ASYMPTOTICALLY OPTIMAL SPHERICAL CODES

Jon Hamkins
hamkins@uiuc.edu

Kenneth Zeger
zeger@uiuc.edu

Coordinated Science Lab., University of Illinois, 1308 W. Main St., Urbana, IL 61801

Abstract

A technique analogous to laminated lattice construction is given for creating spherical codes. These "laminated spherical codes" outperform many other spherical codes, such as those derived from lattices, concatenated codes, and other lower bounds. Additionally, for fixed dimension $k$, the density of codepoints on the unit sphere approaches the best known sphere packing density in $k - 1$ dimensions for $k = 3 - 8$, as the minimum distance $d \to 0$. In particular, the three-dimensional laminated spherical code is asymptotically optimal, in the sense that its density approaches the well-known Fejes Tóth upper bound as $d \to 0$. Our codes are also highly structured, which simplifies decoding.

1 Introduction

A spherical code is a set of codepoints on the surface of a multi-dimensional unit radius sphere. Let $S_k$ be the surface of the unit radius $k$-dimensional sphere:

$$S_k \equiv \{(x_1, \ldots, x_k) : \sum_{i=1}^{k} x_i^2 = 1\}.$$  

Denote the surface area of $S_k$ by $A(S_k)$, and the volume by $V(S_k)$. For a $k$-dimensional spherical code $C \subseteq S_k$, the minimum distance is defined as

$$d \equiv \min_{x, y \in C, x \neq y} \|x - y\|,$$

where $\|x - y\|$ is the Euclidean distance between codepoints $x$ and $y$. $C(k, d)$ denotes a $k$-dimensional spherical code with minimum distance $d$. The angular separation between two points $x, y \in S_k$ is $\cos^{-1}(x \cdot y)$. The minimum angular separation (expressed in terms of $d$), is defined as

$$\theta \equiv 2 \sin^{-1}(d/2).$$  

The set of points on $S_k$ whose angular separation from a fixed point $y$ in $S_k$ is at most $\theta$ is called a spherical cap centered at $y$ with angular radius $\theta$, denoted $c(k, \theta)$. That is,

$$c(k, \theta) \equiv \{x \in S_k : x \cdot y \geq \cos \theta\}.$$  

Note that if the codepoints of a spherical code with minimum distance $d$ are the centers of spherical caps of angular radius $\theta/2$, then none of the caps overlap. Let $A(c(k, \theta))$ denote the surface area of $c(k, \theta)$.

The density $\Delta_c(k, d)$ of spherical code $C(k, d)$ is defined as the ratio of the surface area of $[C(k, d)]$ nonoverlapping spherical caps with angular radius $\theta/2$ to the surface area of $S_k$:

$$\Delta_c(k, d) \equiv \frac{|C(k, d)| \cdot A(c(k, \theta/2))}{A(S_k)}.$$  

Let

$$\Delta(k, d) \equiv \max_{c(k, d)} \Delta_c(k, d),$$

and let $\Delta_k \equiv \lim_{d \to 0} \Delta(k, d)$. Define $\Delta^\text{pack}_k$ to be the highest sphere packing density in $k$ dimensions.

It is currently an open problem to determine $\Delta(k, d)$ for $k \geq 3$, although a few special cases have been solved and a number of bounds have been proposed [Ran55, Yag58, Fej59, Wyn68, Cox68, DGST77, GHSW87, EZ95]. For small $d$, [Fej59] and [Cox68] provide the tightest known upper bound and [GHSW87] provides the tightest known lower bound. There is a gap between these two bounds. We construct spherical codes which provide a new lower bound that asymptotically meets the upper bound of [Fej59] for dimension 3, as shown in Table 1.

