Designs, Codes and Cryptography (2023) 91:645–675
https://doi.org/10.1007/s10623-022-01118-z

On strongly walk regular graphs, triple sum sets and their codes

Michael Kiermaier[1] · Sascha Kurz[1] · Patrick Solé[2] · Michael Stoll[1] · Alfred Wassermann[1]

Received: 26 April 2022 / Revised: 9 September 2022 / Accepted: 10 September 2022 / Published online: 15 November 2022

Abstract
Strongly walk regular graphs (SWRGs or s-SWRGs) form a natural generalization of strongly regular graphs (SRGs) where paths of length 2 are replaced by paths of length s. They can be constructed as coset graphs of the duals of projective three-weight codes whose weights satisfy a certain equation. We provide classifications of the feasible parameters of these codes in the binary and ternary case for medium size code lengths. For the binary case, the divisibility of the weights of these codes is investigated and several general results are shown. It is known that an s-SWRG has at most 4 distinct eigenvalues $k > \theta_1 > \theta_2 > \theta_3$, and that the triple $(\theta_1, \theta_2, \theta_3)$ satisfies a certain homogeneous polynomial equation of degree $s - 2$ (Van Dam, Omidi, 2013). This equation defines a plane algebraic curve; we use methods from algorithmic arithmetic geometry to show that for $s = 5$ and $s = 7$, there are only the obvious solutions, and we conjecture this to remain true for all (odd) $s \geq 9$.

Keywords Strongly walk-regular graphs · Triple sum sets · Three-weight codes

Mathematics Subject Classification Primary 05E30 · Secondary 11D41 · 94B05

1 Introduction

A strongly regular graph (SRG) is a regular graph such that the number of common neighbors of two distinct vertices depends only on whether these vertices are adjacent or not. They arise in a lot of applications, see e.g. [8]. As first observed in [14], there is a strong link to projective two-weight codes, see [11] for a survey. The notion of SRGs has been generalized to distance-regular graphs or association schemes. Noting that the number of common neighbors of two vertices equals the number of walks of length two between them, strongly walk-regular graphs

Communicated by Y. Zhou.

✉ Michael Kiermaier
michael.kiermaier@uni-bayreuth.de

1 Department of Mathematics, University of Bayreuth, Bayreuth, Germany

2 I2M, CNRS, Aix-Marseille Univ, Centrale Marseille, Marseille, France

(SWRG) were introduced in [32]. A graph is an s-SWRG if the number of walks of length s from a vertex to another vertex depends only on whether the two vertices are the same, adjacent, or not adjacent. Note that SRGs are s-SWRGs for all $s > 1$. In [32, Theorem 3.4] it is shown that the adjacency matrix of a SWRG has at most four distinct eigenvalues and the following characterization of SWRGs is given.

Lemma 1.1 (van Dam, Omidi [32, Proposition 4.1]) *Let Γ be a k-regular graph with four distinct eigenvalues $k > \theta_1 > \theta_2 > \theta_3$. Then Γ is an s-SWRG for $s \geq 3$ if and only if*

$$(\theta_2 - \theta_3)\theta_1^s + (\theta_3 - \theta_1)\theta_2^s + (\theta_1 - \theta_2)\theta_3^s = 0. \tag{1}$$

Moreover, it is known that s has to be odd. All known examples for s-SWRGs with $s > 3$ satisfy $\theta_2 = 0$ and $\theta_3 = -\theta_1$, where Eq. (1) is automatically satisfied for all odd $s \geq 3$.

Mimicking the mentioned link between SRGs and projective two-weight codes, a construction of SRWGs as coset graphs of the duals of projective three-weight codes was given recently in [27]. The eigenvalues of such graphs are integral and depend on the weights of the three-weight code, so that Eq. (1) turns into a number theory question. In [27], a construction of SWRGs from triple sum sets (TSS) is given.

Several research papers consider the feasible parameters of SRGs, see e.g. [6] for a large table together with references summarizing the state of knowledge. We remark that the smallest cases, where the existence or non-existence of a SRG is unclear, consist of 65 or 69 vertices. The corresponding parameters cannot be attained by two-weight codes since these always give graphs where the number of vertices is a power of the field size. Still, the existence of projective two-weight codes is an important source for the construction of SRGs, see e.g. [21], where several new examples have been found. An online database of known two-weight codes can be found at [12]. Due to a result of Delsarte [14, Corollary 2] the possible weights of two-weight codes are quite restricted, see Lemma 2.2.

Given the relation between the weights of a projective three-weight code and the eigenvalues of the coset graph of its dual, corresponding solutions of Eq. (1) can be easily enumerated. However, not all cases are feasible, i.e., attainable by a projective three-weight code. The aim of this paper is to study feasibility for the smallest cases. For binary codes we give results for lengths smaller than 72 and for ternary codes for lengths up to 39. Within that range only very few cases are left as open problems. This extends and corrects first enumeration results from [27]. Similar results for some special rings instead of finite fields are obtained in [26].

The remaining part of this paper is organized as follows. The necessary preliminaries are introduced in Sect. 2 followed by the enumeration results in Sect. 3. In Sect. 4 it is shown that for $s = 5$ and $s = 7$ the only rational solutions of Eq. (1) are given by the parametric solution $\theta_2 = 0$, $\theta_3 = -\theta_1$. For $s = 5$, this reduces to the determination of the set of rational points on an elliptic curve and for $s = 7$, it leads to a curve of genus 2.

The computational results from Sect. 3 for the case $q = 2$ suggest that projective three-weight codes of length n whose weights satisfy $w_1 + w_2 + w_3 = 3n/2$ possess a high divisibility of the weights and the length by powers of two. In Sect. 5 this is shown, see Lemma 5.11 and the following theorems for the details. In Appendix A we collect generator matrices for the mentioned feasible parameters from Sect. 3.

2 Preliminaries

In this article, $q \geq 2$ will always be the power of some prime p.

A linear q-ary code C of length n and dimension k is called an $[n, k]_q$ code. The number of positions which are not all-zero is called the *effective length* of C. If the length equals the effective length, C is called *full-length*. Two positions $i, j \in \{1, \ldots, n\}$ of C are called *projectively equivalent* if there is a $\lambda \in \mathbb{F}_q^*$ with $c_i = \lambda c_j$ for all codewords $c \in C$. The code C is called *projective* if it is full-length and there are no projectively equivalent positions. For a general full-length code, the *position multiplicity type* is the sequence (m_i) where m_i denotes the number of projective equivalence classes of size i.

2.1 Restrictions on the weights

If there is only a single non-zero weight, C is called a *constant weight code*. The linear constant weight codes are completely classified.

Lemma 2.1 (Bonisoli [2]) *Let C be a full-length $[n, k]_q$ code of constant weight w of dimension $k \geq 1$. Then $q^{k-1} \mid w$, and C is isomorphic to the u-fold repetition of the q-ary simplex code* $\mathrm{Sim}_q(k)$ *of dimension k with $u = w/q^{k-1}$. In particular, $n = u \cdot \frac{q^k-1}{q-1}$.*

If C has exactly two different non-zero weights, C is called a *two-weight code*.

Lemma 2.2 (Delsarte [14, Corollary 2]) *Let C be a projective two-weight code over $\mathbb{F}_q$, where $q = p^e$ for some prime p. Then there exist suitable integers u and t with $u \geq 1$, $t \geq 0$ such that the weights are given by $w_1 = up^t$ and $w_2 = (u+1)p^t$.*

If C has exactly three different non-zero weights, C is called a *three-weight code*. Furthermore, C is called Δ*-divisible* for some integer $\Delta \geq 1$ if all weights of C are divisible by Δ.

Lemma 2.3 *Let C be a linear projective $[n, k]_q$ three-weight code. Then $n \leq \frac{q^k-1}{q-1} - 2$.*

Proof Let G be a generator matrix of C. Since C is projective, G neither has a zero column, nor a pair of projectively equivalent columns. So each of the $\frac{q^k-1}{q-1}$ projective equivalence classes of non-zero vectors in $\mathbb{F}_q^k$ appears at most once as a column of G, showing that $n \leq \frac{q^k-1}{q-1}$. In the case $n = \frac{q^k-1}{q-1}$, C is the simplex code $\mathrm{Sim}_q(k)$ of dimension k over $\mathbb{F}_q$, which is a code of constant weight q^{k-1}. In the case $n = \frac{q^k-1}{q-1} - 1$, C is the simplex code $\mathrm{Sim}_q(k)$ punctured in a single position, so C has only the two weights q^{k-1} and $q^{k-1} - 1$.[1] This contradicts the assumption that C is a three-weight code. □

2.2 Weight enumerators and the MacWilliams identity

The *weight distribution* of C is the sequence of numbers (A_i) where A_i denotes the number of codewords of weight i. It can also be denoted as $(0^{A_0} 1^{A_1} 2^{A_2} \ldots)$, where entries with $A_i = 0$ may be omitted. The weight distribution is often given in polynomial form as the *(univariate) weight enumerator* $W_C(x) = \sum_i A_i x^i$ or the *homogeneous weight enumerator* $W_C(x, y) = \sum_i A_i x^{n-i} y^i$.

The weight distribution of the dual code $C^\perp$ will be denoted by (B_i). We always have $A_0 = B_0 = 1$. Furthermore, $B_0 = 0$ if and only if C is full-length and $B_0 = B_1 = 0$ if and only C is projective. For the number B_2 of general full-length codes, the following statement can be checked.

[1] In fact, in this case C is a MacDonald code.

Lemma 2.4 *Let C be a full-length q-ary linear code of length n and (m_i) the position multiplicity type of C. Then*

$$\sum_i i m_i = n \quad \textit{and}$$
$$\sum_i (q-1)\binom{i}{2} m_i = B_2$$

The weight distributions of C and $C^\perp$ are related via the *MacWilliams identities* [24]

$$\sum_{j=0}^{n-\nu} \binom{n-j}{\nu} A_j = q^{k-\nu} \cdot \sum_{j=0}^{\nu} \binom{n-j}{n-\nu} B_j \quad \text{for } 0 \le \nu \le n. \tag{2}$$

or in homogeneous polynomial form as

$$W_{C^\perp}(x, y) = \frac{1}{\#C} \cdot W_C(x + (q-1)y, x - y).$$

In fact, the B_i are uniquely determined by the A_i, as can be seen by the following variant of the MacWilliams identities. Based on the *i-th q-ary Krawtchouk polynomial*

$$K_i(x) = \sum_{\nu=0}^{i} (-1)^\nu \binom{x}{\nu} \binom{n-x}{i-\nu} (q-1)^{i-\nu},$$

we have

$$B_i = \frac{1}{\#C} \cdot \sum_{j=0}^{n} K_i(j) A_j.$$

For a binary projective $[n, k]_2$ code, the system of the four equations with $i \in \{0, 1, 2, 3\}$ can be rewritten to

$$\sum_{i>0} A_i = 2^k - 1, \tag{3}$$
$$\sum_{i\ge 0} i A_i = 2^{k-1} n, \tag{4}$$
$$\sum_{i\ge 0} i^2 A_i = 2^{k-2} \cdot n(n+1), \tag{5}$$
$$\sum_{i\ge 0} i^3 A_i = 2^{k-3} \cdot (n^2(n+3) - 6B_3). \tag{6}$$

In this special form of the left hand side, they are also called the first four *(Pless) power moments*, see [25]. Given the length n, the dimension k, and the weights w_1, w_2, w_3 of a projective three-weight code, we can compute A_{w_i} and B_3:

$$A_{w_1} = \frac{2^{k-2} \cdot \left(n^2 - 2nw_2 - 2nw_3 + 4w_2w_3 + n\right) - w_2w_3}{(w_2 - w_1)(w_3 - w_1)} \tag{7}$$
$$A_{w_2} = \frac{2^{k-2} \cdot \left(n^2 - 2nw_1 - 2nw_3 + 4w_1w_3 + n\right) - w_1w_3}{(w_2 - w_3)(w_2 - w_1)} \tag{8}$$
$$A_{w_3} = \frac{2^{k-2} \cdot \left(n^2 - 2nw_1 - 2nw_2 + 4w_1w_2 + n\right) - w_1w_2}{(w_3 - w_1)(w_3 - w_2)} \tag{9}$$

$$3B_3 = \frac{n^2(n+3)}{2} - (w_1 + w_2 + w_3)\, n(n+1) \\ +2\,(w_1w_2 + w_1w_3 + w_2w_3)\, n - 4w_1w_2w_3 + w_1w_2w_3 \cdot 2^{2-k} \tag{10}$$

All A_j except $A_0 = 1$ and A_{w_1}, A_{w_2}, A_{w_3} are equal to zero, so that the B_i with $i \geq 4$ are uniquely determined using the remaining MacWilliams identities, i.e., those for $\nu \geq 4$. Note that (10) implies that the product $w_1w_2w_3$ has to be divisible by 2^{k-2}. We remark that we will obtain stronger divisibility conditions in Sect. 5. Of course, similar explicit expressions can also be determined for field sizes $q > 2$. However, we will mostly restrict our theoretical considerations to $q = 2$ in the remaining part of the paper.

For a linear $[n_1, k_1]_q$ code C_1 and a linear $[n_2, k_2]_q$ code C_2, the *direct sum* of C_1 and C_2 is defined as

$$C_1 \oplus C_2 = \{(c_1, c_2) \mid c_1 \in C_1, c_2 \in C_2\}.$$

It is a linear $[n_1 + n_2, k_1 + k_2]_q$ code. Its weight enumerator is given by

$$W_{C_1 \oplus C_2}(x) = W_{C_1}(x) \cdot W_{C_2}(x).$$

2.3 The coset graph triple sum sets

A *coset* of a linear code C is any translate of C by a constant vector. A *coset leader* of a fixed coset is any element that minimizes the weight. The *weight* of a coset is the weight of any of its coset leaders. With this, the *coset graph* Γ_C of a linear code C is defined on the cosets of C as vertices, where two cosets are connected iff they differ by a coset of weight one. To ease notation, we speak of the eigenvalues of a graph Γ meaning the eigenvalues of the corresponding adjacency matrix. For a projective code C the eigenvalues of the coset graph $\Gamma_{C^\perp}$ of its dual code are completely determined by the occurring non-zero weights w_i of C, see [7, Theorem 1.11.1]:

Theorem 2.5 *Let C be a projective $[n, k]_q$ code with distinct weights $w_0 = 0, w_1, \ldots, w_r$. Then, the coset graph $\Gamma_{C^\perp}$ of its dual code $C^\perp$ is $n(q-1)$-regular and the eigenvalues are given by $n(q-1) - qw_i$ for $i \in \{0, \ldots, r\}$.*

Triple sum sets (TSS) have been introduced in [13] as generalization of partial difference sets. A set $\Omega \subseteq \mathbb{F}_q^k$ is called a *triple sum set* if it is closed under scalar multiplication and there are constants σ_0 and σ_1 such that each non-zero $h \in \mathbb{F}_q^k$ can be written as $h = x + y + z$ with $x, y, z \in \Omega$ exactly σ_0 times if $h \in \Omega$ and σ_1 times if $h \in \mathbb{F}_q^k \backslash \Omega$.

