Packing Spheres on a Sphere: An Introduction to Experimental Mathematics

Noah Giansiracusa

November 2007

Abstract

In this talk we describe work of Henry Cohn et al. on the sphere packing problem. Cohn, who received his Ph.D. in 2000 under Noam Elkies and has since been at Microsoft Research, has been using computers to aid his research for a while. In the case of his thesis, he proved that the existence of functions with certain properties would guarantee sphere packing bounds that drastically improved those previously known in many dimensions. Using the help of computers he found such functions, and his work was published in the Annals. More recently, Cohn has been using computers to survey the mathematical landscape of problems such as sphere packing in order to gain a better conceptual understanding and possibly discover new and beautiful mathematical objects. This latter research has been rejected from all the journals to which it was submitted, except for journals focusing exclusively on experimental math. We will summarize some of Cohn’s work, illustrate the wide range of mathematical subjects involved in sphere packing, and hopefully convince some of those in attendance that math should not just be about proof, but about conceptually understanding phenomena as well.

Introduction

This talk, at the most superficial level, is about sphere packing. For those who have not seen this elementary yet extremely difficult and multi-disciplinary subject, we will provide a little background and discuss some of the progress that has been made, as well as the vast mountain of challenges that remain. For those aware of the classical results in sphere packing, we will describe some of the more recent approaches by Henry Cohn and his collaborators.

In addition to providing a brief survey of sphere packing and an exposition of Cohn’s papers, we would like to use this area of research to illustrate a more philosophical aspect of math, namely the interaction of computers with pure math. Computers have been used in sphere packing for everything from helping to write an inhumanly long proof (similar to the infamous Four Color problem), to finding functions whose existence alone completes a rigorous hand-written proof, to searching for new sphere packings and phenomena related to sphere packing that may later be analyzed by hand.
One could argue that this last example comes the closest to playing the role in pure math that experimental physics plays in theoretical physics. Many of the problems involved in sphere packing seem intractably difficult—just as the universe seems too complicated to admit a complete and unifying theory—so rather than simply giving up all hope, we can try to explore the math behind sphere packing heuristically. We can strive to describe certain phenomena without being able to completely articulate them, hoping that later generations may have the necessary theoretical and technical machinery to develop a precise theory that explains the experimentally witnessed phenomena from our generation.

Therefore, the objectives of this talk could be categorized according to the following trichotomy:

- Explain some background and recent approaches to the sphere packing problem
- Discuss the complex and controversial role of computers in mathematics
- Discuss what it means for mathematics to be “experimental” and try to illustrate that precisely stated theorems and rigorous proofs are not the only way (and perhaps not even the best way) to understand certain phenomena.

Regarding the third point, we mainly wish to demonstrate the importance of a conceptual understanding where a precise theorem and proof are lacking, but as an amusing example of the converse, namely a carefully stated theorem and proof that seem to yield almost no understanding whatsoever, consider the following example found in Manin’s book on cubic forms[1]:

**Theorem.** Every rational number is a sum of three rational cubes.

**Proof.** For \(a \in \mathbb{Q}\),

\[
\begin{align*}
\frac{a^3 - 3^6}{3^2a^2 + 3^3a + 3^3} &+ \left(\frac{-a^3 + 3^5a + 3^5}{3^2a^2 + 3^3a + 3^3}\right)^3 + \left(\frac{a^3 + 3^7a}{3^2a^2 + 3^3a + 3^3}\right)^3.
\end{align*}
\]

Surely, there is more to math than the mere veracity of a proof. The attempts to find it, methods used to construct it, and mathematical phenomena underlying are all part of the proof, even if only the result of this process is ever published.

**Sphere Packing in \(\mathbb{R}^n\)**

An extensive reference on sphere packing is the book of Conway and Sloane[2]. We treat very briefly the basics. Essentially, the sphere packing problem is to determine the maximal density one can obtain by packing \(\mathbb{R}^n\) with non-overlapping balls of a given radius. By density, we mean the total volume covered by the balls\(^1\) divided by the volume of the ambient Euclidean space. Since these are both infinite in general, one needs to look at this density in a compact region and take the limit as these compact regions expand to cover all

\(^1\)Recall that the volume of a radius \(r\) ball in \(\mathbb{R}^n\) is \(\pi^{n/2}r^n/\Gamma(n/2 + 1)\)
of $\mathbb{R}^n$. This limit need not exist for pathological packings, but one can show that a maximal packing density exists regardless, and all the specific packings typically considered are so regular that no problems arise regarding this issue. Note that the radii of the spheres are irrelevant since we can re-scale everything without changing the packing or its density. One typically studies packings modulo isometries and scaling.

