




EvenQuads Game and Error-Correcting Codes

Nikhil Byrapuram¹ · Hwiseo Choi¹ · Adam Ge¹ · Selena Ge¹ · Sylvia Zia Lee¹ · Evin Liang¹ · Rajarshi Mandal¹ · Aika Oki¹ · Daniel Wu¹ · Michael Yang¹ · Tanya Khovanova² 

Received: 24 August 2024 / Accepted: 10 August 2025
© The Author(s) 2025

Abstract

EvenQuads is a new card game that is a generalization of the SET game, where each card is characterized by three attributes, each taking four possible values. Four cards form a quad when, for each attribute, the values are the same, all different, or half and half. For any ℓ cards selected from the deck of EvenQuads, it is possible to construct an error-correcting linear binary code of length ℓ and Hamming distance 4, where quads correspond to codewords of weight 4. Using error-correcting codes, we calculate the number of possible quads that can be formed with up to 8 cards. We also estimate the number of cards that do not contain quads for decks of different sizes. In addition, we discuss properties of error-correcting codes built on semimagic, magic, and strongly magic quad squares. This highlights a rich interplay between recreational mathematics games and coding theory and encourages others to explore similar combinatorial games for hidden connections!

Keywords Error-detecting codes · Error-correcting codes · EvenQuads game

Introduction

EvenQuads is a new card game introduced by Rose and Pereira [13] as a generalization of the SET game. The new game was initially called SuperSET, but now it is called EvenQuads.

The original SET deck consists of 81 cards, where each card is characterized by four attributes: number (1, 2, or 3 symbols), color (green, red, or purple), shading (empty, striped, or solid), and shape (oval, diamond, or squiggle). A set is formed by three cards that are either all the same or all different in each attribute. The players try to find sets among given cards, and the fastest player wins.

Similarly, the EvenQuads deck consists of 64 cards, where each card is characterized by three attributes: number (1, 2, 3, or 4), color (red, green, yellow, or blue), and shape (square, icosahedron, circle, or spiral). A quad is made up of four cards that are either all the same, all different, or half and half in each attribute. The players try to find quads among the given cards, and the fastest player wins. One can view the EvenQuads deck as a deck of numbers from 0 to 63

inclusive. Four numbers form a quad if their bitwise XOR is 0 [4, 9].

Error-correcting codes allow us to encode transmissions in such a way that when an error is introduced, the decoder can still completely recover the transmission [2, 6]. The first error-correcting code was introduced in 1950 by Hamming [12]. Since then, many different types of error-correcting codes have been invented serving different goals. For example, error detection and correction using the game of SET was discussed in [11].

This paper introduces linear binary error-correcting codes based on the cards from an EvenQuads deck. The codes help us study quads.

We start with preliminary information about the 64-card EvenQuads game and error-correcting codes in “Preliminaries” section. We then generalize the game by extending the number of cards in the deck to any power of 2.

In “Quads and Error-Correcting Codes” section, we build error-correcting codes using a set of cards in an EvenQuads-2ⁿ deck. We define the quad code of a given set of distinct cards to be a binary linear code such that each bit corresponds to one card in the set, and the weight of each codeword is even. Thus, the length of the quad codes corresponds to the number of cards in the set. Four bits corresponding to four cards forming a quad generate codewords of weight 4. Thus, the number of quads in a given set of

✉ Tanya Khovanova
tanya@math.mit.edu

¹ MIT PRIMES STEP, Cambridge, MA, USA

² MIT, Cambridge, MA, USA

cards corresponds to the number of codewords of weight 4. Moreover, as cards are different, codewords of weight 2 do not exist. Thus, the minimum distance between codewords is 4.

In “Linear Algebra to Help” section, we describe how to construct a quad code from a set of cards. Then in the reverse direction, we prove that we can find a set of cards in an EvenQuads- 2^n deck from any quad code of length ℓ and dimension k if and only if $n \geq \ell - k - 1$. In “Codes and Symmetries” section, we describe symmetries of an EvenQuads deck and show that quad codes correspond to sets of cards up to affine transformations.

In “Possible Number of Quads for a Small Number of Cards” section, we calculate the possible numbers of quads that can be formed with up to 7 cards. Understanding the connection to codewords helps us estimate the minimum and maximum number of quads possible among given cards. For a given number of cards and quads, we also calculate the smallest deck size in which we can have the given number of cards that contain exactly the given number of quads. We use combinatorial methods in this section, but our methods become tedious for more cards.

The “Weight Enumerator” section introduces the weight enumerator of a code. We then demonstrate how it can be applied to simplify the calculation of the possible numbers of quads given 8 cards. We show that the number of possible quads is one of 0, 1, 2, 3, 5, 6, 7, and 14.

We finish by applying error-correcting codes to the EvenQuads when discussing equivalents of two constructions found in the original SET game. In “No-Quads Sets” section, we consider sets of cards from a given deck that do not contain any quads, which are the analogs of the well-known cap sets in the original game of SET. We calculate the maximum size of such a set for small decks and provide bounds for larger decks.