If $\Omega \subseteq \mathbb{F}_q^k$ and $0 \notin \Omega$, then we denote by $C(\Omega)$ the projective code of length $n = \#\Omega/(q-1)$ obtained as the kernel of the $k \times n$ matrix H whose columns are the projectively non-equivalent elements of Ω. Thus, H is the parity check matrix of the linear code $C(\Omega)$. In order to ease the notation, we abbreviate $\Gamma_{C(\Omega)}$ as $\Gamma(\Omega)$. In [27, Theorem 2] it was shown that Ω is a TSS if and only if $\Gamma(\Omega)$ is a 3-SWRG. (Actually, [27, Theorem 2] states the equivalence of $\Gamma(\Omega)$ being an s-SWRG and Ω being an s-sum set, where the element h in the definition of a TSS is a sum of s elements from Ω.)

Lemma 2.6 *Let $s \geq 2$ be an integer. The following equation holds for all θ_1, θ_2, θ_3 over any commutative ring:*

$$\begin{aligned}&(\theta_2-\theta_3)\theta_1^s+(\theta_3-\theta_1)\theta_2^s+(\theta_1-\theta_2)\theta_3^s\\&\quad=(\theta_1-\theta_2)(\theta_1-\theta_3)(\theta_2-\theta_3)\cdot\sum_{h+i+j=s-2}\theta_1^h\theta_2^i\theta_3^j\,.\end{aligned}\tag{11}$$

A coding-theoretic characterization of triple sum sets is given as follows, see [13, Theorem 2.1] or [27, Theorem 5].

Theorem 2.7 *If $\Omega\subseteq\mathbb{F}_q^k$ so that $C(\Omega)^\perp$ has length n and attains exactly three non-zero weights w_1, w_2, and w_3, then Ω is a TSS iff $w_1+w_2+w_3=\frac{3n(q-1)}{q}$.*

Proof Using Eq. (11) from Lemma 2.6 for $s=3$, (1) becomes

$$(\theta_1-\theta_2)(\theta_1-\theta_3)(\theta_2-\theta_3)(\theta_1+\theta_2+\theta_3)=0\,.$$

Theorem 2.5 shows that the eigenvalues are pairwise different and we conclude that (1) is satisfied iff $\theta_1+\theta_2+\theta_3=0$. Plugging in the formula for the eigenvalues from Theorem 2.5 gives the condition $w_1+w_2+w_3=\frac{3n(q-1)}{q}$. □

As mentioned in the introduction, all known examples for s-SWRGs satisfy $\theta_2=0$ and $\theta_3=-\theta_1$, i.e., they are s-SWRGs for all odd $s\geq 3$. So, starting from projective three-weight codes to construct SWRGs it is sufficient to study those that satisfy the weight constraint $w_1+w_2+w_3=3n(q-1)/q$. We do so in Sect. 3. We would like to point out that there are (many) binary projective three-weight codes with $w_1+w_2+w_3\neq 3n(q-1)/q$. As an example, consider the binary $[6,5]_2$ parity check code. It is a projective three-weight-code with weight distribution is $(0^1 2^{15} 4^{15} 6^1)$. The sum of its weights is 12, but $\frac{3}{2}n=9$.

2.4 The geometric point of view to linear codes

For a vector space V, let $\begin{bmatrix}V\\k\end{bmatrix}_q$ be the set of all subspaces of dimension k. If V is of finite dimension v, $\#\begin{bmatrix}V\\k\end{bmatrix}_q$ equals the Gaussian binomial coefficient $\begin{bmatrix}v\\k\end{bmatrix}_q$. Conveniently, we will identify a vector space V with the set $\begin{bmatrix}V\\1\end{bmatrix}_q$ of *points* contained in V.

A *multiset* on a base set S is a mapping $M:S\to\mathbb{Z}_{\geq 0}$, assigning a multiplicity to each element in S. For $T\subseteq S$, we set $M(T)=\sum_{s\in T}M(s)$. The *cardinality* of M is $\#M=M(S)$, which is the sum of the multiplicities of all elements. In enumerative form, a multiset may be written by statements of the form $M=\{\!\{s_1,s_2,\ldots,s_n\}\!\}$, with the obvious interpretation. If N is a multiset on a base set T and ϕ is a predicate on T, we may also use multiset-builder notations like $M=\{\!\{t\in N\mid\phi(t)\}\!\}$.

We will make use of the geometric description of linear codes as in [16]. There is a bijective correspondence of (semi-)linear equivalence classes of linear full-length $[n,k]_q$-codes C and (semi-)linear equivalence classes of spanning multisets $\mathcal{C}$ of n points in $\mathrm{PG}(V)\cong\mathrm{PG}(k-1,q)$, where V is a $\mathbb{F}_q$-vector space of dimension k. For a concrete assignment, let G be a generator matrix of C and $v_1,\ldots,v_n$ the columns of G and consider the multiset $\mathcal{C}=\{\!\{\langle v_1\rangle,\ldots,\langle v_n\rangle\}\!\}$ of points in $V=\mathbb{F}_q^k$. In this way, a codeword $c=xG$ is represented by the hyperplane $H=x^\perp$ (in fact, H represents all the $q-1$ codewords which are projectively equivalent to c). The weight of c has a natural geometric description, namely $w(c)=n-\mathcal{C}(H)$, where the hyperplane H is identified with the set of points contained in H. In other words, $w(c)$ is the number of points in $\mathcal{C}$, counted with multiplicity, which are not contained in H. The code C is projective if and only if $\mathcal{C}$ is a proper set.

The following codes have an easy geometric description:

(i) The q-ary simplex code $\mathrm{Sim}_q(k)$ of dimension $k \geq 1$ corresponds to the set of all points contained in a vector space of (algebraic) dimension k. It is a projective linear $[(q^k-1)/(q-1), k]_q$ constant code weight code with weight enumerator $W_{\mathrm{Sim}_q(k)}(x) = 1 + (q^k - 1)x^{q^{k-1}}$.

(ii) The q-ary first order Reed-Muller code $\mathrm{RM}_q(k)$ of dimension $k \geq 2$ corresponds to an affine subspace of dimension $k-1$, that is the set of points contained in $A = V \setminus W$, where V is an $\mathbb{F}_q$-vector space of dimension k and W is a subspace of codimension 1. It is a projective linear $[q^{k-1}, k]_q$ two-weight code with weight enumerator $W_{\mathrm{RM}_q(k)}(x) = 1 + (q^k - q)x^{(q-1)q^{k-2}} + (q-1)x^{q^{k-1}}$. In the geometric description, the space W is known as the *hyperplane at infinity* of A. It corresponds to the $q-1$ codewords of weight q^{k-1}.

For a fixed point P in V, we consider the standard projection $\pi_P : V \to V/P$, $x \mapsto x+P$. It is extended to the multiset $\mathcal{C}$ of points as

$$\pi_P(\mathcal{C}) = \{\!\{\pi_P(Q) \mid Q \in \mathcal{C} \text{ with } Q \neq P\}\!\}.$$

We have $\#\pi_P(\mathcal{C}) = \#\mathcal{C} - \mathcal{C}(P)$.

The projections $\pi_P(\mathcal{C})$ with $P \in \begin{bmatrix} V \\ 1 \end{bmatrix}_q$ correspond to the subcodes C' of C of codimension one. A codeword $c \in C$ with corresponding hyperplane $H < V$ is contained in C' if and only if $P \in H$.

Let C be a projective $[n, k]_q$-code with weight enumerator $W_C(x) = \sum_i A_i x^i$ and $\mathcal{C}$ a corresponding point set. If the complement $\mathcal{C}^{\complement} = \begin{bmatrix} \mathbb{F}_q^k \\ 1 \end{bmatrix}_q \setminus \mathcal{C}$ is spanning (or equivalently, $A_{q^{k-1}} = 0$), we call its corresponding $[\frac{q^k-1}{q-1} - n, k]_q$-code $C^{\complement}$ the *anticode* of C. In this way, the anticode of C is defined up to isomorphism. Its weight enumerator is $W_{C^{\complement}}(x) = 1 + \sum_{i>0} A_{q^{k-1}-i} x^i$.

3 Feasible parameters of projective three-weight codes satisfying $w_1 + w_2 + w_3 = 3n(q-1)/q$

As outlined in Sect. 2 we can construct 3-SWRGs from projective $[n, k]_q$ three-weight codes if the weights satisfy $w_1 + w_2 + w_3 = 3n(q-1)/q$. So, here we study the feasible sets of parameters n, k, w_1, w_2, w_3 such that a corresponding projective three-weight code exists. In Sect. 3.1 we consider the admissible parameters for all lengths $n < 72$ in the binary case and in Sect. 3.2 we consider the admissible parameters for all lengths $n \leq 39$ in the ternary case.

In that range we can simply loop over all weight-triples (w_1, w_2, w_3) with $1 \leq w_1 < w_2 < w_3 \leq n$ satisfying $w_1 + w_2 + w_3 = 3n(q-1)/q$. For $q = 2$, (10) implies that the product $w_1 w_2 w_3$ is divisible by 2^{k-2}, which restricts the possible choices for the dimension k. For $q = 3$ we may use the trivial bounds $1 \leq k \leq n$. Then, the MacWilliams identities uniquely determine the values of all A_is and B_is. As a first check we test if all of these values are non-negative integers. As a consequence of [33, Theorem 1], any full-length Δ-divisible $[n, k]_q$ code is the $\Delta/\gcd(\Delta, q^{k-1})$-fold repetition of some code. As projectivity forbids proper repetitions, we can restrict ourselves to the cases where $\gcd(\Delta, q^{k-1}) = \gcd(w_1, w_2, w_3, q)$ is a power of p. Examples where we can apply this criterion to exclude the existence of codes are $q = 2, n = 36, (w_1, w_2, w_3) = (12, 18, 24)$, and $k \in \{6, 7, 8\}$. The corresponding values of $(A_{w_1}, A_{w_2}, A_{w_3})$ are $(2, 56, 5)$, $(10, 104, 13)$, and $(26, 200, 29)$. For $q = 3$, this criterion can be applied to the parameters $n = 24$, $k = 4$ and weight triple $w = (14, 16, 18)$ as well as $n = 36$, $k \in \{5, 6\}$ and weight triple $w = (18, 24, 30)$. In order to find examples, we have used the software package `LinCode` [4] to enumerate matching codes or tried to reduce the problem complexity by prescribing automorphisms and applying exact or heuristic solvers for the resulting integer linear programs.

Summarizing the above, we call parameters (q, n, k, w_1, w_2, w_3) *admissible* if

(i) $1 \le w_1 < w_2 < w_3 \le n$ and $w_1 + w_2 + w_3 = 3n(q-1)/q$ and
(ii) $\gcd(w_1, w_2, w_3, q)$ is a power of p and
(iii) $w_1 w_2 w_3$ is divisible by 2^{k-2} (if $q = 2$) or $1 \le k \le n$ (if $q = 3$) and
(iv) all A_i and B_i with $i \in \{0, \ldots, n\}$ are non-negative integers and
(v) $B_1 = B_2 = 0$.

3.1 Feasible parameters for projective binary three-weight codes with $w_1 + w_2 + w_3 = 3n/2$

In Table 1 we list the admissible parameters for projective binary three-weight codes with $w_1 + w_2 + w_3 = 3n/2$. For each length $4 \le n < 72$ we list the admissible dimensions k, weight triples $w = (w_1, w_2, w_3)$, and the weight distribution in the form $(A_{w_1}, A_{w_2}, A_{w_3})$. The last column contains known results about the existence of codes with these parameters. For some cases we can state the number of isomorphism types of those codes. The 8-divisible $[n, k]_2$ codes with length at most 48 are classified in [1] and the projective codes are extracted in [18]. If not mentioned otherwise, the remaining complete classification results are obtained with the software package `LinCode` [4]. For the parameters marked with ≥ 1 we constructed at least one code by prescribing an automorphism group, see [5].

We mark the non-existence results with the keyword "**None**" in the comment column of Table 1 and give a reference to the used method. One frequently showing up is the following.

Lemma 3.1 ([15, Proposition 5], cf. [28]) *Let C be an $[n, k, d]_2$-code with all weights divisible by $\Delta = 2^a$ and let $(A_i)_{i=0,1,\ldots,n}$ be the weight distribution of C. Put*

$$\begin{aligned} \alpha &:= \min\{k - a - 1, a + 1\}, \\ \beta &:= \lfloor (k - a + 1)/2 \rfloor, \text{ and} \\ \delta &:= \min\{2\Delta i \mid A_{2\Delta i} \ne 0 \text{ and } i > 0\}. \end{aligned}$$

Then the integer

$$T := \sum_{i=0}^{\lfloor n/(2\Delta) \rfloor} A_{2\Delta i}$$

satisfies the following conditions.

