New Lower Bounds on Error-Correcting Ternary, Quaternary and Quinary Codes

Antti Laaksonen and Patric R. J. Östergård

Department of Communications and Networking
Aalto University School of Electrical Engineering
P.O.Box 15400, 00076 Aalto, Finland
antti.2.laaksonen@aalto.fi, patric.ostergard@aalto.fi

Abstract. Let $A_q(n,d)$ denote the maximum size of a q-ary code with size n and minimum distance d. For most values of n and d, only lower and upper bounds on $A_q(n,d)$ are known. In this paper we present 19 new lower bounds where $q \in \{3,4,5\}$. The bounds are based on codes whose automorphisms are prescribed by transitive permutation groups. An exhaustive computer search was carried out to find the new codes.

Keywords: bounds on codes, error-correcting codes, transitive groups

1 Introduction

A q-ary code C of length n is a subset of Z_q^n where $Z_q = \{0, 1, \ldots, q-1\}$. Each element $c \in C$ is called a *codeword*, and Z_q is called the *alphabet* of C. The *size* of C is $|C|$, and the *minimum distance* of C is $\min_{a,b \in C, a \neq b} d_H(a,b)$ where d_H denotes the Hamming distance. A q-ary code with length n, size M and minimum distance d is called an $(n,M,d)_q$ code.

Let $A_q(n,d)$ denote the maximum size of an $(n,M,d)_q$ code. As it is difficult to determine exact values of $A_q(n,d)$, an important problem in coding theory is to find lower and upper bounds on the function. While binary codes have received the most attention [1,4,17], also ternary [8], quaternary [5] and quinary [7] codes have been studied.

Lower bounds on $A_q(n,d)$ can be found by discovering codes: if there is an $(n,M,d)_q$ code, then $A_q(n,d) \geq M$. Computer search techniques are often used to find such codes. However, as the search space is typically very large, assumptions about the structure of the code are usually needed to make the search efficient enough.

One way to limit the search space is to assume that there are *symmetries* in the code and only consider codes with prescribed automorphisms [12,14,19]. Such automorphisms may permute coordinates and coordinate values of codewords. In [16], new binary codes were found by focusing on codes whose groups of automorphisms are transitive permutation groups. In this paper, we extend this approach to ternary, quaternary and quinary codes.

We carry out computer searches to systematically go through transitive permutation groups and search for codes with automorphisms prescribed by those groups. For a fixed group of automorphisms, the problem of finding a large code can be transformed into a graph problem where each vertex of the graph consists of an orbit of codewords and each clique in the graph corresponds to a code with the given automorphisms.

It turns out that several lower bounds on the maximum size of ternary, quaternary and quinary error-correcting codes can be improved by creating codes whose symmetries are prescribed by transitive permutation groups. We present 17 new codes, each of which yields a new lower bound on $A_q(n, d)$ where $q \in \{3, 4, 5\}$. In addition, two more lower bounds can be derived from the new codes.

The structure of the rest of the paper is as follows: In Section 2, we discuss the method using which we construct codes with prescribed automorphisms. In Section 3, we describe the computer searches we used to find the new codes. Finally, in Section 4, we present the new lower bounds on $A_q(n, d)$.

2 Code Construction

An *automorphism* of a code is a mapping from the code to itself that may permute coordinates and coordinate values of codewords. The general idea in our work is to search for codes with prescribed groups of automorphisms, i.e., to focus on codes that have certain symmetries.

In this context, it is convenient to represent codewords as sets of integers as follows: Let $[n] = \{1, 2, \ldots, n\}$. The representation of a q-ary codeword $c_1 c_2 \cdots c_n$ is a set

$$\{c_k n + k \mid k \in [n]\},$$

so that each codeword is an n-element subset of $[nq]$. The idea in using this representation is that we can permute both coordinates and coordinate values of codewords using permutations of $[nq]$.