We refer to a class of codes $\{C(k, d)\}$ as asymptotically optimal if $\Delta_c(k, d)/\Delta(k, d) \to 1$ as $d \to 0$. Although $\Delta(k, d)$ is not known, it is not hard to prove that $\Delta_k \leq \Delta^\text{pack}_k$ for all $k$, which can be used to show that laminated spherical codes developed in this paper are asymptotically optimal for $k = 3$. For $4 \leq k \leq 8$, the laminated spherical codes are asymptotically optimal if and only if the the best known sphere packing

*This work was supported in part by the National Science Foundation and Engineering Research Associates.
<table>
<thead>
<tr>
<th>Reference</th>
<th>Asymptotic density achieved</th>
<th>$k$</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>$\Delta_k^L$ = $\Delta_{k-1}^L V(S_{k-1}) / 2V(S_{k-2})$</td>
<td>2</td>
</tr>
<tr>
<td>[GHS87]</td>
<td>$\Delta_{k-1}^L V(S_{k-1}) / 2V(S_{k-2})$</td>
<td>1</td>
</tr>
<tr>
<td>[Yag58]</td>
<td>$\Delta_{k-1}^L V(S_{k-1}) / A(S_k)$</td>
<td>1</td>
</tr>
<tr>
<td>[Wyn55]</td>
<td>$2^{-k} = C_k^L$</td>
<td>1</td>
</tr>
<tr>
<td>Parameters used</td>
<td>$\Delta_k^{\text{pack}}$; best packing density</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 1: Summary of asymptotic performance of various spherical codes, as $d \to 0$. The asymptotic performance may be expressed precisely in terms of the parameters of the bottom rows.

in the previous dimension is optimal. A corollary of this is that the apple-peeling lower bound [GHSW87] is not asymptotically optimal for $3 \leq k \leq 8$.

A summary of the densities of the best spherical codes and sphere packings is given in Table 1. In Section 2, our laminating technique for creating spherical codes is given and shown to be asymptotically optimal for $k = 3$. A plot compares the codes constructed here to many other codes. In Section 3 the structure of the code is shown to allow easy decoding of a received point.

2 Laminated spherical codes

By placing codepoints on concentric $(k-1)$-dimensional spheres and projecting the points onto $S_k$, a $k$-dimensional spherical code may be obtained. By nesting the concentric spheres closely, and placing codepoints of one sphere at the radial extension of the deep-holes of codepoints of the previous sphere, the method of constructing laminated lattices comes to mind (see, e.g., [CS88]), and thus we denote our codes by $C_k^L$. The procedure is illustrated with $k = 3$ in Figure 1. The method is similar to the Yaglom [Yag58] and apple-peeling [GHSW87] lower bounds in that a projection from $k-1$ dimensions to $k$ dimensions is used; the difference lies in the placement of points before the projection. The technique is practical for creating codes of any size and thus provides a lower bound on achievable minimum distance as a function of code size. For $k = 3$, this lower bound is tighter than the Yaglom and apple-peeling lower bounds, and even tight enough to prove its asymptotic optimality.

Figure 1: (a) 5 scaled 2-dimensional codes. (b) A 3-dimensional code derived from the 2-dimensional codes by projecting codepoints out of the page. Codepoints are the centers of the caps.
2.1 The laminated spherical code construction

Fix the dimension $k$ and minimum distance $d \in (0, 1]$. Given a finite increasing sequence $r_0, \ldots, r_N$ of real numbers in $[0, 1]$, we define a collection of shells, where the $i$th shell consists of those points in $\mathbb{R}^k$ whose distance to the origin is in the interval $[r_{i-1}, r_i]$. Define $g: \mathbb{Z} \to \mathbb{Z}$ by $g(i) = \min\{j: r_j \geq i\sqrt{d}\}$ and $s: \mathbb{Z} \to \mathbb{Z}$ by $s(i) = g(\max\{j: r_g(j) \leq r_i\})$. We refer to the $i$th annulus as the set of points in $\mathbb{R}^k$ whose distance to the origin is in the interval $[r_g(i), r_{g(i)+1}]-1$. That is, an annulus is a collection of consecutive shells. Note that $s(i)$ is the smallest integer for which the set of points whose distance from the origin is in the interval $[r_g(i), r_i]$ is a subset of one annulus. Between the $(i-1)th$ and $ith$ annuli is the $ith$ buffer zone. We call shells, annuli, and buffer zones which are projected onto the unit $k$-dimensional sphere $S_k$ again shells, annuli, and buffer zones, respectively. These quantities are formally defined by:

$$T_i = \left\{ (x_1, \ldots, x_k) \in S_k : \sum_{i=1}^{r_{i-1}, r_i} z_i^2 \right\}$$