(i) T is divisible by $2^{\lfloor (k-1)/(a+1) \rfloor}$.
(ii) If $T < 2^{k-a}$, then

$$T = 2^{k-a} - 2^{k-a-t}$$

for some integer t satisfying $1 \le t \le \max\{\alpha, \beta\}$. Moreover, if $t > \beta$, then C has an $[n, k - a - 2, \delta]_2$-subcode and if $t \le \beta$, it has an $[n, k - a - t, \delta]_2$-subcode.
(iii) If $T > 2^k - 2^{k-a}$, then

$$T = 2^k - 2^{k-a} + 2^{k-a-t}$$

Table 1 Admissible and realizable parameters of binary projective three-weight codes

n	k	(w_1, w_2, w_3)	$(A_{w_1}, A_{w_2}, A_{w_3})$	isom. types
4	3	(1, 2, 3)	(1, 3, 3)	1
8	4	(2, 4, 6)	(1, 11, 3)	1
8	5	(2, 4, 6)	(5, 19, 7)	1
8	6	(2, 4, 6)	(13, 35, 15)	1
12	5	(4, 6, 8)	(6, 16, 9)	4
12	6	(4, 6, 8)	(18, 24, 21)	2
16	5	(6, 8, 10)	(6, 15, 10)	5
16	6	(6, 8, 10)	(22, 15, 26)	1
16	7	(6, 8, 10)	(54, 15, 58)	**None** Lem. 3.1
16	5	(4, 8, 12)	(1, 27, 3)	1
16	6	(4, 8, 12)	(5, 51, 7)	1
16	7	(4, 8, 12)	(13, 99, 15)	2
20	5	(8, 10, 12)	(5, 16, 10)	3
20	6	(8, 10, 12)	(25, 8, 30)	**None** Lem. 3.1
24	5	(10, 12, 14)	(3, 19, 9)	1
24	6	(10, 12, 14)	(27, 3, 33)	**None** Lem. 3.1
24	6	(8, 12, 16)	(6, 48, 9)	8
24	7	(8, 12, 16)	(18, 88, 21)	52
24	8	(8, 12, 16)	(42, 168, 45)	66
24	9	(8, 12, 16)	(90, 328, 93)	13
24	10	(8, 12, 16)	(186, 648, 189)	2
24	11	(8, 12, 16)	(378, 1288, 381)	1
32	6	(12, 16, 20)	(6, 47, 10)	≥ 1
32	7	(12, 16, 20)	(22, 79, 26)	≥ 1
32	8	(12, 16, 20)	(54, 143, 58)	≥ 1
32	9	(12, 16, 20)	(118, 271, 122)	≥ 1
32	10	(12, 16, 20)	(246, 527, 250)	≥ 1
32	6	(8, 16, 24)	(1, 59, 3)	1
32	7	(8, 16, 24)	(5, 115, 7)	1
32	8	(8, 16, 24)	(13, 227, 15)	2
32	9	(8, 16, 24)	(29, 451, 31)	1
32	10	(8, 16, 24)	(61, 899, 63)	**None** Lem. 3.1
40	6	(18, 20, 22)	(25, 3, 35)	**None** Lem. 3.1
40	6	(16, 20, 24)	(5, 48, 10)	≥ 1
40	7	(16, 20, 24)	(25, 72, 30)	≥ 1
40	8	(16, 20, 24)	(65, 120, 70)	≥ 1
40	9	(16, 20, 24)	(145, 216, 150)	≥ 1
40	10	(16, 20, 24)	(305, 408, 310)	**Open**
48	6	(22, 24, 26)	(18, 15, 30)	1
48	6	(20, 24, 28)	(3, 51, 9)	1
48	7	(20, 24, 28)	(27, 67, 33)	$\geq 209\,586$
48	8	(20, 24, 28)	(75, 99, 81)	≥ 86

Table 1 continued

n	k	(w_1, w_2, w_3)	$(A_{w_1}, A_{w_2}, A_{w_3})$	isom. types
48	9	(20, 24, 28)	(171, 163, 177)	**Open**
48	7	(16, 24, 32)	(6, 112, 9)	8
48	8	(16, 24, 32)	(18, 216, 21)	66
48	9	(16, 24, 32)	(42, 424, 45)	≥ 7
48	10	(16, 24, 32)	(90, 840, 93)	≥ 2
48	11	(16, 24, 32)	(186, 1672, 189)	≥ 2
48	12	(16, 24, 32)	(378, 3336, 381)	≥ 1
52	6	(24, 26, 28)	(13, 24, 26)	1
56	6	(26, 28, 30)	(7, 35, 21)	1
56	7	(24, 28, 32)	(28, 64, 35)	≥ 1
56	8	(24, 28, 32)	(84, 80, 91)	≥ 1
56	9	(24, 28, 32)	(196, 112, 203)	≥ 1
56	10	(24, 28, 32)	(420, 176, 427)	**Open**
58	8	(24, 31, 32)	(76, 128, 51)	**None** Cor. 5.7
64	7	(28, 32, 36)	(28, 63, 36)	≥ 1
64	8	(28, 32, 36)	(92, 63, 100)	≥ 1
64	9	(28, 32, 36)	(220, 63, 228)	≥ 1
64	10	(28, 32, 36)	(476, 63, 484)	**Open**
64	11	(28, 32, 36)	(988, 63, 996)	**None** codetables
64	7	(24, 32, 40)	(6, 111, 10)	≥ 1
64	8	(24, 32, 40)	(22, 207, 26)	≥ 1
64	9	(24, 32, 40)	(54, 399, 58)	≥ 1
64	10	(24, 32, 40)	(118, 783, 122)	≥ 1
64	11	(24, 32, 40)	(246, 1551, 250)	42
64	12	(24, 32, 40)	(502, 3087, 506)	1
64	13	(24, 32, 40)	(1014, 6159, 1018)	**None** [19]
64	7	(16, 32, 48)	(1, 123, 3)	≥ 1
64	8	(16, 32, 48)	(5, 243, 7)	≥ 1
64	9	(16, 32, 48)	(13, 483, 15)	≥ 1
64	10	(16, 32, 48)	(29, 963, 31)	≥ 1
64	11	(16, 32, 48)	(61, 1923, 63)	1 [22]
64	12	(16, 32, 48)	(125, 3843, 127)	**None** Theorem 3.2
64	13	(16, 32, 48)	(253, 7683, 255)	**None** Theorem 3.2
64	14	(16, 32, 48)	(509, 15363, 511)	**None** Theorem 3.2
64	15	(16, 32, 48)	(1021, 30723, 1023)	**None** Theorem 3.2
68	9	(30, 32, 40)	(64, 299, 148)	**None** Theorem 3.2

for some integer t satisfying $0 \le t \le \max\{\alpha, \beta\}$. Moreover, if $a = 1$, then C has an $[n, k-t, \delta]_2$ subcode. If $a > 1$, then C has an $[n, k-1, \delta]_2$ subcode unless $t = a+1 \le k-a-1$, in which case it has an $[n, k-2, \delta]_2$ subcode.

A special and well-known subcase of Lemma 3.1 is that the number of even weight codewords in a $[n, k]_2$ code is either 2^{k-1} or 2^k, see Lemma 5.1. As an example, for $n = 32$, $k = 10$, and weight triple $w = (8, 16, 24)$ we obtain $(A_{w_1}, A_{w_2}, A_{w_3}) = (61, 899, 63)$. Applying Lemma 3.1 gives $\Delta = 8, a = 3, \alpha = 4, \beta = 4, \delta = 16$, and $T = 900$. As required by Part (i), T is divisible by 4. However, Part (iii) gives $t = 5$, which contradicts $0 \le t \le \max\{\alpha, \beta\}$, so that a code cannot exist.

Bounds for the largest possible minimum distance for given length and dimension are well studied in the literature, see e.g. the online tables [17]. For length $n = 64$ and dimension $k = 11$ the largest possible minimum distance is known to be either 26 or 27, which rules out the existence of a projective code with weight triple $w = (28, 32, 36)$. We use the comment "codetables" in this case. For $n = 64$ and $w = (24, 32, 40)$ we use a classification result from [19], i.e., every 13-dimensional 8-divisible binary linear code with non-zero weights in $\{24, 32, 40, 56, 64\}$ has to contain a codeword of weight 64. Anticipating the results from Sect. 5 we also apply Corollary 5.7, which shows that the length n has to be divisible by 4. The case $n = 58$ is excluded by that criterion. For length $n = 64$ and weight triple $w = (16, 32, 48)$, the dimension can be at most 11 by Theorem 3.2. Just four cases remain undecided. They occur for length $n \in \{40, 48, 56, 64\}$ and are marked by "**Open**". For each feasible case we give a suitable generator matrix as an example in Appendix A.

Based on [23, Thm. 4] (for the projective case an alternative proof is found in [20, Sec. 4]), we derive the following classification result on three-weight codes.

Theorem 3.2 *Let $\Delta = 2^a$ with $a \ge 3$ an integer and let C be a full-length $[n, k]_2$ three-weight code with the non-zero weights Δ, 2Δ and 3Δ and length $3\Delta \le n \le 4\Delta$. Then $k \le 2a+3$. In the case of equality, we have that $n \in \{4\Delta - 1, 4\Delta\}$, C is projective and falls into one of the following two cases.*

(i) For $n = 4\Delta - 1$, C is isomorphic to the direct sum of the binary simplex code of dimension $a+1$ and the binary first order Reed-Muller code of dimension $a+2$. The weight enumerator of C is

$$W_C(x) = 1 + (6\Delta - 3)x^{\Delta} + (8\Delta^2 - 8\Delta + 3)x^{2\Delta} + (2\Delta - 1)x^{3\Delta}.$$

(ii) For $n = 4\Delta$, C is isomorphic to the code with the generator matrix

$$\begin{pmatrix} \boxed{\begin{matrix} R_{a+2} \\ 1 \ \cdots \ 1 \end{matrix}} & \\ & \boxed{\begin{matrix} R_{a+2} \\ 1 \ \cdots \ 1 \end{matrix}} \end{pmatrix} \in \mathbb{F}_2^{(2a+3)\times 4\Delta},$$

where

$$\begin{pmatrix} R_{a+2} \\ 1 \ \cdots \ 1 \end{pmatrix} \in \mathbb{F}_2^{(a+2)\times 2\Delta}$$

denotes a generator matrix of the binary first order Reed-Muller code of dimension $a+2$, such that the all-one word is the last row of the generator matrix. The weight enumerator of C is

$$W_C(x) = 1 + (4\Delta - 3)x^{\Delta} + (8\Delta^2 - 8\Delta + 3)x^{2\Delta} + (4\Delta - 1)x^{3\Delta}.$$

Proof After appending zero positions, we may consider C as a code of length 4Δ. Let $\mathbf{1}$ be the all-one word of length 4Δ. The code $\bar{C} = C + \langle \mathbf{1} \rangle$ is a Δ-divisible binary linear code of effective length 4Δ containing the all-one word $\mathbf{1}$. By [23, Thm. 4], $\dim(\bar{C}) \le 2a + 4$, and in the case of equality we may assume $\bar{C} = \mathrm{RM}_2(a+2) \oplus \mathrm{RM}_2(a+2)$, up to isomorphism. So $k = \dim(C) \le 2a + 3$, and in the case of equality, C is a codimension 1 subcode of $\bar{C}$ not containing $\mathbf{1}$.

We switch to the geometric description of linear codes. The corresponding point set of $\bar{C} = \mathrm{RM}_2(a+2) \oplus \mathrm{RM}_2(a+2)$ has the form $\bar{\mathcal{C}} = A_1 \cup A_2$ with $A_1 = V_1 \setminus W_1$ and $A_2 = V_2 \setminus W_2$, where V_1 and V_2 are vector spaces over $\mathbb{F}_2$ of dimension $a + 2$ having trivial intersection, and $W_1 < V_1$, $W_2 < V_2$ are codimension 1 subspaces.[2] The ambient vector space is $V = V_1 \oplus V_2$. The codeword $\mathbf{1} \in \bar{C}$ corresponds to a hyperplane H_0 of V not containing any point of $\bar{\mathcal{C}}$. By the dimension formula, $\dim(H_0 \cap V_1) \ge a + 1$, which forces $H_0 \cap V_1 = W_1$. In the same way, $H_0 \cap V_2 = W_2$ and therefore, $W_1 + W_2 < H_0$. Since $W_1 + W_2$ has codimension 2 in V, there are only $\begin{bmatrix}2\\1\end{bmatrix}_2 = 3$ hyperplanes of V containing $W_1 + W_2$. Two of these are $V_1 + W_2$ and $W_1 + V_2$ which do contain points of $\bar{\mathcal{C}}$, so H_0 is the third one.

As C is a subcode of $\bar{C}$ of codimension 1, a corresponding point set $\mathcal{C}$ of C is given by the multiset image $\pi_P(\bar{\mathcal{C}})$ of the projection $\pi_P : V \to V/P$, $x \mapsto x + P$ with respect to a suitable point $P \in \begin{bmatrix}V\\1\end{bmatrix}_q$. Since $\mathbf{1} \notin C$, we have that $P \notin H_0$, so P must be contained in one of the other two hyperplanes containing $W_1 + W_2$. Without restriction, we may assume $P \in V_1 + W_2$. Together with $P \notin H_0$, this implies $P \in (V_1 + W_2) \setminus (W_1 + W_2)$.

Case 1: $P \in \bar{\mathcal{C}}$, so $P \in V_1 \setminus W_1$. We get that $\pi_P(A_1)$ is the set of all points in a subspace of algebraic dimension $a + 1$, $\pi_P(A_2)$ is again an affine subspace of dimension $a + 2$, and $\langle \pi_P(A_1) \rangle \cap \langle \pi_P(A_2) \rangle = \{\mathbf{0}\}$. Therefore, $C \cong \mathrm{Sim}_2(a+1) \oplus \mathrm{RM}_2(a+2)$. The weight enumerator is computed as $W_C(x) = W_{\mathrm{Sim}_2(a+1)}(x) \cdot W_{\mathrm{RM}_2(a+2)}(x)$.

Case 2: $P \notin \bar{\mathcal{C}}$, so $P \in (V_1 + W_2) \setminus ((W_1 + W_2) \cup V_1)$. A moment's reflection shows that all these choices for P lead to equivalent point sets $\bar{\mathcal{C}}$. As P is not collinear with two different points of $\bar{\mathcal{C}}$, the projection $\mathcal{C}$ with respect to P is a proper set and therefore, C is projective. So $\mathcal{C}$ is the disjoint union of the two affine subspaces $\pi_P(A_1)$ and $\pi_P(A_2)$ of dimension $a + 1$.

The dimension formula leads to $\dim(\pi_P(V_1) \cap \pi_P(V_2)) = 1$. There are unique points $Q_1 \in V_1 \setminus W_1$ and $Q_2 \in W_2$ such that P is the third point on the line $L = Q_1 + Q_2$ The affine space $\pi_P(A_2)$ has the hyperplane at infinity $(W_2 + P)/P$, which contains the single point $(Q_1 + P)/P = (Q_2 + P)/P = L/P$ of the affine space $\pi_P(A_1)$. So the point $\pi_P(V_1) \cap \pi_P(V_2) = L/P$ is contained in $\pi_P(A_1)$ and in the hyperplane at infinity of $\pi_P(A_2)$. This leads to the generator matrix stated in the theorem.

By construction, the code corresponding to the point set $\mathcal{C}$ is a projective $[2^{a+1}, 2a+3]_2$-code with (at most) the weights Δ, 2Δ and 3Δ. Equations (7), (8) and (9) evaluate to the stated weight enumerator of C. □

Looking at the feasible cases in Table 1, we notice that all of them satisfy $w_2 = n/2$, which corresponds to $\theta_2 = 0$, $\theta_3 = -\theta_1$ for the eigenvalues of s-SWRGs, see Eq. (1). While we conjecture that all integral solutions of Eq. (1) satisfy this extra constraint for all $s \ge 5$, see Sect. 4, the condition $\theta_1 + \theta_2 + \theta_3 = 0$, i.e., $w_1 + w_2 + w_3 = 3n(q-1)/q$, is sufficient for $s = 3$. So, it is an interesting open question, if 3-SWRGs obtained from the coset graph of the dual code of a projective three-weight code also have to satisfy this extra condition. To stimulate research into this direction we propose:

[2] Remember that $V_1 \setminus W_1$ is a lazy way for writing $\begin{bmatrix}V_1\\1\end{bmatrix}_q \setminus \begin{bmatrix}W_1\\1\end{bmatrix}_q$.

