Title: Data for "Optimality of spherical codes via exact semidefinite programming bounds"

Authors: Henry Cohn, David de Laat, and Nando Leijenhorst

List of files, types, and sizes:

File name	Type	Size
`verify.jl`	plain text	16 KB
`codes.txt`	plain text	252 MB
`energy.txt`	plain text	203 MB
`ClusteredLowRankSolver.zip`	ZIP file	113 KB
`3-180.txt`	plain text	2.2 MB
`check3-180.jl`	plain text	5 KB
`log20-56.txt`	plain text	22 KB
`log21-50.txt`	plain text	21 KB
`log21-77.txt`	plain text	32 KB
`checklog.jl`	plain text	2 KB
`checktwopoint.jl`	plain text	4 KB
`checkcodes.jl`	plain text	5 KB

Notes:

This data set includes the numerical data and computer-assisted proofs from the paper *Optimality of spherical codes via exact semidefinite programming bounds* by Cohn, de Laat, and Leijenhorst. To check the proof of the main theorem, one can execute the Julia code `verify.jl`, which uses the data files `codes.txt` and `energy.txt`.

These scripts have been developed with Julia 1.10.2 and Nemo 0.43.1. To install this specific version of Nemo start the Julia session and type the following (including the] symbol): `]add Nemo@0.43.1`

Files:

`verify.jl`
This file contains Julia code that rigorously checks the calculations needed for the proof of the main theorem in the paper. The proof is simply a straightforward verification that the hypotheses of the three-point semidefinite programming bound hold. Comments in the file explain what each part of the code is checking.

`codes.txt` and `energy.txt`
These text files provide the data needed by `verify.jl` for the spherical code optimality and energy minimization, respectively. In `codes.txt`, each case is a list on a line by itself, where a list is enclosed in square brackets and separated by commas. The elements of the list are as follows, in the notation of Section 3 of the paper:

(1) The dimension n
(2) The number of points N.
(3) The list of all the inner products that occur between points in the code except 1.
(4) The degree d used in the semidefinite programming bound.
(5) The list of matrices F_i, each denoted by a list of lists that are its rows.
(6) The list of corresponding transformation matrices, as in Section 2.2.
(7) The list of positive semidefinite matrices for the trivariate sum of squares.
(8) The list of corresponding transformation matrices.
(9) A low-rank decomposition of the S_3-invariant sum-of-squares constraint matrices. This decomposition is a list $[[c_1, \ldots, c_k], [v_1, \ldots, v_k]]$ with positive scalars c_0 and vectors v_i (written as lists), corresponding to the matrix $\sum_i c_i v_i v_i^{\mathsf{T}}$.
(10) The list of positive semidefinite matrices for the univariate sum of squares.
(11) The list of corresponding transformation matrices.
(12) A low-rank decomposition of the univariate sum-of-squares constraint matrices.

In `energy.txt`, each line instead is a list of the following, in the notation of Section 4:

(1) The dimension n
(2) The number of points N.
(3) The list of all the inner products that occur between points in the code except 1.
(4) The degree d used in the semidefinite programming bound.
(5) The potential function f.
(6) The minimal energy E.
(7) The list of matrices F_i.
(8) The list of corresponding transformation matrices.
(9) The list of positive semidefinite matrices for the trivariate sum of squares.
(10) The list of corresponding transformation matrices.
(11) A low-rank decomposition of the S_3-invariant sum-of-squares constraint matrices.

`ClusteredLowRankSolver.zip`
This ZIP file is a snapshot of the repository `https://github.com/nanleij/ClusteredLowRankSolver.jl`, which implements the algorithms described in the paper and includes documentation. We expect that this repository will be maintained and updated, while the snapshot will not, but it serves as a long-term archive.

`3-180.txt`
This text file lists exact coordinates for the 180-point spherical code shown in Figure 1.1 in the paper. The first line is an irreducible polynomial $p(x)$ of degree 36 in the variable x, and the second line is a numerical approximation to a root α of $p(x)$ (there is a unique real root within 0.01 of this approximation). In the remainder of the file, all polynomials in x are the be interpreted modulo $p(x)$ and represent elements of the number field $\mathbb{Q}(\alpha)$. The third line expresses $\sqrt{5}$ as an element of $\mathbb{Q}(\alpha)$, and the remaining lines list the coordinates for the 180 points in the code, with each point's coordinates enclosed in square brackets.

`check3-180.jl`
This file contains Julia code that verifies the correctness of `3-180.txt`. Specifically, it checks each assertion from the caption of Figure 1.1, including the icosahedral symmetry, which is checked using the expression for $\sqrt{5}$ from `3-180.txt`.

`log20-56.txt`, `log21-50.txt`, and `log21-77.txt`
These text files support the assertion from Section 1.4 that the optimal spherical codes with 56 points in 20 dimensions, 50 points in 21 dimensions, and 77 points in 21 dimensions are suboptimal for the logarithmic energy, by providing coordinates for lower-energy configurations. Each line gives coordinates for a single point, separated by commas.

`checklog.jl`
This file contains Julia code verifying that the previous three files improve on the logarithmic energy.

`checktwopoint.jl`
This file contains extremely inefficient Julia code verifying the two-point bound calculations in Appendix A. Because the two-point bounds are not an important paper of the paper, we have not optimized it.

`checkcodes.jl`
This file contains Julia code that verifies the calculations in Appendix B. Again it is not very efficient.