In “Error-Correcting Codes and Quad Squares” section introduces error-correcting codes related to semimagic, magic, and strongly magic quad squares, which were defined in [5]. Semimagic quad squares are 4-by-4 squares of cards where each row and column forms a quad. They are similar to semimagic integer squares, where each row and column has to sum up to the same number. Magic and strongly magic quad squares are the analog of magic SET squares [7, 10]. We calculate the smallest dimension for such codes and the weight enumerator corresponding to this smallest

dimension. We also enumerate such codes in each possible dimension.

Preliminaries

EvenQuads

An EvenQuads deck consists of $64 = 4^3$ cards with different objects. The cards have 3 attributes with 4 values each:

- Number: 1, 2, 3, or 4 symbols.
- Color: red, green, yellow, or blue.
- Shape: square, icosahedron, circle, or spiral.

Figure 1 shows examples of cards.

A *quad* consists of four cards so that for each attribute, the cards must be either all the same, all different, or half and half. Figure 2 shows two quad examples. In the first example, the shape is divided half and half while the number and color are all different. In the second example, the number is all different, while the color and shape are divided half and half.

In EvenQuads, the dealer lays out 12 cards from the deck facing up. Then, without touching the cards, players look for four cards that form a quad. When a player finds a quad, they yell “quad”, and the first person to do so gets to take the four cards as long as they were correct. If the four cards they chose do not form a quad, they have to

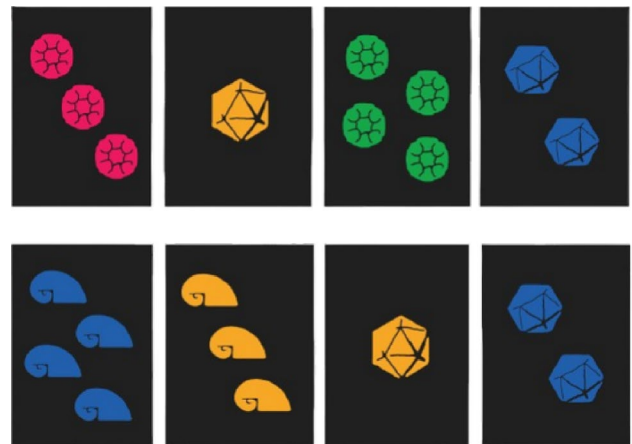
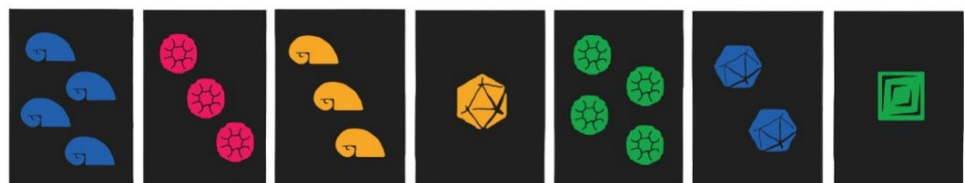


Fig. 2 Quad examples

Fig. 1 Cards from the Even-Quads deck



put the cards back. If the cards do form a quad, then four new cards are drawn from the deck. If players cannot find a quad after a while, the dealer may lay out an additional card. The game ends when no more cards are left in the deck, and there are no possible quads among the remaining cards. The winner of the game is the one with the most quads.

We can assume, as suggested in [9], that each attribute takes values in the set $\{0, 1, 2, 3\}$. Then, four cards form a quad if and only if the bitwise XOR (also called parity sum) of the values in each attribute is zero. This is equivalent to saying that each attribute takes values in \mathbb{Z}_2^2 and four cards form a quad if and only if for each attribute the sum of the vector values in \mathbb{Z}_2^2 is the zero vector. Thus, we can view our cards as vectors in \mathbb{Z}_2^6 . For example, let the first three cards in our quad be represented as 100101, 111101, and 010011. To find the fourth card that completes the quad, we use the bitwise XOR operation: $100101 \oplus 111101 \oplus 010011 = 001011$, which is the missing card.

For generalizations, we can consider an EvenQuads deck of size 2^n . It corresponds to vectors in \mathbb{Z}_2^n . It follows that four vectors $\vec{a}, \vec{b}, \vec{c}$, and \vec{d} form a quad if and only if

$$\vec{a} + \vec{b} + \vec{c} + \vec{d} = \vec{0}.$$

Consider four vectors $\vec{a}, \vec{b}, \vec{c}$, and \vec{d} forming a quad. By a translation, we can assume that \vec{a} is the origin. Then, from the above equation we get $\vec{d} = -\vec{b} - \vec{c} = \vec{b} + \vec{c}$. This means that vector \vec{d} is in the same plane as the origin and vectors \vec{b} and \vec{c} . Thus, four cards form a quad if and only if their endpoints belong to the same plane.

In the rest of the paper, we often use numbers from 0 to $2^n - 1$ inclusive to label the cards in the EvenQuads- 2^n deck. Sometimes, to better visualize quads, we represent them as quaternary strings: we convert each number to base 4 and pad it with zeros on the left to reach a length of $\lceil \frac{n}{2} \rceil$. When n is odd, the first digit in a string representing a card can only be 0 or 1. For example, a quad in the standard deck can be represented as four numbers 0, 21, 42, and 63. The cards in quaternary notation are 000, 111, 222, and 333.