Table 2 Parameters of potential counterexamples to Conjecture 3.3

n	w_1	w_2	w_3	$y = 2^{k-2}$	A_1	A_2	A_3	B_3
112	50	54	64	128	48	336	127	322
116	54	56	64	128	256	56	199	440
120	54	62	64	64	72	120	63	1180
124	56	64	66	64	72	119	64	1296
140	64	72	74	64	71	120	64	1840
202	96	103	104	64	67	128	60	5396
212	96	110	112	256	297	640	86	1860
212	96	110	112	512	649	896	502	1090
240	110	122	128	256	288	480	255	2450

Conjecture 3.3 *Let C be a projective $[n,k]_2$ three-weight code with non-zero weights $w_1 < w_2 < w_3$ satisfying $w_1 + w_2 + w_3 = \frac{3n}{2}$. Then $w_2 = \frac{n}{2}$.*

We remark that the MacWilliams identities, using the non-negativity and integrality constraints, are not sufficient to prove Conjecture 3.3. As an example, the values $(n, w_1, w_2, w_3) \in \{(58, 24, 31, 32), (68, 30, 32, 40)\}$ go in line with these conditions for $q = 2$ but are excluded with more sophisticated methods, see the details stated above. Given the results obtained so far we can state that Conjecture 3.3 is true for all $n < 72$. The next case, where all non-negativity and integrality constraints for the B_i are satisfied, is given by $(n, w_1, w_2, w_3) = (100, 46, 48, 56)$. Here we have $k = 7$, $A_{w_1} = 32$, $A_{w_2} = 145$, $A_{w_3} = 78$, and $B_3 = 580$. However, we can apply Lemma 3.1 to conclude the non-existence of a binary linear code with these parameters. More precisely, Lemma 3.1.(iii), applied with $a = 1$ and $T = 224$, yields a contradiction since $T - 2^k + 2^{k-a} = 96$ is not a power of two. In Table 2 we list all parameters $\left(n, w_1, w_2, w_3, y = 2^{k-2}, A_1, A_2, A_3, B_3\right)$ up to $n = 256$, where all B_i are integral and non-negative and also Lemma 3.1 does not yield a contradiction, i.e., the parameters of potential counterexamples to Conjecture 3.3.

3.2 Feasible parameters for projective ternary three-weight codes with $\mathbf{w_1 + w_2 + w_3 = 2n}$

In Table 3 we list the admissible parameters for projective ternary three-weight codes with $w_1 + w_2 + w_3 = 2n$. For each length $3 \le n \le 39$ we list the admissible dimensions k and weight triples (w_1, w_2, w_3), and the weight distribution in the form $(A_{w_1}, A_{w_2}, A_{w_3})$. The last column contains known results about the existence of codes with these parameters. For some cases we can also state the number of isomorphism types of those codes. If not mentioned otherwise, the classification results are obtained with the software package `LinCode` [4]. For the parameters marked with ≥ 1 we constructed at least one code by prescribing an automorphism group, see [5].

We also list those non-existence results where more sophisticated methods are necessary. We mark the non-existence results with the keyword "**None**" in the comment column of Table 3 and give a reference to the used method.

For $n = 27$, $k = 6$, and weight triple $(9, 18, 27)$ we have used exhaustive enumeration using `LinCode` to exclude the existence of the corresponding code. It would be nice to also have a theoretical argument. For $36 \le n \le 39$ four cases remain undecided, which we

Table 3 Admissible and realizable parameters of ternary projective three-weight codes

n	k	(w_1, w_2, w_3)	$(A_{w_1}, A_{w_2}, A_{w_3})$	isomorphism types
3	3	(1, 2, 3)	(6, 12, 8)	1
6	3	(3, 4, 5)	(8, 6, 12)	1
9	3	(5, 6, 7)	(6, 8, 12)	1
9	4	(3, 6, 9)	(6, 66, 8)	1
18	4	(9, 12, 15)	(8, 60, 12)	4
18	5	(9, 12, 15)	(44, 150, 48)	213
18	6	(9, 12, 15)	(152, 420, 156)	52
27	4	(15, 18, 21)	(6, 62, 12)	2
27	5	(15, 18, 21)	(60, 116, 66)	$\geq$ 2 695 546
27	6	(15, 18, 21)	(222, 278, 228)	6
27	5	(9, 18, 27)	(6, 228, 8)	1
27	6	(9, 18, 27)	(24, 678, 26)	**None** exhaustive enumeration
36	5	(21, 24, 27)	(72, 90, 80)	$\geq$ 1
36	6	(21, 24, 27)	(288, 144, 296)	$\geq$ 1
36	7	(21, 24, 27)	(936, 306, 944)	**Open**
39	5	(21, 27, 30)	(42, 188, 12)	**Open**
39	6	(21, 27, 30)	(156, 494, 78)	**Open**
39	7	(21, 27, 30)	(498, 1412, 276)	**Open**

mark with the keyword "**Open**". For each feasible case we give a suitable generator matrix in Appendix A.

Similar to Conjecture 3.3, the numerical data suggests the conjecture $w_2 = \frac{2}{3}n$. Based on our computational data, we dare to state the following q-ary version of Conjecture 3.3.

Conjecture 3.4 *Let C be a projective $[n,k]_q$ three-weight code with non-zero weights $w_1 < w_2 < w_3$ satisfying $w_1+w_2+w_3 = 3(1-\frac{1}{q})n$. Then $w_2 = (1-\frac{1}{q})n$. Moreover, $w_1 = w_2-t$ and $w_3 = w_2+t$, where t is a power of the characteristic p of $\mathbb{F}_q$.*

For $q = 2$, Conjecture 3.4 follows from Conjecture 3.3 by Lemma 5.10. We further remark that the precondition $w_1 + w_2 + w_3 = 3(1 - \frac{1}{q})n$ cannot be dropped, as seen by the binary $[7,4]_2$ Hamming code, which is a three-weight code with weight distribution $(0^1 3^3 4^3 7^1)$.

4 Plane curves given by the sum of all monomials of given degree

In this section, we present some results on rational (or integral) solutions of the equation

$$\sum_{h+i+j=s-2} \theta_1^h \theta_2^i \theta_3^j = 0\,, \tag{12}$$

which for pairwise distinct $\theta_1, \theta_2, \theta_3$ is equivalent to (1) by Lemma 2.6. We restrict to the case that s is odd. When s is even, then there are no nontrivial real solutions, so *a fortiori* no rational solutions.

We denote by C_{s-2} the plane projective curve defined by (12), and we will rename the variables $\theta_1, \theta_2, \theta_3$ in this section as x, y, z. As already mentioned, C_1 is the line $x+y+z = 0$, and there are many rational points on this curve. In general, it is not hard to see that C_d is smooth over $\mathbb{Q}$, so the curve is in particular geometrically irreducible and has genus $g(C_d) = (d-1)(d-2)/2$.

For $d = 3$ (corresponding to $s = 5$), C_3 is a curve of genus 1 with some rational points, so it is an elliptic curve. A standard procedure (implemented, for example, in Magma [3]) produces an isomorphic curve in Weierstrass form. It turns out that C_3 is isomorphic to the curve with label $50a1$ in the Cremona database (http://www.lmfdb.org/EllipticCurve/Q/50/a/350.$a3$ in the LMFDB [30]). In Cremona's tables or under the link above, one can check that this curve has exactly three rational points. This proves the following.

Lemma 4.1

$$C_3(\mathbb{Q}) = \{(1 : -1 : 0), (-1 : 0 : 1), (0 : 1 : -1)\}.$$

The curve C_5 is a plane quintic of genus 6. Note that there is an action of the symmetric group S_3 on three letters on every curve C_d by permuting the coordinates. We can restrict this action to an action of the subgroup A_3 generated by a cyclic permutation. The quotient C_5' of C_5 by this action of A_3 is a curve of genus 2. We can compute a singular plane model of C_5' by taking the image of C_5 under the map

$$\mathbb{P}^2 \to \mathbb{P}^2, \quad (x : y : z) \mapsto (xyz : (xy + yz + zx)(x + y + z) : (x - y)(y - z)(z - y)).$$

A procedure implemented in Magma [3] then produces the hyperelliptic equation

$$H_5\colon y^2 = -3x^6 + 8x^5 - 28x^4 - 30x^3 + 40x^2 + 16x - 15$$

for C_5'. A 2-descent as described in [29] (and implemented in Magma) shows that the Mordell-Weil rank of the Jacobian J of H_5 is at most 1. Since one finds a point on J of infinite order (with Mumford representation $(x^2 - x + 2, 7x + 7)$), the rank is indeed 1. Using the Magma implementation of Chabauty's method combined with the Mordell-Weil sieve (see [10]), one quickly finds that the only rational point on this hyperelliptic curve is $(-1, 0)$. This point must be the image of the three obvious rational points on C_5. Since any other rational point would have to map to a different point on H_5, this proves the following.

Lemma 4.2

$$C_5(\mathbb{Q}) = \{(1 : -1 : 0), (-1 : 0 : 1), (0 : 1 : -1)\}.$$

The combination of Lemmas 4.1 and 4.2 with Lemma 1.1 leads to the following theorem.

Theorem 4.3 *Let Γ be a k-regular graph with four distinct eigenvalues $k > \theta_1 > \theta_2 > \theta_3$ and let $s \in \{5, 7\}$. Then Γ is an s-SWRG if and only if $\theta_2 = 0$ and $\theta_3 = -\theta_1$.*

Considering larger odd d, we can say the following. The quotient C_7'' of C_7 by the full S_3-action is an elliptic curve, which is isomorphic to the curve with label $10368w1$ in the Cremona database (http://www.lmfdb.org/EllipticCurve/Q/10368/j/110368.$j1$ in the LMFDB [31]). Unfortunately, this curve has rank 2 and therefore has infinitely many rational points. So we cannot use this approach to determine the set of rational points on C_7.

The quotient C_9'' of C_9 by the S_3-action is a smooth plane quartic curve, isomorphic to the curve with equation

$$\begin{aligned} &x^4 + 2x^3y + x^2y^2 - xy^3 - y^4 + 2x^3z - 4x^2yz - 3xy^2z \\ &\quad + 2y^3z + 4x^2z^2 - 3xyz^2 + 3y^2z^2 + 3xz^3 - 4yz^3 + z^4 = 0. \end{aligned}$$

A point search finds the two rational points $(-5 : 1 : 4)$ and $(-1 : 1 : 0)$. The first is the image of the three obvious rational points on C_9, whereas the second point does not lift to a rational point on C_9. Let J be the Jacobian of the curve. Then $\#J(\mathbb{F}_3) = 3^3$ and $\#J(\mathbb{F}_7) = 11 \cdot 31$, so $J(\mathbb{Q})$ has trivial torsion subgroup. Therefore, the point in $J(\mathbb{Q})$ given by the difference of the two rational points has infinite order. It might be possible to use the methods of [9] to determine the rank of $J(\mathbb{Q})$. If the rank turns out to be ≤ 2, then an application of Chabauty's method might show that the two known rational points are the only ones.

In any case, searching for rational points does not exhibit any other points than the obvious ones when $d \geq 3$ is odd. This leads to the following conjecture, which generalizes the results of Lemmas 4.1 and 4.2.

Conjecture 4.4 *If $d \geq 3$ is odd, then*

$$C_d(\mathbb{Q}) = \{(1 : -1 : 0), (-1 : 0 : 1), (0 : 1 : -1)\}.$$

Equivalently, all solutions $(\theta_1, \theta_2, \theta_3)$ in integers of (1) with $s \geq 5$ odd and $\theta_1 > \theta_2 > \theta_3$ satisfy $\theta_2 = 0$ and $\theta_3 = -\theta_1$.

5 Divisibility for binary linear codes with few weights

In this section we want to study the divisibility properties of the weights and the length of the binary linear codes with few weights. A first but very powerful tool are the MacWilliams identities. In Eqs. (3)–(6), the code has been assumed to be projective, i.e. $B_2 = 0$. For the more general situation, we prepare

$$\sum_{i>0} A_i = q^k - 1, \tag{13}$$

$$\sum_{i\geq 0} iA_i = q^{k-1}n, \tag{14}$$

$$\sum_{i\geq 0} i^2 A_i = q^{k-1}(B_2 + n(n+1)/2), \tag{15}$$

$$\sum_{i\geq 0} i^3 A_i = q^{k-2}(3(B_2 n - B_3) + n^2(n+3)/2), \tag{16}$$

for an $[n, k]_q$ code C with $B_1 = 0$.

Lemma 5.1 (Folklore) *Let C be an $[n, k]_2$ code and C_2 the subset of all codewords of even weight. Then C_2 is a linear subcode of C of dimension k or $k - 1$.*

Proof Consider the $\mathbb{F}_2$-linear map $f : C \to \mathbb{F}_2$, $c \mapsto \sum_{i=0}^{n} c_i$. Then $C_2 = \ker f$ is a linear subspace of C. By the rank-nullity theorem, the codimension of C_2 in C equals $\dim \ker f \in \{0, 1\}$. □

We call C_2 the *even-weight subcode* of C. A direct consequence of Lemma 5.1 is the following.

Lemma 5.2 *Let C be an $[n, k]_2$ code of dimension $k \geq 2$. Then C has a non-zero even weight.*

Lemma 5.3 *Let C be a linear binary $[n, k]_2$ three-weight code. Then $k \geq 2$. If C is projective, then $k \geq 3$.*

Proof if $k \leq 1$, then C consists of at most a single non-zero codeword, so C cannot have three different weights.

Assume that C is projective of dimension $k = 2$ and let G be a generator matrix of C. Then G neither has a zero column, nor a repeated column. Therefore, each of the $2^k - 1 = 3$ possible column vectors in $\mathbb{F}_2^2 \setminus \{\mathbf{0}\}$ appears at most once as a column of G, implying that $n \leq 3$. As C has three different non-zero weights, $n \geq 3$, so together we get $n = 3$ and each of the 3 non-zero vectors appears exactly once as a column of G. Therefore, C is isomorphic to the simplex code $\mathrm{Sim}_2(2)$, which is a constant weight code. This is a contradiction. $\square$

Remark 5.4 There are indeed (many) non-projective binary three-weight codes of dimension 2. An example for the smallest possible length $n = 3$ is given by the generator matrix $\left(\begin{smallmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \end{smallmatrix}\right)$, which spans a code with the weight distribution $(0^1 1^1 2^1 3^1)$.

Lemma 5.5 *Let C be a projective full-length $[n, k]_2$ three-weight code with non-zero weights w_1, w_2 and w_3, such that n is even and exactly one weight is odd. W.l.o.g. let w_2 be the odd weight.*

Then $w_2 = \frac{n}{2}$ and the even-weight subcode C_2 of C has effective length n and is a 2-fold replication of a projective $\left[\frac{n}{2}, k-1\right]_2$ two-weight code with non-zero weights $\frac{w_1}{2}$ and $\frac{w_3}{2}$.