Our general method to construct a q-ary code of length n is as follows: Let G be a permutation group of degree nq such that the group has a *block system* where each block is of the form

$$\{k, n + k, \ldots, (q - 1)n + k\}$$

where $k \in [n]$. Such a group corresponds to a group of automorphisms of a q-ary code of length n. The *orbit* of a codeword c is

$$\{gc \mid g \in G\}.$$

We construct the code as a union of orbits of codewords. Thus, for each orbit, we either include all words in the code or none of them.

Example: Let us construct a ternary code of length 4 whose group of automorphisms is

$$G = \langle (1\ 5\ 9) \rangle.$$

Here G determines that whenever we include a codeword $c_1c_2c_3c_4$ in the code, we also include all other codewords of the form $xc_2c_3c_4$ where x is any element. For example, we may create a code

$$C = \{0000, 1000, 2000, 0111, 1111, 2111, 0222, 1222, 2222\}$$

that consists of 3 orbits whose representatives are 0000, 1111 and 2222.

To prove that $A_q(n, d) \geq M$, it suffices to find a code with size M and minimum distance d. This corresponds to finding a clique of weight M in a graph that is generated as follows: Each vertex of the graph corresponds to an orbit where the minimum distance of any two words is at least d. The weight of such a vertex is the number of words in the orbit. There is an edge between two vertices if the minimum distance between any two words in the corresponding orbits is at least d.

Thus, to find a maximum-size code based on a permutation group, we should find a maximum-weight clique in a graph. Unfortunately, this is an NP-hard problem in general graphs. However, in many cases, it may be possible to find a clique whose weight is large by using, for example, backtracking or stochastic algorithms. The benefit in focusing on codes that consist of orbits of codewords is that the size of the resulting graph is moderate.

3 Computer Search

We carried out computer searches to find codes whose automorphisms are prescribed by *transitive* permutation groups. In [16], new binary codes were found using this approach; now we focus on ternary, quaternary and quinary codes. Transitive permutation groups have been classified [9,13] up to degree 47, so it is possible to systematically go through them for the parameters considered in this work.

Let T denote the collection of transitive permutation groups up to degree 47. We performed four separate searches over all groups in T. We describe the first search in detail, and the other searches are variations of it. Consider a group $G \in T$ whose degree is d. We first generate all block systems of G with block size $q \in \{3, 4, 5\}$. Then, for each such block system, we relabel the elements of $[d]$ so that the blocks are of the form

$$\{k, n + k, \ldots, (q - 1)n + k\}$$

where $n = d/q$ and $k \in [n]$. This yields a group of automorphisms for a q-ary code of length n. Finally, we select the orbits that produce the code by conducting a clique search in the corresponding graph.

In the second search, we searched for q-ary codes of length $n+1$ such that G acts transitively on n coordinates (in the manner described above) and fixes one coordinate. In the third search, we searched for q-ary codes of length nk using k copies of G that act transitively and simultaneously on n coordinates each. Finally, in the fourth search, we searched for q-ary codes of length $nk + 1$ by combining the two previous techniques.

We used the *Cliquer* software [18] to find maximum-weight cliques in orbit graphs. We restricted ourselves to graphs that contain at most 5000 vertices, because processing larger graphs would have been too slow. Each clique search was run for at most 1000 seconds; in most cases the maximum-weight clique was found in a couple of seconds.

4 New Lower Bounds

We found several new codes that improve lower bounds on $A_q(n, d)$ where $q \in \{3, 4, 5\}$. The groups and orbits are given in the Appendix.

Table 1 summarizes the new lower bounds. As many as 17 new lower bounds follow directly from the new codes. In addition, using the facts

$$A_q(n, d) \geq A_q(n + 1, d)/q$$

and

$$A_q(n, d) \geq A_q(n + 1, d + 1)$$

we obtain two more lower bounds (on $A_3(14, 4)$ and $A_4(8, 5)$), resulting in a total of 19 new lower bounds.