$$A_i = \bigcup_{j = s(i)+1}^{s(i+1)-1} T_j$$

$$B_i = \left\{ (x_1, \ldots, x_k) \in S_k : \sum_{i=1}^{r_g(i)-1, r_g(i)} z_i^2 \right\}$$

$$W_i = \left\{ (x_1, \ldots, x_k) \in S_k : \sum_{i=1}^{r_{i-1}, r_i} z_i^2 < d^{1/4} \right\}$$

$$W_2 = \left\{ (x_1, \ldots, x_k) \in S_k : \sum_{i=1}^{r_{i-1}, r_i} z_i^2 > 1 - d^{1/4} \right\}$$

$$W = W_1 \cup W_2$$

$$T = T_i \cup T_i$$

$$B = B_i \cup B_i.$$
It remains to determine a formula for the smallest allowable radius \( r_i \) for the next concentric sphere, given \( r_0, \ldots, r_{i-1} \). The quantity \( r_i \) is chosen as small as possible and yet large enough so that the codepoints at radius \( r_i \) are at least distance \( d \) from the codepoints at radius \( r_{i-1} \) after projecting them onto the \( k \)-dimensional sphere. The solution is obtained by:

\[
\begin{align*}
  r_i &= \left( r_{i-1} \left( 1 - \frac{d^2}{2} \right) \sqrt{1 - \frac{c_k d}{r_{(i)}}} \right) + d \sqrt{1 - r_{i-1}^2} \left( 1 - \frac{d^2}{4} - \frac{2c_k r_{i-1}}{r_{(i)}} \right)^2 \left( 1 - \frac{c_k d^2}{r_{(i)}} \right)^2 .
\end{align*}
\]

The solution above is used only for shells within the same annulus. Between annuli, the radius \( r_i \) is defined such that every point on the \( i \)th shell is at least a distance \( d \) from every point on the \((i-1)\)th shell. This gives:

\[
r_i = r_{i-1} \left( 1 - \frac{d^2}{2} \right) + d \sqrt{1 - r_{i-1}^2} \left( 1 - \frac{d^2}{4} \right) .
\]

Summarizing, the \( r_i \)'s may be determined by the following algorithm:

\[
\begin{align*}
  r_s &:= r_0 := i := 0; \\
  \text{while } r_i \leq \sqrt{1 - d^2/4} \{ \\
  \quad i := i + 1; \\
  \quad r := (\text{RHS of (4)}); \\
  \quad \text{if } (r \in \mathbb{R}) \text{ and } (r > r_i) \text{ and } (r - r_s \leq \sqrt{d}) \text{ then } r_i := r; \text{ /* regular solution */ } \\
  \quad \text{else } \{ \text{/* begin new annulus */} \\
  \quad \quad r_s := r_i := (\text{RHS of (5)}); \\
  \quad \} \\
  \} \\
  N := i - 1;
\end{align*}
\]

The apple-peeling spherical code and \( C^L \) are compared in Figure 3 when \( k = 3 \) and \( d = 0.05 \).

### 2.2 Laminated spherical code density

We are now ready to compute the density of a laminated spherical code. Let \( \Delta^L(k,d) \) be the density of the \( k \)-dimensional laminated spherical code with minimum distance \( d \), and let \( \Delta^L = \lim_{d \to 0} \Delta^L(k,d) \).

**Theorem 1** The density of a laminated spherical code in \( k \) dimensions satisfies \( \Delta^L(k,d) = \frac{\Delta_{k-1}^L V(S_{k-1})}{2\sqrt{1 - c_k^2 V(S_{k-2})}} - O(d^{1/4}) \).

**Sketch of proof:** Within an annulus, layers of shells are stacked much the same as layers of lattices are stacked in the construction of laminated lattices. Thus, we expect that the density of \( C^L(k,d) \) within an annulus should be nearly the best packing density in \( k - 1 \) dimensions. After much algebra and asymptotic analysis, the density of codepoints within region \( T \), denoted \( \delta_T \), can be bounded as:

\[
\delta_T \geq \frac{\Delta_{k-1}^L V(S_{k-1})}{2\sqrt{1 - c_k^2 V(S_{k-2})}} - O(d^{1/4}).
\]