Proof Let A_{w_i} be the number of codewords of weight w_i in C. Furthermore, let (B_i) be the dual weight distribution of C and (B_i') the dual weight distribution of C_2. Since C is projective, we have $B_1 = B_2 = 0$. We set $y = 2^{k-2} = \frac{1}{4}\#C$. By Lemma 5.3, $y \in \mathbb{Z}$. Since w_2 is the only odd weight, Lemma 5.1 gives $A_{w_2} = 2y$. Now Eq. (4) applied to C yields

$$w_1 A_{w_1} + w_3 A_{w_3} = 2y(n - w_2).$$

From Lemma 5.1 we conclude that C_2 is a two-weight code of dimension $k-1$ and effective length $n' \leq n$ with non-zero weights w_1 and w_3. Since C is projective, we have $n' \in \{n-1, n\}$. Noting that A_{w_1} and A_{w_3} are also the numbers of codewords of weights w_1 and w_3 in C_2, the application of Eq. (14) to the full-length code arising from C_2 after (possibly) removing the zero position yields

$$w_1 A_{w_1} + w_3 A_{w_3} = n' y. \tag{17}$$

Hence $n'y = 2y(n - w_2)$ and thus $n' = 2(n - w_2)$ is even. By the assumtion that n is even, $n' = n - 1$ is not possible. Therefore $n' = n$ and $w_2 = \frac{n}{2}$. So C_2 is full-length and hence $B_1' = 0$. Now the difference of Eq. (15) for C and C_2 with $w_2 = \frac{n}{2}$, $A_{w_2} = 2y$ and $B_2 = 0$ gives

$$\frac{n^2}{4} \cdot 2y = 2y \cdot \frac{n(n+1)}{2} - y\left(B_2' + \frac{n(n+1)}{2}\right),$$

which simplifies to $\frac{n^2}{2} = \frac{n(n+1)}{2} - B_2'$ and further to $B_2' = \frac{n}{2}$. As C is projective, the position multiplicities of the codimension 1 subcode C_2 are at most 2. Denoting the number of position pairs of multiplicity 2 by m, Lemma 2.4 yields $m = B_2' = \frac{n}{2}$. Therefore, all positions of C_2 appear with multiplicity 2 and thus, C_2 is the two-fold repetition of a projective two-weight code with non-zero weights $\frac{w_1}{2}$ and $\frac{w_3}{2}$. $\square$

Remark 5.6 As seen in the above proof, in the situation of Lemma 5.5 we have $A_{w_2} = 2y$. Moreover, we can use Eqs. (13) and (14) to compute the frequencies

$$A_{w_1} = \frac{(2y-1)w_3 - yn}{w_3 - w_1} \quad \text{and} \quad A_{w_3} = \frac{yn - (2y-1)w_1}{w_3 - w_1}$$

depending on the weights w_1 and w_3.

From now on, we add the extra constraint $w_1 + w_2 + w_3 = \frac{3n}{2}$.

Corollary 5.7 *Let C be a projective $[n, k]_2$ three-weight code with non-zero weights w_1, w_2 and w_3 satisfying $w_1 + w_2 + w_3 = \frac{3n}{2}$. Then n is a multiple of 4.*

Proof Since $\frac{3n}{2} = w_1 + w_2 + w_3$ is an integer, n has to be even, so that we assume $n \equiv 2 \pmod 4$. Then $\frac{3n}{2} = w_1 + w_2 + w_3$ is odd. By Lemma 5.2, C has an even weight, so exactly one weight of C is odd. Without restriction, let w_2 be the odd weight. Lemma 5.5 yields $w_2 = \frac{n}{2}$. From $w_1 + w_2 + w_3 = \frac{3}{2}n$ we may further assume $w_1 < w_2 < w_3$, so $w_1 = \frac{n}{2} - t$ and $w_3 = \frac{n}{2} + t$ for some positive integer t. Since w_1 and w_3 are even, t has to be odd. Moreover, Lemma 5.5 says that $\frac{w_1}{2}$ and $\frac{w_3}{2}$ are the weights of a projective binary two-weight code. By Lemma 2.2 the weight difference $\frac{w_3}{2} - \frac{w_1}{2} = t$ has to be a power of 2. Since t is odd, we conclude that $t = 1$, so

$$w_1 = \frac{n}{2} - 1, \quad w_2 = \frac{n}{2} \quad \text{and} \quad w_3 = \frac{n}{2} + 1.$$

Writing $y = 2^{k-2}$, the frequencies from Remark 5.6 evaluate to

$$A_{w_1} = y - \frac{n}{4} - \frac{1}{2}, \quad A_{w_2} = 2y \quad \text{and} \quad A_{w_3} = y + \frac{n}{4} - \frac{1}{2}. \tag{18}$$

Plugging these expressions into Eq. (5) leads to

$$n^2 y + 2y + \frac{n^2}{4} - 1 = n(n+1)y$$

and further to the quadratic equation

$$n^2 - 4ny + (8y - 4) = 0$$

with the two solutions $n \in \{2, 4y - 2\}$. Since the length of a three-weight code is at least 3, necessarily $n = 4y - 2$. Now Eq. (18) yields $A_{w_1} = 0$ – a contradiction. □

Using the abbreviation $y = 2^{k-2}$, we prepare Eqs. (7)–(10) in the special case $w_1 = \frac{n}{2} - t$, $w_2 = \frac{n}{2}$ and $w_3 = \frac{n}{2} + t$.

$$A_{w_1} = \frac{n(4y - n - 2t)}{8t^2} \tag{19}$$

$$A_{w_2} = 4y - 1 - \frac{n(4y - n)}{4t^2} \tag{20}$$

$$A_{w_3} = \frac{n(4y - n + 2t)}{8t^2} \tag{21}$$

$$3B_3 = \frac{n(n - 2t)(n + 2t)}{8y} \tag{22}$$

Lemma 5.8 *Let C be a projective $[n, 3]_2$ three-weight code with non-zero weights $w_1 < w_2 < w_3$ satisfying $w_1 + w_2 + w_3 = \frac{3n}{2}$. Then C has length $n = 4$, weight distribution $(0^1 1^1 2^3 3^3)$ and is isomorphic to the code spanned by the generator matrix*

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}.$$

Proof By Lemma 2.3, $n \leq 5$, and by Corollary 5.7, $4 \mid n$. Therefore $n = 4$. The code C is isomorphic to a systematic code, which has a generator matrix of the form $(I_3 \mid v)$, where I_3 denotes the 3×3 unit matrix and v is a vector in $\mathbb{F}_2^3$. As C is projective, $w(v) \geq 2$. Furthermore, $w(v) = 3$ is not possible, as C would have only the two weights 2 and 4. So $w(v) = 2$. We note that the three possibilities for v lead to equivalent codes, and that the resulting code has the stated weight distribution. □

We remark that geometrically, the above $[4, 3]_2$ code corresponds to the complement of a triangle in the projective plane $\mathrm{PG}(2, 2)$.

Theorem 5.9 *Let C be a projective $[n, k]_2$ three-weight code with non-zero weights $w_1 < w_2 < w_3$ satisfying $w_1 + w_2 + w_3 = \frac{3n}{2}$. Then n is a multiple of 4, and one of the following cases occurs.*

(i) $k \geq 4$, $n \geq 8$ and all weights of C are even.
(ii) The code C has the parameters $[4, 3]_2$ and is isomorphic to the code in Lemma 5.8.

Proof By Corollary 5.7, n is a multiple of 4.

In the case that n has only even weights, the largest weight is at least 6, so $n \geq 8$. Moreover, $k \geq 4$ by Lemma 5.8.

Now we assume that C has an odd weight. As $\frac{n}{2}$ is even, C has at least two odd weights by Lemma 5.5. Since $w_1 + w_2 + w_3 = n$ is even, we get that C has exactly two odd weights, say w_1 and w_3. Let C_2 be the even-weight subcode of C. The code C is projective and by Lemma 5.1, the codimension of C_2 in C is 1. Therefore, the maximum position multiplicity of C_2 is at most 2, and the effective length n' of C_2 is either $n - 1$ or n. Since w_2 is the only even weight of C, the subcode C_2 is a code of constant weight w_2 and frequency $A_{w_2} = \#C_2 - 1 = 2^{k-1} - 1$. From Lemma 2.1 we conclude that $w_2 = u \cdot 2^{k-2}$ and $n' = u \cdot (2^{k-1} - 1)$ with an integer $u \in \{1, 2\}$, where $u \leq 2$ follows from the maximum position multiplicity.

Let us first investigate the case $u = 2$. Here, $n' = 2^k - 2$ and $n \in \{2^k - 2, 2^k - 1\}$. Let G be a generator matrix of C. Since C is projective, G neither has a zero column, nor a repeated column. So each of the $2^k - 1$ non-zero vectors in $\mathbb{F}_2^k$ occurs exactly once as a column in G, possibly with the excection that a single vector might not occur at all. In the case $n = 2^k - 1$, all vectors occur as a column in G, so $C = \mathrm{Sim}_2(k)$, which is a code of constant weight 2^{k-1}. In the case $n = 2^k - 2$, C is the simplex code $\mathrm{Sim}_2(k)$ punctured in a single position, so C has the two weights 2^{k-1} and $2^{k-1} - 1$.[3] This contradicts the assumption that C is a three-weight code.

It remains to consider $u = 1$. Here, $w_2 = 2^{k-2}$ and $n' = 2^{k-1} - 1$, which is odd. Since n is even, necessarily $n' = n - 1$, so $n = 2^{k-1}$ and $w_2 = \frac{n}{2}$. Combined with $w_1 + w_2 + w_3 = \frac{3n}{2}$ we can write $w_1 = \frac{n}{2} - t$ and $w_3 = \frac{n}{2} + t$ for some positive integer t. Together with the abbreviation $y = 2^{k-2}$, Eqs. (19)–(21) can be simplified to

$$\begin{aligned} A_{w_1} &= \frac{y(y-t)}{2t^2} \\ A_{w_2} &= 4y - 1 - \frac{y^2}{t^2} \\ A_{w_3} &= \frac{y(y+t)}{2t^2}. \end{aligned}$$

[3] In fact, in this case C is a MacDonald code.

Now we use $A_{w_2} = 2^{k-1} - 1 = 2y - 1$ to conclude $y = 2t^2$ (or $y = 0$, which is impossible). As y is a power of 2, so is t. From $w_1 = t(2t-1)$ odd we get that t is is odd. Together, this forces $t = 1$, which gives $y = 2$ and therefore $k = 3$. Therefore, C is isomorphic to the code in Lemma 5.8. □

Lemma 5.10 *Let C be a projective $[n, k]_2$ three-weight code with non-zero weights $w_1 < w_2 < w_3$ satisfying $w_1 + w_2 + w_3 = \frac{3n}{2}$ and $w_2 = \frac{n}{2}$. Then $w_1 = w_2 - t$ and $w_3 = w_2 + t$ where t is a power of 2. Moreover, $2t \mid n$, and t is the largest integer Δ such that C is Δ-divisible.*

Proof Equations (19) and (21) give $A_{w_3} - A_{w_1} = \frac{n}{2t}$, so $2t \mid n$. Therefore $t \mid w_2 = \frac{n}{2}$, implying that t is the greatest common divisor of $w_1 = w_2 - t$, w_2 and $w_3 = w_2 + t$, so $\Delta = t$. Since C is projective, C cannot be the proper repetition of some code. Now as a consequence of [33, Theorem 1], the number $\Delta = t$ must be a power of 2. □

Lemma 5.11 *Let C be a projective $[n, k]_2$ three-weight code with non-zero weights $w_1 < w_2 < w_3$ satisfying $w_1 + w_2 + w_3 = \frac{3n}{2}$ and $w_2 = \frac{n}{2}$. Let $a \geq 0$ be the largest integer such that C is 2^a-divisible. Then $k \leq 8a + 9$.*

Proof As before, we will use the abbreviation $y = 2^{k-2}$ and write $w_1 = \frac{n}{2} - t$ and $w_3 = \frac{n}{2} + t$ with an integer $t \geq 1$. By Lemma 5.3, $y \in \mathbb{Z}$. By Lemma 5.10, $t = 2^a$ is the largest integer Δ such that C is Δ-divisible. There is an odd integer z and a non-negative integer x with $n = 2^x \cdot z$. Since C is projective, $n \leq 2^k - 1$. Together with $2t \mid n$, we get $a + 1 \leq x \leq k - 1$.

Plugging $t = 2^a$, $n = 2^x z$ and $y = 2^{k-2}$ into Eqs. (19)–(22) we get

$$A_{w_1} = \frac{z \cdot (2^{k-a-1} - 2^{x-a-1} z - 1)}{2^{a+2-x}}, \tag{23}$$

$$A_{w_2} = 2^{2(x-a-1)} z^2 + 2^k - 2^{x+k-2a-2} z - 1, \tag{24}$$

$$A_{w_3} = \frac{z \cdot (2^{k-a-1} - 2^{x-a-1} z + 1)}{2^{a+2-x}}, \tag{25}$$

$$3B_3 = \frac{z \cdot \left(2^{x-a-1} z - 1\right) \cdot \left(2^{x-a-1} z + 1\right)}{2^{k-x-2a-1}}. \tag{26}$$

First case: $k \geq x + 2a + 2$. Equivalently, $k - x - 2a - 1 \geq 1$, so the denominator of the right hand side of Eq. (26) is even. By $B_3 \in \mathbb{Z}$, the numerator is even, too. Since z is odd, this implies $a = x - 1$. Now $0 < w_1 = \frac{n}{2} - t = 2^{x-1} z - 2^a = 2^a (z - 1)$ yields $z > 1$. Equation (26) with $a = x - 1$ yields

$$3B_3 = \frac{z \cdot (z-1) \cdot (z+1)}{2^{k-3x+1}}.$$

From our precondition $k \geq x + 2a + 2 = 3x$ we have $k - 3x \geq 0$, so 2^{k-3x} is an integer. Since $\gcd(z - 1, z + 1) = 2$, we have that 2^{k-3x} either divides $z - 1$ or $z + 1$. Therefore, $z = s \cdot 2^{k-3x} + \alpha$ for some integer s and $\alpha \in \{-1, 1\}$. By $z > 1$, $s \geq 1$. Now Eq. (23) yields

$$0 < \frac{2^{a+2-x}}{z} \cdot A_{w_1} = 2^{k-x} - \left(s \cdot 2^{k-3x} + \alpha + 1\right) \leq 2^{k-x} - s \cdot 2^{k-3x},$$

so that $s < 2^{2x}$ and hence $s \leq 2^{2x} - 1$.

Using Eq. (24) we get

$$
\begin{aligned}
0 &< s(A_{w_2}+1)\\
&= s(z^2+2^k)+s\cdot z\cdot 2^{k-x}\\
&\le (2^{2x}-1)((s2^{k-3x}+\alpha)^2+2^k)-s(s2^{k-3x}+\alpha)2^{k-x}\\
&= (2^{2x}-1)(s^2 2^{2k-6x}+2\alpha s2^{k-3x}+1+2^k)-s(s2^{k-3x}+\alpha)2^{k-x}\\
&= s^2 2^{2k-4x}+2\alpha s2^{k-x}+2^{2x}+2^{k+2x}-s^2 2^{2k-6x}-\alpha s2^{k-3x+1}-1-2^k\\
&\quad - s^2 2^{2k-4x}-\alpha s2^{k-x}\\
&= \alpha s(2^{k-x}-2^{k-3x+1})+(2^{2x}+2^{k+2x})-s^2 2^{2k-6x}-(1+2^k)\\
&\le 2s\cdot 2^{k+2x}+2s\cdot 2^{k+2x}-s2^{2k-6x}\\
&= s(2^{k+2x+2}-s^{2k-6x}),
\end{aligned}
$$

where in the second last step $s \geq 1$ has been used. Therefore $k+2x+2 > 2k-6x$ and hence

$$
k \le 8x+1 = 8(a+1)+1 = 8a+9.
$$

Second case: $k \le x+2a+1$. Equation (19) implies

$$
0 < \frac{8t^2}{n}A_{w_1} = 4y-n-2t < 4y-n.
$$

We have $2^x \mid n$ and from $x \le k-1$ also $2^x \mid 2^k = 4y$. Therefore $2^x \mid 4y-n > 0$ and thus $4y-n \ge 2^x$.