One of the codes in the Appendix gives the bound $A_3(15, 5) \geq 7452$ when prescribing the given group G. By augmenting this code with 360 additional codewords, the final lower bound $A_3(15, 5) \geq 7812$ is obtained. The additional codewords are presented via another group H—which is a subgroup of G—and orbits under the action of H. No codewords can be added to the other codes.

Acknowledgments This work was supported in part by the Academy of Finland, Project #289002.

References

1. Agrell, E., Vardy, A., Zeger, K.: A table of upper bounds for binary codes. IEEE Trans. Inform. Theory 47, 3004–3006 (2001)
2. Assmus, E. F., Mattson, H. F.: New 5-designs. J. Combin. Theory 6, 122–151 (1969)
3. Assmus, E. F., Mattson, H. F.: On weights in quadratic-residue codes. Discrete Math. 3, 1–20 (1972)
4. Best, M., Brouwer, A. E., MacWilliams, F. J., Odlyzko, A. M., Sloane, N. J. A.: Bounds for binary codes of length less than 25. IEEE Trans. Inform. Theory 24, 81–93 (1978)
5. Bogdanova, G. T., Brouwer, A. E., Kapralov, S. N., Östergård, P. R. J.: Error-correcting codes over an alphabet of four elements. Des. Codes Cryptogr. 23, 333–342 (2001)
6. Boukliev, I., Kapralov, S., Maruta, T., Fukui, M.: Optimal linear codes of dimension 4 over F_5. IEEE Trans. Inform. Theory 43, 308–313 (1997)
7. Bogdanova, G. T., Östergård, P. R. J.: Bounds on codes over an alphabet of five elements. Discrete Math. 240, 13–19 (2001)

Old lower bound	New lower bound
$A_3(13,4) \geq 8559$ [10]	$A_3(13,4) \geq 13122$
$A_3(14,4) \geq 24786$ [10]	$A_3(14,4) \geq 27702$
$A_3(15,4) \geq 72171$ [20]	$A_3(15,4) \geq 83106$
$A_3(15,5) \geq 6561$ [11]	$A_3(15,5) \geq 7812$
$A_3(15,6) \geq 2187$ [11]	$A_3(15,6) \geq 3321$
$A_3(16,7) \geq 729$ [2]	$A_3(16,7) \geq 1026$
$A_3(16,8) \geq 297$ [20]	$A_3(16,8) \geq 387$
$A_4(8,4) \geq 320$ [5]	$A_4(8,4) \geq 352$
$A_4(8,5) \geq 70$ [5]	$A_4(8,5) \geq 76$
$A_4(9,4) \geq 1024$ [3]	$A_4(9,4) \geq 1152$
$A_4(9,6) \geq 64$ [3]	$A_4(9,6) \geq 76$
$A_4(10,3) \geq 17408$ [5]	$A_4(10,3) \geq 24576$
$A_4(10,4) \geq 4096$ [3]	$A_4(10,4) \geq 4192$
$A_4(11,3) \geq 65536$ [15]	$A_4(11,3) \geq 77056$
$A_5(8,4) \geq 1125$ [7]	$A_5(8,4) \geq 1225$
$A_5(8,5) \geq 160$ [7]	$A_5(8,5) \geq 165$
$A_5(9,4) \geq 3750$ [7]	$A_5(9,4) \geq 4375$
$A_5(9,5) \geq 625$ [6]	$A_5(9,5) \geq 725$
$A_5(10,4) \geq 15625$ [3]	$A_5(10,4) \geq 17500$