We can consider the sphere packing problem in Euclidean space $\mathbb{R}^n$ for any dimension. Some information for low dimensional cases is summarized below:

### $n=1$ Intuition: Laying match sticks in a row.

**Results:** This case is trivial since one can cover the entire real line with non-overlapping intervals of fixed length. The maximal density is 1.

### $n=2$ Intuition: Placing pennies on a tabletop.

**Results:** The best packing has been known for thousands of years, but it was first proven in the early 20th century. To obtain it, take a regular hexagonal tiling of the plane and inscribe a disk in each hexagon. The maximal density is $\frac{\pi}{\sqrt{12}} \approx 0.9069$. Interestingly, this is precisely the structure that bees have developed for their hives (see Figure 1).

### $n=3$ Intuition: Storing cannon balls in a large crate.

**Results:** Again the best packing has been known for a while, but this time the proof only came in the 1990s and even to this day it remains somewhat controversial due to the heavy reliance on computers to break the problem into many cases and perform many computations that are too complex to be undertaken by hand. In this dimension the solution is not exactly unique, but essentially all the maximal packings are obtained by stacking the 2-dimensional hexagonal packings on top of each other. These packings have density $\frac{\pi}{\sqrt{18}} \approx 0.7405$. See Figure 2 for a rendering of the most natural such layering (namely the one that forms a 3-dimensional lattice).

Let us discuss the 3-dimensional case a little further. The problem of packing spheres in $\mathbb{R}^3$ was first posed by Johannes Kepler in 1611, so it is often referred to as the “Kepler Conjecture.” In 1831 Gauss was able to prove that the packing pictured in Figure 2 is optimal among lattice packings, but this still leaves out a lot of possible packings. In 1900 Hilbert included this problem in his famous list of unsolved problems (as part of the 18th). Hales’ proof is 282 pages long, and as he put it himself, “every aspect of it is based on even longer computer calculations.” The proof involves many intricate geometric and combinatoric arguments, subtle estimates and approximations, and a brute force exhaustion of 5,000 combinatorial cases that arise. One of these cases became a Ph.D. thesis for a student of Hales, and to solve the remaining 4,999 Hales had to solve roughly 100,000 separate optimization problems. A key insight in the proof is realizing how to recognize these as linear programming problems rather
Figure 1: The hexagonal lattice produces the optimal packing in $\mathbb{R}^2$

Figure 2: The optimal lattice packing in $\mathbb{R}^3$
than arbitrary optimization problems, so that a computational solution would be at least possible, even if it is still far from aesthetically pleasing.

For dimensions above 3, the sphere packing problem seems too difficult to solve entirely. Usually one simply aims to provide upper and lower bounds on the maximal packing density. For instance, there is a lower bound of $2^{-n}$ that holds for each $\mathbb{R}^n$. The proof is simple, though it requires the axiom of choice: continue to place spheres randomly, as long as they do not overlap, until there is no room left to place any more. If this is done with balls of radius $r$, then it is easy to see that all of $\mathbb{R}^n$ will be covered by balls of radius $2r$ centered at the same locations as before. Let us call these two packings $S_r$ and $S_{2r}$, respectively. Since volume scales according to $2^n$, we get the desired bound:

$$\text{Density}(S_r) = \frac{\text{Volume}(S_r)}{\text{Volume}(\mathbb{R}^n)} \geq \frac{\text{Volume}(S_r)}{\text{Volume}(S_{2r})} = \frac{1}{2^n}.$$ 

This shows that there always exists a packing with density at least $2^{-n}$ (and it is not hard to see that we can take it to be a lattice), but we actually have no way to find it for large values of $n$. The best known method for constructing a sequence of lattices that comes closest to attaining this bound uses quite sophisticated methods from number theory and algebraic geometry\([3]\). It produces lattices of density $2^{-cn}$ for $c \approx 1.39$.