By Eq. (20),

$$
\begin{aligned}
0 &< 4t^2 A_{w_2}\\
&= 4t^2(4y-1)-n(4y-n)\\
&\le 4t^2\cdot 4y-n\cdot 2^x\\
&= 2^{k+2a+2}-2^{2x}z\\
&\le (2^{k+2a+2}-2^{2x})
\end{aligned}
$$

and thus $k+2a+2 > 2x$ and $k \ge 2x-2a-1$. Combined with $k \le x+2a+1$, finally

$$
k = 2k-k \le 2(x+2a+1)-(2x-2a-1) = 6a+3 < 8a+9.
$$

□

By Lemmas 5.11 and 2.3, the following numbers $K(r)$ and $N(r)$ are well-defined.

Definition 5.12 Let $a \ge 1$ be an integer. We define $K(a)$ (resp. $N(a)$) as the largest dimension (resp. length) of a projective $[n,k]_2$ three-weight code C with non-zero weights $w_1 < w_2 < w_3$ satisfying $w_1+w_2+w_3 = \frac{3n}{2}$ and $w_2 = \frac{n}{2}$ such that C is not 2^a-divisible.

Theorem 5.13 *Let $a \ge 1$ be an integer. Then*

$$
2a+1 \le K(a) \le 8a+1 \quad \textit{and} \quad 2^{2a+1}-2^{a+1} \le N(a) \le 2^{K(a)}-3.
$$

Proof $K(a) \leq 8(a-1)+9 = 8a+1$ by Lemma 5.11 and $N(a) \leq 2^{K(a)} - 3$ by Lemma 2.3.

For the lower bounds, let $C = \mathrm{Sim}_2(a) \oplus \mathrm{RM}_2(a+1)$. It is a projective three-weight code of length $n = (2^a - 1) + 2^a = 2^{a+1} - 1$, dimension $k = a + (a+1) = 2a + 1$ and weights $w_1 = 2^{a-1}$, $w_2 = 2^a$ and $w_3 = 3 \cdot 2^{a-1}$.[4] Since C does not have the weight $2^{k-1} = 2^{2a}$, the anticode $C^\complement$ is defined. It is a projective three-weight code of the same dimension k, length $n^\complement = (2^k - 1) - n = 2^{2a+1} - 2^{a+1}$ and the three non-zero weights

$$\begin{aligned} w_1^\complement &= 2^{k-1} - w_3 = 2^{2a} - 3 \cdot 2^{a-1} = 2^{a-1}(2^{a+1} - 3), \\ w_2^\complement &= 2^{k-1} - w_2 = 2^{2a} - 2^a = 2^{a-1}(2^{a+1} - 2) \quad \text{and} \\ w_3^\complement &= 2^{k-1} - w_1 = 2^{2a} - 2^{a-1} = 2^{a-1}(2^{a+1} - 1), \end{aligned}$$

so $C^\complement$ is 2^{a-1}-divisible, but not 2^a-divisible. Furthermore

$$w_1^\complement + w_2^\complement + w_3^\complement = 3 \cdot 2^{2a} - 3 \cdot 2^a = \frac{3}{2} n^\complement \qquad \text{and} \qquad w_2^\complement = \frac{n^\complement}{2}.$$

Therefore, $K(a) \geq k = 2a + 1$ and $N(a) \geq n^\complement = 2^{2a+1} - 2^{a+1}$. □

For small values of a, we can determine the exact values of $N(a)$ and $K(a)$.

Theorem 5.14 *(a)* $K(1) = 3$ *and* $N(1) = 4$.
(b) $K(2) = 6$ *and* $N(2) = 56$.
(c) $K(3) = 11$ *and* $N(3) = 2024$.

Proof The case $a = 1$: The values $K(1) = 3$ and $N(1) = 4$ are a direct consequence of Theorem 5.9.[5]

The case $a = 2$: Using $K(2) \leq 17$ from Theorem 5.13 and $t = 2^a = 4$ from Lemma 5.10, we determined all feasible parameters computationally. The ones with $k \geq 7$ are listed below.

n	k	w_1	w_2	w_3
244	8	120	122	124
116	7	56	58	60
112	7	54	56	58
16	7	6	8	10

The last code has already been excluded in Sect. 3 via Lemma 3.1. Of the first three codes, the anticodes would have the parameters [11, 8, 4], [11, 7, 4] and [15, 7, 6]. The application of the Hamming bound to the punctured parameters [10, 7, 3], [10, 6, 3] and [14, 7, 5] shows that these codes do not exist.

Among all feasible parameters with $k \leq 6$, the ones with the largest possible length are $n = 56$, $k = 6$, $w_1 = 26$, $w_2 = 28$ and $w_3 = 30$. These parameters are realized by the anticode of the binary $[7, 6]_2$ parity check code.

The case $a = 3$: Similar as before, based on $K(3) \leq 25$ and $t = 8$ we were able to determine all feasible parameters computationally. There is only a single feasible parameter set with $k \geq 12$, which is $n = 4040$, $k = 12$, $w_1 = 2016$, $w_2 = 2020$ and $w_3 = 2024$. The anticode would have the parameters [55, 12, 24], which does not exist according to the online tables [17].

---

4 We have already seen the code C in Theorem 3.2 Case (i), with $a - 1$ in instead of a.

5 Theorem 5.9 refers to the code in Lemma 5.8. It is isomorphic to the code $C^\complement$ considered in the proof of Theorem 5.13 in the smallest case $a = 1$.

Among all parameters with $k \le 11$, the ones with the largest possible length are $n = 2024$, $k = 11$, $w_1 = 1008$, $w_2 = 1012$ and $w_3 = 1016$. These parameters are realized by the code $(C^{\perp})^{\mathsf{C}}$, where C is the binary $[23, 12]_2$ Golay code. The code $C^{\perp}$ has the weight enumerator $1 + 506x^8 + 1288x^{12} + 253x^{16}$. □

Theorem 5.14 indicates that in general, neither the lower nor the upper bound of Theorem 5.13 are sharp. We leave it as a research problem to improve the bounds and further investigate the asymptotic growth.

Funding Open Access funding enabled and organized by Projekt DEAL.

Data Availability The datasets generated during and/or analysed during the current study are available from the corresponding author on reasonable request.

A: Generator matrix of projective three-weights codes satisfying $w_1 + w_2 + w_3 = 3(q-1)n/q$

In this appendix we list examples of generator matrices corresponding to the feasible cases listed in Sect. 3.

- $q = 2, n = 4, k = 3, w = [1, 2, 3]$: $\begin{pmatrix} 1000 \\ 0101 \\ 0011 \end{pmatrix}$
- $q = 2, n = 8, k = 4, w = [2, 4, 6]$: $\begin{pmatrix} 01111011 \\ 01101010 \\ 10101100 \\ 10110010 \end{pmatrix}$
- $q = 2, n = 8, k = 5, w = [2, 4, 6]$: $\begin{pmatrix} 11101110 \\ 01010000 \\ 00111010 \\ 10001000 \\ 11000011 \end{pmatrix}$
- $q = 2, n = 8, k = 6, w = [2, 4, 6]$: $\begin{pmatrix} 00110110 \\ 00010001 \\ 01010011 \\ 10010110 \\ 10100110 \\ 01111101 \end{pmatrix}$
- $q = 2, n = 12, k = 5, w = [4, 6, 8]$: $\begin{pmatrix} 100100111001 \\ 010100111100 \\ 001000111101 \\ 000010110010 \\ 000001101010 \end{pmatrix}$
- $q = 2, n = 12, k = 6, w = [4, 6, 8]$: $\begin{pmatrix} 100000000111 \\ 010000110010 \\ 001000110100 \\ 000100111011 \\ 000010001011 \\ 000001100011 \end{pmatrix}$
- $q = 2, n = 16, k = 5, w = [6, 8, 10]$: $\begin{pmatrix} 1011111000110000 \\ 1111011001011001 \\ 0111111110101000 \\ 0110010100111111 \\ 1000011111101011 \end{pmatrix}$

- $q = 2, n = 16, k = 6, w = [6, 8, 10]$: $\begin{pmatrix} 0000100110010011 \\ 0010110001101000 \\ 0000011011001001 \\ 1001010000110100 \\ 0001001100100101 \\ 0101100001010010 \end{pmatrix}$
- $q = 2, n = 16, k = 5, w = [4, 8, 12]$: $\begin{pmatrix} 1100011101100100 \\ 1100110001001011 \\ 1010101001110010 \\ 1101100001101001 \\ 0111111011100111 \end{pmatrix}$
- $q = 2, n = 16, k = 6, w = [4, 8, 12]$: $\begin{pmatrix} 1101001010010101 \\ 1110001000011101 \\ 1101000101001011 \\ 1001011111100000 \\ 1100111011111011 \\ 1110011100011000 \end{pmatrix}$
- $q = 2, n = 16, k = 7, w = [4, 8, 12]$: $\begin{pmatrix} 1000000111011110 \\ 0100000101100000 \\ 0010000001110000 \\ 0001000001101000 \\ 0000100001100100 \\ 0000010111011101 \\ 0000001101011111 \end{pmatrix}$
- $q = 2, n = 20, k = 5, w = [8, 10, 12]$: $\begin{pmatrix} 11101110100010010100 \\ 00101001011100000110 \\ 10100101000111011001 \\ 11101000100110100011 \\ 11010101010001000001 \end{pmatrix}$
- $q = 2, n = 24, k = 5, w = [10, 12, 14]$: $\begin{pmatrix} 111011100010100000010111 \\ 011010110110000111101011 \\ 000011011110001101010110 \\ 010110011101101000001101 \\ 100000101111110110010001 \end{pmatrix}$
- $q = 2, n = 24, k = 6, w = [8, 12, 16]$: $\begin{pmatrix} 100000010111100100011111 \\ 010001010011011111001100 \\ 001000001011011010110111 \\ 000101011101010001111010 \\ 000011000100101000110001 \\ 000000101000011000001111 \end{pmatrix}$
- $q = 2, n = 24, k = 7, w = [8, 12, 16]$: $\begin{pmatrix} 001111111011101110110001 \\ 010111110111011101101010 \\ 010110111111111010101001 \\ 100111101110111111001001 \\ 001111011101111101110010 \\ 100101111111110101011010 \\ 000000000011111111111100 \end{pmatrix}$
- $q = 2, n = 24, k = 8, w = [8, 12, 16]$: $\begin{pmatrix} 100011011111001001111111 \\ 011100111001100010001101 \\ 001101100110110000010111 \\ 001010101101010101100101 \\ 011101110001000111010001 \\ 110111011011101100101110 \\ 010011010110010111010001 \\ 001111011111101100111001 \end{pmatrix}$
- $q = 2, n = 24, k = 9, w = [8, 12, 16]$: $\begin{pmatrix} 011110011110001111001111 \\ 111101111000010111110101 \\ 001011101100101000011011 \\ 111110100101000000101101 \\ 001100110000101011110101 \\ 010101000001110001111011 \\ 001101010110000110101011 \\ 110000101110101110100001 \\ 001111111100001110110111 \end{pmatrix}$
- $q = 2, n = 24, k = 10, w = [8, 12, 16]$: $\begin{pmatrix} 100000000001101010001011 \\ 010000000101001001000111 \\ 001000000001111000010101 \\ 000100000000011100111100 \\ 000010000101001100101010 \\ 000001000001001001111001 \\ 000000100001001010110110 \\ 000000010101000100110101 \\ 000000001101000111001001 \\ 000000000100001101111011 \end{pmatrix}$

- $q = 2, n = 24, k = 11, w = [8, 12, 16]$: $$\begin{pmatrix} 100000000000011011011010 \\ 010000000000010111000111 \\ 001000000000110010011101 \\ 000100000000110001110011 \\ 000010000010011001010101 \\ 000001000010011110001001 \\ 000000100000011000101111 \\ 000000010010001011100011 \\ 000000001010001010111100 \\ 000000000100001101110110 \\ 000000000001000110111011 \end{pmatrix}$$
- $q = 2, n = 32, k = 6, w = [12, 16, 20]$: $$\begin{pmatrix} 10010000110101101100111100111000 \\ 10011001111100100111000010100011 \\ 11110111000111000110101101110110 \\ 00001111111111110001111000000000 \\ 10101010000101111011001110100100 \\ 00000100100100101101111011101101 \end{pmatrix}$$
- $q = 2, n = 32, k = 7, w = [12, 16, 20]$: $$\begin{pmatrix} 01111000000111111000011111111110 \\ 11100110110010000111111000110000 \\ 10011111001000101101101100100100 \\ 10110110011001001011011100101000 \\ 01111100000000001111111111011111 \\ 11001111100100001110110100100010 \\ 01111110000000000000000001111110 \end{pmatrix}$$
- $q = 2, n = 32, k = 8, w = [12, 16, 20]$: $$\begin{pmatrix} 11101100110111010010100111110011 \\ 10011000100011110110110010001101 \\ 11011101101110100001011110100111 \\ 00110101000110101101110100011001 \\ 10111111001101000110101101001111 \\ 01101010001100011111101000110001 \\ 01111110011011001001011011011011 \\ 11000100011001111011010001100101 \end{pmatrix}$$
- $q = 2, n = 32, k = 9, w = [12, 16, 20]$: $$\begin{pmatrix} 10000000001011011001111000001010 \\ 01000000010000011101100001101110 \\ 00100000000010010101011011001101 \\ 00010000010010101011101111110110 \\ 00001000000001011011001011100011 \\ 00000100011111001101000000111000 \\ 00000010011000110000001110110011 \\ 00000001011011000010001100011110 \\ 00000000100110011110101000101001 \end{pmatrix}$$
- $q = 2, n = 32, k = 10, w = [12, 16, 20]$: $$\begin{pmatrix} 11111000001111110000001000000000 \\ 00000111111111110000000100000000 \\ 00011000110000111111100010000000 \\ 01101000010011010011110001000000 \\ 10110011011101000100100000100000 \\ 01100101110101001000110000010000 \\ 11010010110111000001010000001000 \\ 11100011011010100011000000000100 \\ 10101010110011101000100000000010 \\ 10011001000011101011010000000001 \end{pmatrix}$$
- $q = 2, n = 32, k = 6, w = [8, 16, 24]$: $$\begin{pmatrix} 01111011100101011111111101101111 \\ 01001000011110111001110001101001 \\ 01010011011000001100111010110011 \\ 10011110100001010010011010110110 \\ 01000110101011010100110101011010 \\ 10011000101110001110100111100010 \end{pmatrix}$$
- $q = 2, n = 32, k = 7, w = [8, 16, 24]$: $$\begin{pmatrix} 11010100001101001100010111100101 \\ 01010011101101010011000101010110 \\ 11101110100111100010000010110100 \\ 11100110010010111100111001100000 \\ 11111101000000100111110000101100 \\ 01000011101110010000011101110110 \\ 10001000111111010101110011010000 \end{pmatrix}$$
- $q = 2, n = 32, k = 8, w = [8, 16, 24]$: $$\begin{pmatrix} 10000001011111101000000101111110 \\ 01000001010000010100000101000001 \\ 00100001001000010010000100100001 \\ 00010001000100011110111011101110 \\ 00001001000010011111011011110110 \\ 00000101000001011111101011111010 \\ 00000011000000111111110011111100 \\ 00000000111111111111111100000000 \end{pmatrix}$$
- $q = 2, n = 32, k = 9, w = [8, 16, 24]$: $$\begin{pmatrix} 10101110000101001100011100001111 \\ 01010101001101110001010100101011 \\ 10100010100110111000011100001111 \\ 00110000101001101110100101101101 \\ 11001111111111111110110110100101 \\ 01101111010110010000101010101010 \\ 10010000000000000001100010001110 \\ 01100000000000000001000110011100 \\ 01011000010100110111010100101011 \end{pmatrix}$$