Table 1. The new lower bounds

8. Brouwer, A. E., Hämäläinen, H. O., Östergård, P. R. J, Sloane, N. J. A.: Bounds on mixed binary/ternary codes. IEEE Trans. Inform. Theory 44, 140–161 (1998)
9. Cannon, J. J., Holt, D. F.: The transitive permutation groups of degree 32. Exp. Math. 17, 307–314 (2008)
10. Código [pseud.], A discussion on *Foros de Free1X2.com*, cited by Andries Brouwer at `https://www.win.tue.nl/~aeb/codes/ternary-1.html`
11. Conway, J. H., Pless, V., Sloane, N. J. A.: Self-dual codes over GF(3) and GF(4) of length not exceeding 16. IEEE Trans. Inform. Theory 25, 312–322 (1979)
12. Elssel, K., Zimmermann, K.-H.: Two new nonlinear binary codes. IEEE Trans. Inform. Theory 51, 1189–1190 (2005)
13. Hulpke, A.: Constructing transitive permutation groups. J. Symbolic Comput. 39, 1–30 (2005)
14. Kaikkonen, M. K.: Codes from affine permutation groups. Des. Codes Cryptogr. 15, 183–186 (1998)
15. Kschischang, F. R., Subbarayan, P.: Some ternary and quaternary codes and associated sphere packings. IEEE Trans. Inform. Theory 38, 227–246 (1992)
16. Laaksonen, A., Östergård, P. R. J.: Constructing error-correcting binary codes using transitive permutation groups. Submitted. Preprint at arXiv:1604.06022
17. MacWilliams, F. J., Sloane, N. J. A.: The Theory of Error-Correcting Codes, North-Holland, Amsterdam (1977)
18. Niskanen, S., Östergård, P. R. J.: Cliquer User's Guide, Version 1.0, Communications Laboratory, Helsinki University of Technology, Espoo, Finland, Tech. Rep. T48 (2003)
19. Östergård, P. R. J.: Two new four-error-correcting binary codes. Des. Codes Cryptogr. 36, 327–329 (2005)

20. Plotkin, M.: Binary codes with specified minimum distance. IRE Trans. Inform. Theory 6, 445–450 (1960)

Appendix: Codes for the New Lower Bounds

Bound: $A_3(13,4) \geq 13122$
Generators of G**:**
(1 10 35 32 31 28)(2 14 36 22 19 5)(3 25 20 4 24 21)
(6 18 15 27 23 9)(7 17 37 34 29 12)(8 16 38 33 30 11),
(1 25 22 21 5 17)(2 24 32 3 23 33)(4 27 38 35 8 31)
(6 16 36 7 28 11)(9 34 18 30 14 12)(10 20 15 37 19 29)
Orbit representatives:
1112000000000, 2001001000000, 0110201000000, 0220021000000,
0202002000000, 1201000000001, 2111010000001, 2022110000001,
2200120000001, 0002000100001, 1011100000002, 2220010000002,
0212110000002, 1202011000002, 2200021100002

Bound: $A_3(15,4) \geq 83106$
Generators of G**:**
(1 41 6)(2 45 7 5 42 25)(3 4 8 9 28 44)(10 32 30 37 35 27)(11 21 31)
(12 40 17 15 22 20)(13 14 18 19 38 39)(16 26 36)(23 24 43 29 33 34),
(1 2 10 44)(3 33)(4 11 7 35)(5 19 26 37)(6 27 45 39)(8 43)(9 36 42 30)
(12 15 24 21)(13 38)(14 31 32 25)(16 17 40 29)(18)(20 34 41 22)(23 28)
Orbit representatives:
200101000000000, 111201000000000, 022202000000000, 120112000000000,
222210100000000, 012120100000000, 102200200000000, 021111010000000,
020022110000000, 220100210000000, 210021210000000, 201210020000000,
001022020000000, 222001001000000, 210201201000000, 011012011000000,
012021111000000, 211001121000000, 201222221000000, 222211012000000,
210101112000000, 211110222000000, 001102222000000, 000000110100000,
221021021100000, 010112112200000

Bound: $A_3(15,5) \geq 7812$
Generators of G**:**
(1 13 34 22)(2 6 23 39 17 21 38 9 32 36 8 24)
(3 14 27 26 33 44 42 41 18 29 12 11)(4 7 16 43)
(5 45 35 30 20 15)(19 37 31 28),
(1 45 41 25 21 5)(2 4 12 29 7 9)(3 23 28)
(6 35 16 30 26 10)(8 13 18)(11 40 36 20 31 15)
(14 22 24 17 34 27)(19 42 44 37 39 32)(33 38 43)
Orbit representatives:
212221100000000, 002122200000000, 210110001000000, 112100102000000,
121212212000000, 000002022010000, 010202102122000
Generators of H**:**
(16 4 7 10 43)(31 19 37 40 28)(2 35 38 26 29)(17 5 8 41 14)