Let us turn now to methods for finding an upper bound. The nicest type of packing to work with is a lattice packing, but with only slightly more complexity we can generalize to a period packing, which is by definition a union of finitely many translates of a given lattice. It has been known for a while that period packings come arbitrarily close in density to the maximal density of any packing, so any theorem proving an upper bound on packing densities can immediately reduce to the case of a periodic packing. The best known upper bounds in dimensions 4 through 36 were proven in Henry Cohn’s Ph.D. thesis with Noam Elkies\([4]\). The main theorem they prove is the following:

**Theorem.** Suppose $0 \neq f : \mathbb{R}^n \to \mathbb{R}$ is a function to which the Poisson summation formula applies and such that 1) $f(x) \leq 0$ for $|x| \geq 1$ and 2) the Fourier transform $\hat{f}$ of $f$ is non-negative. Then the density of any sphere packing in $\mathbb{R}^n$ is bounded above by $\frac{\pi^{n/2} f(0)}{2^n (n/2)! f(0)}$. 

5
The idea is to fix one of \( f(0) \) and \( \hat{f}(0) \), and view extremizing the other as an infinite-dimensional linear programming problem. The approach Cohn takes requires some subtle analysis. Since the hypotheses and conclusion of this theorem are invariant under rotating \( f \), we can replace \( f \) with the average of its rotations and therefore assume it has radial symmetry. To find a function \( f \) that provides a good bound, Cohn considers a linear combination of a certain class of functions that form a basis for the radial eigenfunctions of the Fourier transform. It turns out that prescribing a root at the origin and letting the other roots roam subject to certain conditions and minimizing the distance to the last sign change of this function produces functions that provide better bounds than any previously known for dimensions 4 through 36.

The proof of the principal theorem is completely rigorous. The analysis indicating which types of functions would provide a good bound and where to look for them is also rigorous. A computer was needed to perform the optimization that explicitly finds the functions, but once such a function was found, the fact that it proves an upper bound is also completely rigorous. This differs quite strongly from Hales’ proof of the Kepler Conjecture in which significant portions of the proof itself are computer written/analyzed. However, in both these cases one could argue that the “thinking” was done by a human who discovered the ideas underlying the results and the computer only carried out certain calculations, whereas later we shall see a case in which the computer in a sense thinks up the interesting math and the human endeavors to perform calculations and prove properties about what the computer has discovered.

**Discrete sphere packing**

In addition to the evident difficulty of sphere packing and its historical tenure (which, as in the case of Fermat’s last theorem, helps an unsolved math problem gain widespread popularity), and the fact that Hilbert specifically listed the \( \mathbb{R}^3 \) case as an important problem (which is a good indication of its theoretical significance), solutions to variants of the sphere packing problem actually have practical importance as well—and by “practical” we mean useful even to (gasp!) non-mathematicians!

For instance, let us take a quick detour to the world of coding theory. Suppose we are given two people, WLOG Alice and Bob, such that Alice wishes to transmit a message through some line of communication to Bob. A common schematic for this system is as follows:

1. Alice translates her message from English to a more natural alphabet for transmission, typically binary if computers are involved.

2. The binary message zips across unimaginable distances at nearly the speed of light!

3. Bob receives the binary message and translates it back to English.
Under ideal conditions the method of translation is not terribly important: one could use the natural composition

\[
\text{ASCII} \rightarrow \{0, \ldots, 255\} \rightarrow \mathbb{Z}_2^8
\]

so that each letter in Alice’s message becomes an 8 digit long binary sequence. However, things become more complicated—and more interesting!—when we allow for the possibility of errors during the second phase of communication. Specifically, suppose there is a small but nonzero probability that each binary digit flips during transmission. If Alice sends the letter “A”, which is encoded as “65” in ASCII, hence “1000001” in binary, and just one of the digits flips, e.g. Bob receives “1001011”, he will mistakenly think Alice sent the letter “E”. If every tiny error corrupts the message, we had better rethink our communication system. The key is that our method of translating, or in other words encoding, our message is too greedy. If we are willing to sacrifice some efficiency in transmission, then we can help reduce the effect of errors occurring due to noise.