- $q = 2, n = 40, k = 6, w = [16, 20, 24]$:
$$\begin{pmatrix} 0100111000101111011111001001011000001001 1 \\ 1010010100011111101111001001011000001001 \\ 1101001010000111110111110110010010000010 1 \\ 0110100101001011110011110110010010000011 \\ 0011010010101101111001111011000001001001 \\ 1001100001011110111110011010110000010010 1 \end{pmatrix}$$
- $q = 2, n = 40, k = 7, w = [16, 20, 24]$:
$$\begin{pmatrix} 1000011111010101010111110100111110011010 10 \\ 0000100101101011101110100000011011011101 \\ 0100000011110101110110010001001100101111 \\ 0001001001011101111110100000011010011101 1 \\ 1000011110101011011110101101111001110100 \\ 0010010000111011111010001001100011010111 \\ 0111100000000111011110000000000011111111 \end{pmatrix}$$
- $q = 2, n = 40, k = 8, w = [16, 20, 24]$:
$$\begin{pmatrix} 1111100000011111011111000000111111010110 \\ 0110011000101000110100111101011011000011 \\ 1100010001110000101001111011110101000011 \\ 0001100110100010100101101111010111000011 \\ 0000011111000000111111000000100000101101 0 \\ 1000100011100001110010110111101011000011 \\ 0011001100100100101010011111101101000011 \\ 1111100000011111000000000000100000011110 01 \end{pmatrix}$$
- $q = 2, n = 40, k = 9, w = [16, 20, 24]$:
$$\begin{pmatrix} 0100010100011110100101111111111011001101 \\ 1101110000010010111011001100011011000101 \\ 0001001100100111100011111100111011111110 \\ 0000011101111110111010000110010100011011 0 \\ 1101001011110010001000000101000000111000 01 \\ 1000101000100000010011001000000100101111 11 \\ 1111101001011011001010010010111110101 01 \\ 0010100010101001000010011111000010110100 \\ 0110010000100111001101001010010010110000 \end{pmatrix}$$
- $q = 2, n = 48, k = 6, w = [22, 24, 26]$:
$$\begin{pmatrix} 100100010010111011001011111101000110011100111001 \\ 100010001101011101100101111110000011001110111100 \\ 100001010110101010110110111011100001000111011110 \\ 100000101111010001011111011101110000100011001111 \\ 110000010011101100101111101110011000110001100111 \\ 101000000110111011001011111001100110011100011001 1 \end{pmatrix}$$
- $q = 2, n = 48, k = 6, w = [20, 24, 28]$:
$$\begin{pmatrix} 111011100011101101000010100000010000110111110011 0 \\ 001011011110010000100111110011000100111011100100 \\ 111110001110011011000100001000000101001111100110 1 \\ 010110110101100000000111110111000000011111100010 01 \\ 111101010101110110000000101000000101001101110100 11 \\ 100101101010110010000101110111010000101111010100 10 \end{pmatrix}$$
- $q = 2, n = 48, k = 7, w = [20, 24, 28]$:
$$\begin{pmatrix} 100010111101100101011111000000001000101010011110 1 \\ 001010110111010101110100011001000101000111110001 \\ 000101111011001010111111000100010001010010111101 0 \\ 010100101111100011101000111010000001000111110011 0 \\ 010001111110010110101011100000000100010111001111 0 \\ 101000011111001111010001110100000010001011100110 1 \\ 000000000000111000000011111111111111111111111111 00 \end{pmatrix}$$
- $q = 2, n = 48, k = 8, w = [20, 24, 28]$:
$$\begin{pmatrix} 111111111000000000111111111100000000000000001000000 \\ 000000000111111111111111111110000000000000000100000 \\ 000011111000011111000000111111111111110000000010000 \\ 001100111001100111001110001100001111111110000100 0 \\ 010101001010101001010000110101110011000110000100 \\ 100101010111111111001001001101001010101101000001 0 \\ 111101010001111011100110111000100110101000000001 \\ 101001000000010011001010111001011011011110000000 \end{pmatrix}$$
- $q = 2, n = 48, k = 7, w = [16, 24, 32]$:
$$\begin{pmatrix} 100101110010111001011100101110010111001011000000 \\ 110100000110111111110001100110101101001000101010 0 \\ 100111000010110000000100101010111101110110000111 1 \\ 110101101001000111010000001101111111000110001000 1 \\ 111010000011011111111000110011010110100100000101 0 \\ 100010000111110101001100010000111110101001101101 \\ 111100111100111100111100111100111100111100011110 \end{pmatrix}$$

- $q = 2, n = 48, k = 8, w = [16, 24, 32]$:
$$\begin{pmatrix}
100111101111101111110001110001110111101011101100\\
101000110000110000001001001001000100010110011 0000\\
001010011001011001001110001110001010001011101111\\
001010101010010101111101000010100101111010011001\\
100111111010111010100100100100000111101011100000\\
111001010001010001110110110110001001110111010000\\
001010011001011001110001110001111010001011101100\\
001111010011101100010000101111100000110110101001
\end{pmatrix}$$
- $q = 2, n = 48, k = 9, w = [16, 24, 32]$:
$$\begin{pmatrix}
110110000011101101100000111010100000101100010111 1\\
100111011100100110001000110110010001101111001010\\
110000000011110011111111000001101000011011110010\\
100100011100101001000111001011111101001110100100\\
011001010101101001101010100100111101111001000001\\
001100011110100011000111101010100000101100010111 1\\
110011001100110011001100110010010101010101010110\\
101011011100011010110111000100000001110100111010 0\\
111011101011111111011101011110101110001011000101 1
\end{pmatrix}$$
- $q = 2, n = 48, k = 10, w = [16, 24, 32]$:
$$\begin{pmatrix}
100100110000000101100111011010011000111110110110 0\\
000110000001101000100000001011101111111110011100111\\
110001010110011100100101011100100101011011000101\\
000001101100110001101010001110010101001111111001\\
011100100101100100111010100100111010010101110010\\
111110000011100110001001000110001001001111111000\\
111111100001100011000100100011000100000011111110\\
011110011011101000111001010111000110010010000110\\
001011000000111100101101111110010110110001001011 00\\
101010011001010111101011101000010100011001010110
\end{pmatrix}$$
- $q = 2, n = 48, k = 11, w = [16, 24, 32]$:
$$\begin{pmatrix}
100110100101001011010101010101011010100110100110\\
000110110011001011010101010101011010011001011001\\
000000001111000000000000000000011111111111100000 0\\
010011100000000000000011000011001111000000110011\\
001111000000000110000011000000001111110011000000\\
000000000000100111000000000000001100111100111111 00\\
000000000000011110000000000000000011111111100110011\\
000000000000000000110011000000011110011001111001 1\\
000000000000000000001111000000000111100111111110 0\\
000000000000000000000000110011110011001111110011\\
000000000000000000000000001111001111111001100111 1
\end{pmatrix}$$
- $q = 2, n = 48, k = 12, w = [16, 24, 32]$:
$$\begin{pmatrix}
100101000010110010101010110010110011010010101101 11\\
010011000010110010101010110011010100101100101011 11\\
001111000000000000000000110011000011110011110000 0\\
000000100001100000000011000000111100000001111110 01\\
000000010001100000000011110011110011000000011000 10\\
000000001001110000000011110011001111111111111110 00\\
000000000111100000000011000000111111111100000000 0\\
000000000000001000000011000011110011110011001110 0\\
000000000000000110000111100111111000011000011000\\
000000000000000011001111000011000011111111110000 0\\
000000000000000000011001100001111111110000111100 0\\
000000000000000000000000001111111111001111111000 00
\end{pmatrix}$$
- $q = 2, n = 52, k = 6, w = [24, 26, 28]$:
$$\begin{pmatrix}
0110011010101010101011010100101010010101010101011000 00\\
1110110110000110011011001001100100110011001100010000\\
1101110001100001111000111000011100001111000011001000\\
0011110000011111111100000001111111000000000111111000100\\
0000001111111111111100000000000000011111111111111100 0010\\
0000000000000000000111111111111111111111111111111100 0001
\end{pmatrix}$$
- $q = 2, n = 56, k = 6, w = [26, 28, 30]$:
$$\begin{pmatrix}
01011001100101011111011100100001001101100011111010000100\\
10001111101110110100110100101000000100111100100111110101 0\\
01110111001001000011000111011111101000111100101111101000 0\\
10011111000011000111100111111000111100000111000100100001\\
11011010110001000100011000011100010110111101010010111111\\
00000110101101000110100101000110011111111000011000010011 1
\end{pmatrix}$$
- $q = 2, n = 56, k = 7, w = [24, 28, 32]$:
$$\begin{pmatrix}
11001100110101111111011001000101111101000000110001101110 1\\
01100111011010011111101100101010111010100010110001101110\\
10110010101101101111100110011101011001010001011000110111 1\\
11011001010110110111110011001110101000101000101101010110 11\\
01101100101011111011101001101111010000001010001011110110 1\\
00110111010101111110110010011011110110000101000101111011 0\\
10011011101010111111011001001101111001000011100010011101 1
\end{pmatrix}$$

- $q = 2, n = 56, k = 8, w = [24, 28, 32]$:

$$\begin{pmatrix}
00111011101011010100101010111101100110111001010100001111 \\
01110110010110111001010001111011001101110010101100011110 \\
11101100101101100010100111110110011011100101011000111100 \\
11011001011011010101001011101101110111001010110001111000 \\
10110011110110101010010011011011101110010101100111110000 \\
01100111101101010100100110110111011100111011001011100001 \\
11001110011010111001001001101111111001100110010111000011 \\
10011101110101100010010111011110110011011100101010000111
\end{pmatrix}$$

- $q = 2, n = 56, k = 9, w = [24, 28, 32]$:

$$\begin{pmatrix}
10000000011111000000011111111000000111111110000111111111 \\
01000010011100010000010110010100110111011110010100001010 \\
00100010010110011100000001010011111101010100101001000110 0 \\
0001001000011000011010111111000000110011110000111101001000 \\
00001010101010000110011000100101110110111001000100110010 \\
0000011011101000101110110000000011110000000001110001111100 \\
000000010011110111101110110000011010011000010000111000 10 \\
00000000000000111111111111110000000000000011111111111111 \\
00000000000000000000000000001111111111111111111111111111
\end{pmatrix}$$

- $q = 2, n = 64, k = 7, w = [28, 32, 36]$:

$$\begin{pmatrix}
1010111010110001100001101011011110110111100111100110101000101001 \\
00110000110001011101011101101111111101100100010100011100111100111 \\
0111110001001111011001100101001001100110001001100101000001110111 \\
0001110111000100110010001000100110111101001100100000001110111000 \\
1100000111011000010011000010101011110010000100110011100000111010 \\
0000101101001100010001110111100010010100111011100001111010010111 \\
1111100010001110110011011010000011001110010011001000000001101111
\end{pmatrix}$$

- $q = 2, n = 64, k = 8, w = [28, 32, 36]$:

$$\begin{pmatrix}
10000000001111011010110111000000010010011001110100110001100100011 \\
0100000000100011011110110010000011011010101001110101001010110010 \\
001000000010101110100101111010001110110000000111010000110000111100 \\
000100000001010111010010111101000111011000000011101000011000011110 \\
00001000001010011001001001011010111000011010010011110011001101010 \\
000001010000111111100110100111001010011110100100000000101100000011 \\
00000011010101110011000010001010010110010011101101111000111111011 \\
00000000111101101011011100000001001001100111010011000110010001101
\end{pmatrix}$$

- $q = 2, n = 64, k = 9, w = [28, 32, 36]$:

$$\begin{pmatrix}
10000000001111011010110111000000010010011001110100110001100100011 \\
0100000000100011011110110010000011011010101001110101001010110010 \\
0010000000101100000100000101000011111110011010011100101001111011 \\
000100000001010111010010111101000111011000000011101000011000011110 \\
00001000000101000011111110011010011100101001111010010000000101101 \\
000001010000111111100110100111001010011110100100000000101100000011 \\
00000011001011000110101100001011011111111010011111011111010111100 \\
00000000100011011110110010000001101101010100111010100101011001010 \\
0000000011110110101101110000000100100110011101001100011001000111
\end{pmatrix}$$

- $q = 2, n = 64, k = 7, w = [24, 32, 40]$:

$$\begin{pmatrix}
01001111110001110011101010111000000010111100001010100101001011100 \\
00101111011111100111000100110011001001110100100010010100101110010 \\
101001011111100111001100101011101000010111100000001010110100001110 \\
010101101111110011100110010001110100011011010000000101101010100110 \\
1001011110101110011101010011001000101111000010001001110010111000 \\
111110000000000001111111111000001111111111000000000000100000011111 \\
111110000000000000000000000011111000001111111111111111100000011110
\end{pmatrix}$$

- $q = 2, n = 64, k = 8, w = [24, 32, 40]$:

$$\begin{pmatrix}
10111111101010110101001010001010100011100001111101000011000001101 \\
0111111110101001010101100000101100011100001111100100010100001101 \\
00111111111111111111100110001100011111110001000001101011100011101 \\
110001111111111000000000001111100000011100001000111110100011110101 \\
11101001100100100100000001100011101101111100011100010110010010111 \\
11110100010010010010000001100111110110110110101100010110001010111 \\
1101101000100100100100001010011111011011010110100010110000110111 \\
0011100000000011111111100111111110000000010001111100010001111100110
\end{pmatrix}$$

- $q = 2, n = 64, k = 9, w = [24, 32, 40]$:

$$\begin{pmatrix}
1000110111000011000011000010010100111011001011011110011010011011 \\
01000001000011001111001111011001111101111111011010000110000110000 \\
000001000001001111001111011011111110111111011101100001000011000010 \\
1011001111100111101100111100001000010001011010001000001010111101 \\
1011001011001000101000100110111001101001101101100001110101110100 \\
10001100011100111001010101011111101111111101111000011011010110111 \\
00000111011100101111111011001111101010110000101001011000010011000 \\
0001101100001001100100001001110001001010010101001010001011010000 \\
1111101001010011010101100001011001100100001001111100000011101110
\end{pmatrix}$$