(32 20 23 11 44)(3 36 39 42 15)(18 21 24 27 45)(33 6 9 12 30)
(34 22 25 13 1)

Orbit representatives:

000011021101212, 000022002020201, 000111111111010, 000112122210020, 000122202220010, 000201021012210, 000201122122112, 000210120202021, 000220012002020, 000220120210100, 001010010100011, 001011112001020, 001022001012200, 001110111110022, 001121201222012, 001122121212102, 001200121121111, 001201102011212, 002010011001022, 002021000011202, 002022120122210, 002121120211101, 002200101010211, 002200220120102, 010002012000221, 010012110022220, 010102212200000, 010200100220120, 010211001022200, 010211102102102, 010212110100100, 011000101100201, 011001011002220, 011101210120002, 011102101222122, 011211112021202, 011211201101120, 012001010021222, 012002100102200, 012010111020221, 012210111101101, 012210200100122, 020010022000211, 020110112200110, 020112221101020, 021012021002210, 021111221212022, 022010020011212, 022110220211021, 100001221212110, 100020212202220, 100102000212100, 100111211001212, 100112210200221, 100120202210101, 101000220211112, 101101002211102, 101110220000111, 102000001210111, 102002222010110, 102110011200222, 110011201222100, 110100212220121, 110112010222120, 111010200221102, 111101211202120, 111111012221122, 112012202220101, 112100210201122, 120112222210111, 121112221202110, 122111220201112

Bound: $A_3(15, 6) \geq 3321$

Generators of G:

(1 33 20 7 24 11 13 45 17 34 36 8 40 27 44)
(2 4 6 23 10 42 29 31 18 5 22 39 26 28 15)
(3 35 37 9 41 43 30 32 19 21 38 25 12 14 16),
(1 36 11)(2 10 42 5 22 15)(3 29 43 39 38 4)
(6 26 31)(7 45 17 40 27 20)(8 19 33 14 13 9)
(12 35 37 30 32 25)(16 21 41)(18 44 28 24 23 34)

Orbit representatives:

022102200000000, 111000121000000, 222222012000000, 110012000100000, 101120020200000

Bound: $A_3(16, 7) \geq 1026$

Generators of G:

(1 21 18 38)(2 22 17 37)(3 23 36 8)(4 24 35 7)(5 34 6 33)
(9 29 26 46)(10 30 25 45)(11 31 44 16)(12 32 43 15)
(13 42 14 41)(19 39 20 40)(27 47 28 48),
(1 35 5)(2 36 6)(3 37 33)(4 38 34)(9 43 13)(10 44 14)
(11 45 41)(12 46 42)(17 19 21)(18 20 22)(25 27 29)(26 28 30)

Orbit representatives:

2000000020000000, 0122222201000000, 2221220000100000, 2010010012212000, 0112121210101010, 2001010102101010, 0221120112201010, 1012022020102010, 2002122110201020

Bound: $A_3(16, 8) \geq 387$
Generators of G**:**
(1 20 7 8 5 38)(2 3 18 35 34 19)(4 23 40 21 22 17)(6 33 36 39 24 37)
(9 28 15 16 13 46)(10 11 26 43 42 27)(12 31 48 29 30 25)(14 41 44 47 32 45),
(1 23 8 3 34 22 36)(2 38 4 33 7 24 35)(6 20 17 39 40 19 18)
(9 31 16 11 42 30 44)(10 46 12 41 15 32 43)(13)(14 28 25 47 48 27 26)
Orbit representatives:
2200210222100000, 0201000120210000, 0000221122221100,
0202211002022110, 2020120102022110, 1111002202022110