For instance, we can append one further map, which we call 3-fold repetition, to the above composition. This map \(\mathbb{Z}_2^8 \rightarrow \mathbb{Z}_2^{24}\) sends a word \(a_1a_2\cdots a_8\) to the longer word \(a_1a_1a_1a_2a_2a_2\cdots a_8a_8a_8\). This redundancy adds security, because if Alice sends \(A = 111,000,000,000,000,000,000,111\) and Bob receives the word \(110,000,010,000,100,000,000,101\), then he can look at each triple and take the binary digit to be whichever number occurs at least 2 out of 3 times.

It is nice to be able to correct errors like this, but it is also distressing to realize that we could have sent 3 three letters using the original system using the same bandwidth as one letter in this new system. The issue is that when we are defining our code, i.e. choosing which possible binary words to use for transmission, we want to allow many words so as to enhance efficiency, but few words in order to ensure reliability. Moreover, there are good and bad ways to choose which words to allow even after we decide how many we want. For instance, if we are sending binary triples \(a_1a_2a_3, a_i \in \mathbb{Z}_2\), then clearly it is easier to correct errors in the code \(\{000,111\}\) than the code \(\{000,001\}\) because in the former we are guaranteed to decode the right word if not more than one error occurs, whereas in the latter even a single bit flipping could cause us to misread the message.

How can we describe mathematically why one code is better than the other? The key is to interpret a code as a discrete sphere packing! We can turn our space \(\mathbb{Z}_2^n\) of possible codewords into a metric space using the Hamming metric \(d(a_1\cdots a_n, b_1\cdots b_n) = \#\{i \mid a_i \neq b_i\}\) which declares the distance between two words to be the number of digits in which they differ. A code is a finite set of points in \(n\)-dimensional space, and with the Hamming metric we can imagine placing a small sphere at each point of the code and expanding all the radii until two of the spheres begin to touch. If we have chosen a priori the number of codewords allowable (i.e. the number of spheres to place), then the best code is the one whose corresponding discrete sphere packing has the greatest density. In Figure 4 we see a simple picture illustrating this idea for \(\mathbb{Z}_2^3\). If we
are using two spheres, then placing them at opposite corners covers the entire space, whereas centering the spheres at two adjacent points only covers 2 out of the 8 possible points. This corroborates the error-correcting capabilities of the two codes mentioned in the preceding paragraph.

This example may seem somewhat trivial, but imagine packing a million spheres in $\mathbb{Z}_2^{100}$. It is not at all clear how to do this optimally, and in the real world huge codes are used all the time, such as in internet communication, computer storage, satellite data transmission, etc.

Spherical Sphere Packing

We have seen that upper and lower bounds for sphere packing can sometimes be proven abstractly without actually constructing a sphere packing. It turns out that for lower bounds, at least in low dimensions, the best known bounds are all proven by explicitly producing a packing of a certain density. The locations of the sphere centers often form beautiful objects studied in many areas of math. For instance, in dimension 8 the best known packing is the $E_8$ lattice that comes up in discrete reflection groups, Lie groups/algebras, coding theory, string theory, etc\(^2\). It has been proven that $E_8$ gives a maximal density among lattice packings in $\mathbb{R}^8$, and it is believed to be optimal among all packings.

Perhaps an even more striking case is dimension 24. This dimension is large enough that it is difficult to study the geometry of sphere packings very carefully unless we have a very beautiful and symmetric object describing the spheres. In this case, the amazing object given to us is the Leech lattice. Like $E_8$ it

\(^2\text{In 1982 Michael Freedman discovered a bizarre manifold whose intersection form is given by the } E_8 \text{ lattice. It admits no smooth structure and in fact is not even triangulable.}\)
shows up in many areas of mathematics, and it too is believed to be the densest possible sphere packing in its dimension. In 2004 Cohn and Kumar proved\cite{5} that it is the unique optimal packing among lattice packings. A plot of the best currently known upper and lower bounds for low dimensional sphere packing is pictured in Figure 5. Here it is even more evident how striking dimension 24 really is. The existence of the Leech lattice in $\mathbb{R}^{24}$ is related to the fact that $1^2 + 2^2 + \cdots + 24^2$ is a perfect square. Twenty four is the only integer greater than 1 with this property.