- $q = 2, n = 64, k = 10, w = [24, 32, 40]$:

$$\begin{pmatrix}
100000010000110010000001100001001100100010101011011100000011011001 \\
01000000110111000000100101000010110100011101100101010001000001001 \\
001000011101110011000110100000000010011101010011000000110000010010 \\
00010000110011001110100001001001100000001010110000100110101110000 \\
000010001001001000000110010011000010000000011100010011010011101111 \\
00000101100000010001111011010100101001111011100110111111110000001 \\
00000011000000010110110011011010100010010100100000101101110000010 \\
0000000001110100011000010101000000100001100010101001011001101111 \\
000000000000001111111111111110000000011111111111111111100000000010 \\
0000000000000000000000000000000111111111111111111111111100000000001
\end{pmatrix}$$

- $q = 2, n = 64, k = 11, w = [24, 32, 40]$: $\begin{pmatrix} 0000011010000000100011011111100011101001011110001100100100000000 \\ 0111001100100010000011101000000101110010010110110101000100000000 \\ 0000101110101011000110100111100101110100100101000000000100000000 \\ 0111101110001010001100101011001010000101010011010000000010000000 \\ 1001110100000110100000011010000101011011000100111001100001000000 \\ 1101010010011110010001000110111100000011001100001000100000100000 \\ 0101010001111110001011000001110100001000000011000111100000010000 \\ 0011001111111111000011100000000100101110001111101111111100000001000 \\ 0000111111111111000000011111111000111000000000111111110000000100 \\ 0000000000000001111111111111111000000011111111111111110000000010 \\ 0000000000000000000000000000000111111111111111111111111000000001 \end{pmatrix}$
- $q = 2, n = 64, k = 12, w = [24, 32, 40]$: $\begin{pmatrix} 0000110001101110000100100100100011011000011011011110100000000000 \\ 1011110000100110010000000110001000011110100111011100001000000000 \\ 1010110001001010110010000000101111111000000110010101100100000000 \\ 1111100000001100000001010010011110100001101101110100000001000000 \\ 0111000000000101011011000110001100000011011110011001100001000000 \\ 0000000100001001111110011010010101001101010101010100000010000000 \\ 0101011111010000001000111100111001100010010000010110000000001000 \\ 0011010011001000000111111100111111101111110100110111000000000100 \\ 0000101111000110000000000001111011110000110011110000110000000001000 \\ 0000011111000000111111111111111100000111111000000011111100000000100 \\ 0000000000111111111111111111111100000000000001111111111110000000010 \\ 0000000000000000000000000000000111111111111111111111111100000000001 \end{pmatrix}$
- $q = 2, n = 64, k = 7, w = [16, 32, 48]$: $\begin{pmatrix} 1110001110001101110111000101111010000001000101111010000110010011 0 \\ 0000101111010000111110100101001111100101101101100000110011100010 \\ 0010111101001001011011001011000001111100001001111100101110001001 \\ 0111101000011110000111000111110010110100101000011010110001110001 \\ 1110100000101100001110111111100101001001011000110101100011100010 0 \\ 0111000111000100010001111101000001011101110101000010111101001001 1 \\ 1111010000100011110010101111100101010110100000110101100011100010 \end{pmatrix}$
- $q = 2, n = 64, k = 8, w = [16, 32, 48]$: $\begin{pmatrix} 1000110100011000110100010111001011100101111111000110111001010000 \\ 1101100001001100110101011110000001010110011001011100100011011100 1 \\ 1111011000010010001100101100111011110000010010101110010001111001 \\ 0011110110000101011101110000001010110011101101010110100110100001001 \\ 0011110110000101011100101011001110111100001010110100110100000011 \\ 1000010011110110111000111011110000001010110101110100010100010100 1 \\ 1000010011110101000111010110011101111000000001001011110100011010 1 \\ 0111101100001001000111100000101011001110110100010111101000110011 \end{pmatrix}$
- $q = 2, n = 64, k = 9, w = [16, 32, 48]$: $\begin{pmatrix} 0101011110000001101000011010001101000110100111011101110001001110 1 \\ 0110111110010000000011011011011011011011000101101000111101010000 \\ 1010101010100100011101000110100011010001101010111100101011010101 \\ 1010101010100001101100011010001101000110101010101110010101101010 1 \\ 0101010011100100011101000110100011010001100011111010111001000111 \\ 1100010100110110100010110011101111000001101111010100010110000101 \\ 1111110010011100101111001011100101110010010010000110100100100100 \\ 0001110011011000110111100000101011001110111110001101011000010001 \\ 0000000110110111001101110010111001011100110110000000110110110110 0 \end{pmatrix}$
- $q = 2, n = 64, k = 10, w = [16, 32, 48]$: $\begin{pmatrix} 0010011111000000110010110100111010101001010101111111110000011000 \\ 1001000110100111000010000111101011001111000111011101101001100001 \\ 1110001110010011001001010000001111101111110100010010000110110110 10 \\ 1110000011100110100100010100000100010111110111111011000110101100 \\ 1101100000000001111000000000001010000000000000000101000000000011 100 \\ 0101010010100000111011001001001110101100010011101011010101110001 1 \\ 1010100110001111000111110101110001010100001100010110011100011100 \\ 1111111000011101001010100101010010001011010101001011100001111001 00 \\ 1111110010110110010011110000100100100001111001001100101101001001 \\ 1111110101101001001011100001010010000011110100101101011010100100 \end{pmatrix}$
- $q = 2, n = 64, k = 11, w = [16, 32, 48]$: $\begin{pmatrix} 0100001000110000111100001111111110000000000000000000000000000000 \\ 0010000010000110000000111111111111110000000000000000000000000000 \\ 0001000001000000111111111100001111000000000000000000000000000000 \\ 0000100001111111001100110011001100000000000000000000000000000000 \\ 0000011111010101010101010101010101000000000000000000000000000000 \\ 0000000000000000000000000000000001100001111000011111111110000000 00 \\ 0000000000000000000000000000000000110000001111111111111001100000 00 \\ 0000000000000000000000000000000000000111111111100001111000011000 0 \\ 0000000000000000000000000000000000011111100110011001100110000001100 \\ 0000000000000000000000000000000001111111100110011000011000000000 11 \\ 1111111111111111111111111111111110101010101010101010101010101010 \end{pmatrix}$
- $q = 3, n = 3, k = 3, w = [1, 2, 3]$: $\begin{pmatrix} 001 \\ 112 \\ 210 \end{pmatrix}$
- $q = 3, n = 6, k = 3, w = [3, 4, 5]$: $\begin{pmatrix} 111101 \\ 121011 \\ 100122 \end{pmatrix}$
- $q = 3, n = 9, k = 3, w = [5, 6, 7]$: $\begin{pmatrix} 011011001 \\ 110002111 \\ 100121202 \end{pmatrix}$

- $q = 3, n = 9, k = 4, w = [3, 6, 9]$: $\begin{pmatrix} 100111110 \\ 010201211 \\ 221211121 \\ 112112221 \end{pmatrix}$
- $q = 3, n = 18, k = 4, w = [9, 12, 15]$: $\begin{pmatrix} 111111110000001000 \\ 001112221111100100 \\ 120120120112210010 \\ 002000221120110001 \end{pmatrix}$
- $q = 3, n = 18, k = 5, w = [9, 12, 15]$: $\begin{pmatrix} 111111110000010000 \\ 000011221111001000 \\ 011201010012100100 \\ 111122220012200010 \\ 012021121200000001 \end{pmatrix}$
- $q = 3, n = 18, k = 6, w = [9, 12, 15]$: $\begin{pmatrix} 110011111100100000 \\ 001111122200010000 \\ 010201201111001000 \\ 112112212202000100 \\ 220001011221000010 \\ 122011022001000001 \end{pmatrix}$
- $q = 3, n = 27, k = 4, w = [15, 18, 21]$: $\begin{pmatrix} 111111111111110000000001000 \\ 000011111222221111111000100 \\ 112200122001120011222110010 \\ 120212001120211202012120001 \end{pmatrix}$
- $q = 3, n = 27, k = 5, w = [15, 18, 21]$: $\begin{pmatrix} 011011001111111111111110101 \\ 121000110120001122021011111 \\ 210101012002010120222102222 \\ 100110102211111221112112020 \\ 002201101200222211110001221 \end{pmatrix}$
- $q = 3, n = 27, k = 6, w = [15, 18, 21]$: $\begin{pmatrix} 000111110100110011111111101 \\ 001210101001100122112121011 \\ 012200010011001212122221101 \\ 122000100110012110221221011 \\ 220002001101121100211222101 \\ 200021011010211002111212011 \end{pmatrix}$
- $q = 3, n = 27, k = 5, w = [9, 18, 27]$: $\begin{pmatrix} 111111110000000000000010000 \\ 000000001111111100000001000 \\ 001112220011122211111000100 \\ 120120121201201201122100010 \\ 121202011212020120112200001 \end{pmatrix}$
- $q = 3, n = 36, k = 5, w = [21, 24, 27]$: $\begin{pmatrix} 111001001111011111111011111011101011 \\ 001112110012110222222112022111021101 0 \\ 220010022110221212112212100212210110 \\ 022201100212202121211101212201210101 \\ 100220212011122022122222021221001101 \end{pmatrix}$
- $q = 3, n = 36, k = 6, w = [21, 24, 27]$: $\begin{pmatrix} 101011111110000101100110100101110110 1 \\ 110102021110100101100110100102210210 \\ 211020002111110020010011010010121021 \\ 021101200211012002001001101102012202 \\ 102120210021202200100100110210101120 \\ 010221211002020210010010011122010012 \end{pmatrix}$

References

1. Betsumiya K., Munemasa A.: On triply even binary codes. J. Lond. Math. Soc. **86**(1), 1–16 (2012).
2. Bonisoli A.: Every equidistant linear code is a sequence of dual Hamming codes. Ars Comb. **18**, 181–186 (1984).
3. Bosma W., Cannon J., Playoust C.: The Magma algebra system. I. The user language. J. Symbolic Comput. **24**(3–4), 235–265 (1997).
4. Bouyukliev Iliya, Bouyuklieva Stefka, Kurz Sascha: Computer classification of linear codes. IEEE Trans. Inf. Theory **67**(12), 7807–7814 (2021).
5. Braun M., Kohnert A., Wassermann A.: Optimal linear codes from matrix groups. IEEE Trans. Inf. Theory **51**(12), 4247–4251 (2005).
6. Brouwer A.E.: Parameters of strongly regular graphs. https://www.win.tue.nl/~aeb/graphs/srg/srgtab.html.
7. Brouwer A.E., Cohen A.M., Neumaier A.: Distance-Regular Graphs. Springer-Verlag, Berlin (1989).
8. Brouwer A.E., Haemers W.H.: Spectra of Graphs. Springer Science & Business Media, Berlin (2012).
9. Bruin N., Poonen B., Stoll M.: Generalized explicit descent and its application to curves of genus 3. Forum of mathematics. Sigma **4**, e6 (2016).
10. Bruin N., Stoll M.: The Mordell-Weil sieve: proving non-existence of rational points on curves. LMS J. Comput. Math. **13**, 272–306 (2010).
11. Calderbank R., Kantor W.M.: The geometry of two-weight codes. Bull. Lond. Math. Soc. **18**(2), 97–122 (1986).

12. Chen E.: Two-weight codes. http://moodle.tec.hkr.se/~chen/research/2-weight-codes.
13. Courteau B., Wolfmann J.: On triple-sum-sets and two or three weights codes. Discrete Math. **50**(2–3), 179–191 (1984).
14. Delsarte Ph.: Weights of linear codes and strongly regular normed spaces. Discrete Math. **3**(1–3), 47–64 (1972).
15. Dodunekov S., Guritman S., Simonis J.: Some new results on the minimum length of binary linear codes of dimension nine. IEEE Trans. Inf. Theory **45**(7), 2543–2546 (1999).
16. Dodunekov S., Simonis J.: Codes and projective multisets. Electron. J. Comb. 5, #R37 (1998).
17. Grassl M.: Bounds on the minimum distance of linear codes and quantum codes (2007). http://www.codetables.de.
18. Heinlein D., Honold T., Kiermaier M., Kurz S., Wassermann A.: Projective divisible binary codes. The Tenth International Workshop on Coding and Cryptography 2017 : WCC Proceedings, Saint-Petersburg (2017).
19. Jaffe D.B., Ruberman D.: A sextic surface cannot have 66 nodes. J. Algebraic Geom. **6**(1), 151–168 (1997).
20. Kiermaier M., Kurz S.: Classification of Δ-divisible linear codes spanned by codewords of weight Δ, arXiv preprint arXiv:2011.05872 (2021).
21. Kohnert A.: Constructing two-weight codes with prescribed groups of automorphisms. Discrete Appl. Math. **155**(11), 1451–1457 (2007).
22. Kurz S.: Classification of 8-divisible binary linear codes with minimum distance 24, arXiv preprint arXiv:2012.06163 (2020).
23. Liu X.: Binary divisible codes of maximum dimension. Int. J. Inf. Coding Theory **1**(4), 355–370 (2010).
24. MacWilliams J.: A theorem on the distribution of weights in a systematic code. Bell Syst. Tech. J. **42**(1), 79–94 (1963).
25. Pless V.: Power moment identities on weight distributions in error correcting codes. Inf. Control **6**(2), 147–152 (1963).
26. Shi M., Kiermaier M., Kurz, S., Solé, P.: Three-weight codes over rings and strongly walk regular graphs. Graphs Combin. **38**(3), 23 (2022)
27. Shi M., Solé P.: Three-weight codes, triple sum sets, and strongly walk regular graphs. Des. Codes Cryptogr. **87**(10), 2395–2404 (2019).
28. Simonis J.: Restrictions on the weight distribution of binary linear codes imposed by the structure of Reed–Muller codes. IEEE Trans. Inf. Theory **40**(1), 194–196 (1994).
29. Stoll M.: Implementing 2-descent for Jacobians of hyperelliptic curves. Acta Arith. **98**(3), 245–277 (2001).
30. The LMFDB Collaboration: The L-functions and modular forms database, home page of the l-function $l(s, e)$ for elliptic curve with lmfdb label `50.a3`. https://www.lmfdb.org/EllipticCurve/Q/50/a/3, [Online; accessed 26 April 2022].
31. The LMFDB Collaboration: The L-functions and modular forms database, home page of the l-function $l(s, e)$ for elliptic curve with lmfdb label `10368.j1`. https://www.lmfdb.org/EllipticCurve/Q/10368/j/1, [Online; accessed 26 April 2022].
32. van Dam E.R., Omidi G.R.: Strongly walk-regular graphs. J. Comb. Theory A **120**(4), 803–810 (2013).
33. Ward H.N.: Divisible codes. Arch. Math. (Basel) **36**(1), 485–494 (1981).

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.