Bound: $A_4(8, 4) \geq 352$
Generators of G**:**
(1 20)(2 19)(3 26 11 18 27 10)(4 25 12 17 28 9)
(5 24)(6 23)(7 30 15 22 31 14)(8 29 16 21 32 13),
(1 3)(2 4)(5 7)(6 8)(9 27)(10 28)(11 25)(12 26)(13 31)
(14 32)(15 29)(16 30)(17 19)(18 20)(21 23)(22 24),
(1 17 9)(2 18 10)(3 11 19)(4 12 20)(5 21 13)(6 22 14)(7 15 23)(8 16 24)
Orbit representatives:
30100000, 21320000, 22002200, 11112200, 10201010, 01021010, 21212010,
12122010, 33003010, 03322110, 30232110, 30321210, 03231210, 11220310

Bound: $A_4(9, 4) \geq 1152$
Generators of G**:**
(1 8 28 35 10 17 19 26)(2 7 29 34 11 16 20 25)
(3 33 21 6 12 24 30 15)(4 32 22 5 13 23 31 14),
(1 32 30 34 10 23 21 25)(2 33 31 35 11 24 22 26)
(3 16 28 5 12 7 19 14)(4 17 29 6 13 8 20 15),
(1 13)(2 12)(3 11)(4 10)(5 24)(6 23)(7 35)(8 34)(14 33)
(15 32)(16 26)(17 25)(19 22)(20 21)(28 31)(29 30)
Orbit representatives:
210020000, 022220000, 203220100, 202030300, 332020001, 311030101,
123230101, 130220201, 111220002, 013020102, 320220102, 031230302,
112030003, 221230003, 121020103, 303020203

Bound: $A_4(9, 6) \geq 76$
Generators of G**:**
(1 30 24 14)(2 35 25 36)(3 33 23 19)(4 13)(5 10 12 6)(7 9 11 8)
(15 32 28 21)(16 18 29 17)(20 26 34 27)(22 31),
(1 11 34)(2 7 28)(3 22 17)(4 8 30)(5 9 33)(6 32 27)(10 29 16)
(12 31 35)(13 26 21)(14 18 24)(15 23 36)(19 20 25)
Orbit representatives:
221012000, 000322200

Bound: $A_4(10, 3) \geq 24576$
Generators of G**:**
(1 37 31 7)(2 26 12 36)(3 5)(4 14)(6 22 16 32)(8 40 18 30)(9 19)
(10 38 20 28)(11 27 21 17)(13 15)(23 35)(24 34)(25 33)(29 39),
(1 6 21 36)(2 35)(3 4 13 14)(5 22)(7 10)(8 39 18 29)(9 38 19 28)
(11 16 31 26)(12 25)(15 32)(17 20)(23 34 33 24)(27 30)(37 40)
Orbit representatives:
1310000000, 3120220000

Bound: $A_4(10, 4) \geq 4192$
Generators of G**:**
(1 29 4 6 11 39 24 36)(2 30 3 5 12 40 23 35)(7 38 27 18)(8 37 28 17)
(9 14 16 21 19 34 26 31)(10 13 15 22 20 33 25 32),
(1 19)(2 20)(3 7 23 27 33 37)(4 8 24 28 34 38)(5 36 25 6 35 26)
(9 11 29 31 39 21)(10 12 30 32 40 22)(13 17)(14 18)(15 16)
Orbit representatives:
0000000000, 3311000000, 0110301000, 1001301000, 2332301000,
1032121000, 1210303010

Bound: $A_4(11, 3) \geq 77056$
Generators of G**:**
(1 32 23 10)(2 31)(3 41 14 19)(4 7 37 40)(5 6 16 28)(8 25 30 36)
(9 35)(12 21 34 43)(13 42)(15 18 26 29)(17 27 39 38)(20 24),
(1 35)(2 34 24 23 13 12)(3 10 36 43 14 32)(4 9 26 31 37 42)
(5 30 27 19 38 8)(6 29 39 40 17 7)(15 20)(16 41)(18 28)(21 25)
Orbit representatives:
10120000000, 02130000000, 23330000000, 02312000000, 12000000001,
31220000001, 20130000001, 33000000002, 10200000002, 01120000002,
11100000003, 22300000003, 30002000003