Given that such amazing objects appear to solve the sphere packing problem in certain dimensions, one may turn the problem around from the perspective of the Kepler Conjecture: instead of trying to find a proof that a packing we all believe to be optimal really is, why don’t we forget about proofs and just look for interesting packings themselves? If we could find good packings somehow, perhaps the geometry of the packings will provide us with a new mathematical object useful in other areas of math! This seems quite intriguing, so we are naturally led to the following question: how do we actually go about finding good packings?

One idea is as follows. To have a good packing is to have the sphere centers spaced out as far as possible so that the radii can expand as much as possible without the spheres overlapping. But when working in an unbounded space such as $\mathbb{R}^n$ this does not make too much sense. So let us try a related sphere packing problem, namely packing spheres in the compact space $S^{n-1} \subset \mathbb{R}^n$. A finite subset of $S^{n-1}$ is called a spherical code, and one that maximizes the minimal distance between any two points is called optimal. Since we are concerned with
the distance between points on the sphere, and this distance can be taken to be a geodesic along the sphere or a straight line through the sphere, we can view this problem as packing spheres around a sphere, or packing spherical caps of one lower dimension on the sphere itself. For example, with $S^2 \subset \mathbb{R}^3$ we can think of distributing points as far apart as possible, or packing small marbles around one large marble, or trying to cover $S^2$ as densely as possible with little yarmulkes. One can look at Figures 6 and 7 to see an example of an optimal spherical code found mathematically, and one found in nature (neither one has been proved to be optimal).

Although different from the standard Euclidean sphere packing problem, this spherical sphere packing has much of the same difficulty and intrigue as the classical case, and often the objects that describe good packings on $S^{n-1}$ are related to the objects that describe good packings in $\mathbb{R}^n$. For instance, the minimal vectors in a lattice (i.e. the set of nonzero vectors whose distance to the origin is minimal within the lattice) can be normalized to give a collection of points on the sphere, and for lattices such as $E_8$ and Leech with good packing properties, the corresponding spherical codes often have good properties as well.

As alluded to above, this compact setting allows us to make precise the notion of spacing out points as far as possible. If we want to distribute $N$ points on $S^{n-1} \subset \mathbb{R}^n$, how can we do so in a way that makes the minimal distance between any pair maximal? One approach is to borrow some ideas from nature. We can think of each point on the sphere as a positively charged point-mass. If we dump a bunch of points down randomly, they will want to repel each other, and the result should be a nicely spread out spherical code.

Given a decreasing function $f : (0, 4] \rightarrow [0, \infty)$, we can define the $f$-potential energy of a spherical code $C \subset S^{n-1}$ to be $\sum_{x, y \in C; x \neq y} f(|x - y|^2)$. For a fixed
number of points $N = \#C$ and dimension $n$ we can ask which spherical codes minimize potential energy for a given potential function. Furthermore, we can ask which codes do so for an entire family of potential functions simultaneously. Cohn and Kumar[6] call a spherical code \textit{universally optimal} if it minimizes potential energy for all completely monotonic potential functions$^3$. Thus, we are looking at “nice” potential functions in the sense that they are smooth, decreasing, convex, etc., but still quite general. Examples of such functions are the inverse power laws $f(r) = 1/r^s$ for any fixed $s$. Note that a universally optimal code is in particular an optimal spherical code, because as $s \to \infty$ the minimal distances dominate the inverse power potential function.

As an example of the effect of letting the potential function change, consider 5 points on $S^2$. One configuration which minimizes certain potential functions is a single point placed at the north pole and four points arranged in a square parallel to the equator. As one varies the potential function, the square moves up or down the sphere, depending on whether it is more efficient to be further from the north pole or further from the other points within the square, so this configuration is not universally optimal. However, it is not hard to show that in any dimension the vertices of an $n$-simplex form a universally optimal code. Cohn and Kumar were able to prove that several previously known optimal spherical codes are in fact universally optimal, and in each case the code corresponded to a well-known and intrinsically interesting object. For instance, on $S^2$ the icosahedron is universal, and on $S^{23}$ the minimal vectors of the Leech lattice are universal. One interesting family of universal optima come from the