Consider now the buffer zones. Since \( r_{y(j)} - r_{y(j)-1} < d \), the area of the \( j \)th buffer zone \( B_j \) is bounded above by \( A(S_{k-1})d \). By construction, the total number of annuli in \( C^L \) is \( 2[d^{-1/2}] \), where we include annuli with both positive and negative \( k \)th coordinates. Hence, the total area of \( B = \bigcup B_j \) is:

\[
A(B) < 2 A(S_{k-1})d[d^{-1/2}] = O(d^{1/2}),
\]

and \( A(W) = O(d^{1/4}). \) Thus,

\[
\Delta^L(k,d) \geq \frac{\delta_T} {\Delta(S_k)} = \delta_T \left( 1 - O(d^{1/4}) \right) = \frac{\Delta_{k-1}^L V(S_{k-1})}{2\sqrt{1 - c_k^2 V(S_{k-2})}} - O(d^{1/4}).
\]

**Corollary 1** As \( d \to 0 \), the density of the laminated spherical code \( C^L(k,d) \) approaches the density of the best known \((k - 1)\)-dimensional sphere packing. In particular, \( C^L(3,d) \) is asymptotically optimal, and the Fejes Tóth upper bound is asymptotically tight.

Shown in Figure 4 is a plot of the proposed lower bound \( \Delta^L(3,d) \) versus \( d \). The quantity \( \Delta^L(3,d) \) is derived from actual code constructions, and a logarithmic scale is used for \( d \). All code sizes are normalized by the Fejes Tóth upper bound, i.e., \( \Delta^L(3,d)/\Delta_F(3,d) \) is plotted versus \( d \), where \( \Delta_F(3,d) \) is the upper bound on density. For small \( d \), the code performance is better than any previous codes, and the convergence to the upper bound is evident. Included in the plot are other codes obtained by the authors using a simulating annealing approach which improves upon [GHSW87]. This method produces good codes, but due to time constraints is limited in the code size that can be constructed. Another unstructured code approach is used in a program by [Slo94], which has produced many of the best known codes for small code sizes. Spherical codes can also be generated from shells of lattices (see e.g., [CS88]). Figure 4 also shows codes generated from the first 1000 shells of the face-centered cubic and \( Z_2 \) lattices, whose exact minimum distances were obtained by a computer search. Figure 4 also shows spherical codes formed from concatenations of MPSK and 2-AM codes. As can be seen, for \( d < 0.1 \) all of these codes are outperformed by the proposed laminated codes.
Figure 3: $k = 3, d = 0.05$. (a) The apple-peeling code has 4764 codepoints. Any shell of this code may be rotated without affecting the minimum distance. (b) $C^L$ has 5230 codepoints. Codepoints in adjacent shells are interlaced. Any annulus may be rotated without affecting the minimum distance, but individual shells may not be rotated. The 5 annuli are numbered.

Figure 4: Comparison of 3-dimensional spherical codes.
3 Decoding

Let $r \in S_k$. An important question in channel coding and quantization theory is how to find the nearest codepoint to $r$ in an efficient manner [GRH88, MN91]. An advantage of having a structured code in this case is that the codepoints themselves need not be stored. We consider two approaches to decoding, reflecting a tradeoff in the time and space complexity of the decoding: namely, (I) the decoder stores the sequence of radii $\{r_i\}$, or (II) the decoder stores only the minimum distance $d$.

We can show that if the spherical code $C^L(k,d)$ has size $M$, then there is an optimal decoder using $O(\sqrt{M})$ space and $O(\log M)$ time, or an optimal decoder using $O(1)$ space and $O(\sqrt{M})$ time. This may be proved as follows. Given the received vector $r = (r_1, \ldots, r_k)$, the decoder may perform a binary search of the sequence $\{r_i\}$ in $O(\log M)$ time to determine the index $i$ such that $r_i - r_{i-1} \leq \sum_{j=1}^{k-1} x_j^2 \leq r_i$. Under (I), the search may be performed directly. Under (II), the radii are generated. It is not hard to show that the length of the sequence $\{r_i\}$ is $O(\sqrt{M})$. Only codepoints arising from a constant number of $C_i$'s are candidates for being a nearest neighbor, and the nearest neighbor to $r$ from a shell $C_i$ may be determined in constant time.

References