Bound: $A_5(8, 4) \geq 1225$
Generators of G**:**
(1 20 40 10 27 7 17 36 16 26 3 23 33 12 32 2 19 39 9 28 8 18 35 15 25 4 24 34 11
31) (5 22 37 14 29 6 21 38 13 30),
(1 28 5 26 3 30)(2 27 6 25 4 29)(7 32)(8 31)(9 20 13 18 11 22)
(10 19 14 17 12 21)(15 24)(16 23)(33 36 37 34 35 38)(39 40)
Orbit representatives:
00000000, 41131000, 02241000, 24332000, 43411010, 34212010,
01010110, 02020220

Bound: $A_5(8, 5) \geq 165$
Generators of G**:**
(1 32 19 5 33 24 11 37 25 16 3 29 17 8 35 21 9 40 27 13)
(2 31 20 6 34 23 12 38 26 15 4 30 18 7 36 22 10 39 28 14),
(1 10 17 26 33 2 9 18 25 34)(3 12 19 28 35 4 11 20 27 36)

(5 14 21 30 37 6 13 22 29 38)(7 16 23 32 39 8 15 24 31 40)
Orbit representatives:
33330000, 40304100, 13013200, 12340210, 30134210

Bound: $A_5(9, 4) \geq 4375$
Generators of G**:**
(1 32 10 41 19 5 28 14 37 23)(2 33 11 42 20 6 29 15 38 24)
(3 43 21 25 39 7 12 34 30 16)(4 44 22 26 40 8 13 35 31 17),
(1 3 20 13)(2 4 19 12)(5 16 6 17)(7 33 26 23)(8 32 25 24)
(10 30 11 31)(14 34 42 44)(15 35 41 43)(21 29 40 37)(22 28 39 38)
Orbit representatives:
200020000, 112040000, 104010100, 232030100, 311000001, 023010101,
342010201, 121020301, 433000002, 143040102, 231010003, 411020103,
244000004, 322020004, 402040304

Bound: $A_5(9, 5) \geq 725$
Generators of G**:**
(1 44 37 26)(2 45 38 27)(3 43 39 25)(7 12 16 30)(8 10 17 28)(9 11 18 29)
(13 22 40 31)(14 23 41 32)(15 24 42 33)(19 35)(20 36)(21 34),
(1 9 13 19 27 31 37 45 4 10 18 22 28 36 40)
(2 7 15 20 25 33 38 43 6 11 16 24 29 34 42)
(3 8 14 21 26 32 39 44 5 12 17 23 30 35 41)
Orbit representatives:
444222000, 231140100, 312401100, 003121100, 123014100

Bound: $A_5(10, 4) \geq 17500$
Generators of G**:**
(1 32 33 44 5)(2 3 14 35 21)(4 45 11 42 43)(6 37 38 49 10)
(7 8 19 40 26)(9 50 16 47 48)(12 13 24 25 31)
(15 41 22 23 34)(17 18 29 30 36)(20 46 27 28 39),
(1 42)(2 11)(3 5 13 45 23 35 33 25 43 15)(4 44 34 24 14)
(6 47)(7 16)(8 10 18 50 28 40 38 30 48 20)(9 49 39 29 19)
(12 21)(17 26)(22 31)(27 36)(32 41)(37 46),
(1 32 33 44 25)(2 3 14 5 21)(4 15 11 42 43)(6 37 38 49 30)
(7 8 19 10 26)(9 20 16 47 48)(12 13 24 45 31)
(17 18 29 50 36)(22 23 34 35 41)(27 28 39 40 46)
Orbit representatives:
0000000000, 1331000000, 4342110000, 4112220000, 0442030000,
3114040000, 2333240000, 1222340000