of random, equal-area boundaries in two dimensions. The sides of the boundaries are then linearly extended into three dimensions to for a chaotic 'pan'. The pans are evaluated for evenness of heat distribution using finite difference solutions to the heat equation, and evaluated for packing efficiency according to their 'effective area,' a notion we define in greater detail in the attached paper. Mimicking natural selection with a genetic algorithm, the least effective pans are culled from the population while the most effective are recombined into the next generation. After many repetitions of this cycle, an optimal population is produced.

We hope that our work is enlightening and valuable to you. If you have trouble understanding or implementing our methods, we hope you will consider hiring unemployed college graduates in math.

Thank you for reading – as always, we value the opportunity to express our appreciation for your magazine.

Sincerely,
Wyatt Alt, Atsuya Kumano, Laura Lyman

References

David Bleecker and George Csordas. *Basic Partial Differential Equations*. International Press of Boston, 1997.

B. Chirikov and D. Shepelyansky. Chirikov standard map. *Scholarpedia*, 3(3):3550, 2008.

Robert C Hilborn. *Chaos and nonlinear dynamics: an introduction for scientists and engineers*. Oxford Univ. Press, New York, NY, 1994.

Verena Horak. Parallel numerical solution of 2-d heat equation. *Parallen Numerics*, 2005: 47–56, 2005.

Melanie Mitchell. *An Introduction to Genetic Algorithms*. A Bradford Book, 1998.

Edward Ott. *Chaos in dynamical systems*. Cambridge University Press; New York, NY, 1993.

William H. Press, Saul A. Teukolsky, William T. Vetterling, and Brian P. Flannery. *Numerical Recipes in C: The Art of Scientific Computing*. Cambridge University Press, 1992.

Eric W. Weisstein. Arnold's cat map. Website. URL `http://mathworld.wolfram.com/ArnoldsCatMap.html`. From MathWorld–A Wolfram Web Resource. Viewed on 2013 February 1.