\footnote{Recall that a $C^\infty$ function $f : I \to \mathbb{R}$ is completely monotonic if $(-1)^k f^{(k)}(x) \geq 0$ for all $x \in I$, $k \geq 0$.}
del Pezzo surfaces in algebraic geometry. The only known universal optima in dimensions \( n = 5, 6, 7, 8 \) aside from infinite families such as the \( n \)-simplex have \( N = 16, 27, 56, 240 \) points, respectively. They are obtained as follows. Let \( X \) be the del Pezzo surface\(^4\) of degree \( d = 1, 2, 3, 4 \), which we can think of as \( \mathbb{P}^2 \) blown-up at \( r = 8, 7, 6, 5 \) points. Then \( X \) has \( N = 240, 56, 27, 16 \) exceptional curves\(^5\). The orthogonal complement \( \omega_X^\perp \) of the canonical divisor is negative definite, and the Picard group is free of rank \( r + 1 \), so after changing sign this induces the structure of a Euclidean space \( \mathbb{R}^r \) on \( \omega_X^\perp \otimes \mathbb{Z} \mathbb{R} \). Projecting the divisor classes of the exceptional curves from Pic(\( X \)) \( \otimes \mathbb{Z} \mathbb{R} \) to this space, then scaling so that they lie on \( S^{r-1} \subset \mathbb{R}^r \) gives a universally optimal code.

This seems to indicate that searching for new universal optima may lead to the discovery of fascinating new objects. Moreover, since we have phrased the problem in terms of minimizing potential energy, it is not difficult to imagine writing a computer program to help with the search. Cohn et al.[7] did massive computer searches in dimensions up through 64 looking for configurations that minimize various potential laws by randomly distributing points and then using a gradient descent optimization algorithm. Even though the class of potential functions, namely completely monotonic functions, is uncountably large, it can be shown either with a compactness argument from model theory or a classical argument in analysis that it is enough to look at the functions \( f(r) = (4 - r)^k \) for \( k \in \{0, 1, 2, \ldots\} \).

These computer searches turned up many interesting configurations, but when the computer decides the points have converged to a potential minimum, the coordinates it returns for the points are typically quite messy and random looking. The computer has done its job, namely finding possibly interesting configurations, but it is up to the mathematicians to carefully analyze them and decide exactly how interesting they really are. Figures 8 and 9 show the Gram matrix (i.e. the matrix of dot products between all points in the configuration) first as the points were returned by the computer in random order, and then after they have been manually rearranged with the structure/symmetries in mind.

So how does one go about recognizing structure and studying the objects the computer produces? One can often use a program such as PARI to recognize the coordinates as algebraic numbers, which is sometimes helpful. More geometrically, a very useful piece of information to know about a configuration is its group of symmetries. To each spherical code \( C \) we can create an edge-weighted complete graph on \#\( C \) vertices in which the vertices correspond to points of the code and the “color” of each edge is the inner product of the two points representing its endpoints. The automorphism group of this graph is exactly the symmetry group of the spherical code, so using a program such as Nauty[8] one can obtain all sorts of useful information about a configuration, such as the size of its symmetry group, the number of orbits, etc.

Another handy tool if one is interested in relating the objects the computer

---

\(^4\)A smooth birationally trivial surface with ample anticanonical sheaf.

\(^5\)A curve on \( X \) isomorphic to \( \mathbb{P}^1 \) with self-intersection -1.
Figure 8: The Gram matrix of a configuration as returned by the computer.

Figure 9: The same Gram matrix as above after it has been rearranged to evidence more structure.
finds to other areas of math is the On-Line Encyclopedia of Integer Sequences[9]. For instance, if one were unaware of the construction of those universal optima from del Pezzo surfaces mentioned earlier, one could enter the sequence “16,27,56,240” indicating the number of points $N$ of the only known nontrivial universal optima in dimensions $n = 5, 6, 7, 8$, then the encyclopedia would tell us that these numbers are part of the larger sequence “0, 1, 3, 6, 10, 16, 27, 56, 240” that are the “Number of irreducible exceptional curves of first kind on del Pezzo surface of degree $9-m$”. Even though this does not prove anything, and it may turn out to be a mere coincidence, this sort of evidence can lead a researcher to investigate what may turn out to be very important discovery.

References