Every quad is defined by three distinct cards. The fourth card in this quad can be uniquely found by bitwise XORing the three cards. In other words, the fourth card is the fourth point in the affine plane, defined by the three initial cards.

In particular, the EvenQuads- 2^n deck contains $\binom{2^n}{3}$ quads. Indeed, we can choose any three cards in $\binom{2^n}{3}$ ways, and these three cards always complete to a quad. Each quad is counted 4 times, so we have to divide by 4.

The corresponding sequence is A016290 in the OEIS database [14]:

1, 14, 140, 1240, 10416, 85344, 690880, 5559680, ...

Given four cards from an EvenQuads- 2^n deck, the probability that they form a quad is $\frac{1}{2^{n-3}}$. This follows from the fact that any three cards from this deck can be uniquely extended into a quad. Thus, given m cards from a Quads- 2^n deck, the

expected number of quads is $\binom{m}{4} \frac{1}{2^{n-3}}$.

Error-Correcting Codes

Before we construct error-correcting codes from EvenQuads cards, we first introduce some background information.

Consider a binary string \mathbf{b} of length ℓ . We can view it as a vector in \mathbb{Z}_2^ℓ . The Hamming distance between two strings of equal length is the number of positions at which the corresponding symbols are different.

Suppose we choose a set of binary strings C from \mathbb{Z}_2^ℓ such that they form a linear subspace and the Hamming distance between any two of them is at least 3. The strings in C are called *codewords*. Such a set forms an error-correcting code of length ℓ .

Suppose a sender sends a message \mathbf{b} that is one of the binary strings in C . We allow for noisy transmission, where not more than 1 bit can be corrupted during the transmission. The receiver receives a binary string \mathbf{b}_1 , where the Hamming distance between \mathbf{b} and \mathbf{b}_1 is not more than 1. In addition, the Hamming distance between \mathbf{b}_1 and any other codeword in C is at least 2. This way, the receiver can uniquely correct the corruption and figure out that the sent message is \mathbf{b} .

We assume that our codes are *linear*, meaning that the set of codewords is closed under addition. The fact that the codewords form a subspace means that the number of codewords is a power of 2. We can define a *weight* of the codeword to be the number of ones in its representation. Linearity of the code means that the smallest Hamming distance between two codewords equals the smallest weight of a nonzero codeword.

Example 1 Consider $C = \{000, 111\}$. This is a code of length 3. Any received message corrupted by not more than one bit can be recovered by a majority vote.

Linear binary codes with a minimum distance of 3 are used a lot. These codes can correct any single-bit error. Many such codes can be turned into codes with a minimum distance of 4 by adding a parity check bit. These codes can also correct any single bit error. But their advantage is that they also detect a double error. They cannot correct the double error though. Such codes are often known as

SECDED (abbreviated from single error correction double error detection).

Hamming codes are efficient error-correcting codes. In the n th Hamming code, the codewords have length $2^n - 1$, and the number of codewords equals the maximum theoretical bound with a minimum distance of 3. Hamming codes can be extended by adding a parity check bit to obtain a minimum distance of 4.

Quads and Error-Correcting Codes

We can analyze the number of quads in a set of cards by converting the set of cards into an error-correcting code. Suppose we are given cards \vec{a}_1 through \vec{a}_ℓ from an EvenQuads- 2^n deck. We construct a linear binary code C of length ℓ . We say $\mathbf{c} = c_1c_2 \cdots c_\ell$ is a codeword if and only if:

- An even number of c_1 through c_ℓ are 1.
- $c_1\vec{a}_1 + \cdots + c_\ell\vec{a}_\ell = \mathbf{0}$.

By our definition, every codeword has an even weight. Moreover, the word with all zeros is always a codeword.

As we require the codewords to have an even number of ones, the Hamming distance between any two codewords has to be even. For our codewords, the minimum distance is 4. Suppose, to the contrary, there are two codewords with a distance of 2. Then their sum is a codeword of weight 2. But a weight of 2 is impossible, as it would indicate that two cards in the set are identical. Thus, we cannot have a codeword with two ones.

Example 2 Suppose we have four cards \vec{a}_1 through \vec{a}_4 , then the length of the code ℓ is 4. If the cards form a quad, the only codewords are 0000 and 1111. The weight of the first codeword is 0, and the weight of the second codeword is 4.

The example above shows that if we have a quad among the given cards \vec{a}_1 through \vec{a}_ℓ , then the corresponding word with four ones for c_1 through c_ℓ is a codeword.

Example 3 Suppose we have seven cards from the standard deck depicted in Fig. 1, then the length of the code, ℓ , is 7. Figure 2 shows two quads that correspond to codewords 0101110 and 1011010. The given set of cards contains 5 more quads corresponding to codewords 1101001, 1110100, 0110011, 1000111, and 0011101. The corresponding code contains one more codeword: 0000000. In this particular example, all codewords, except the trivial one, have weight 4.

We can summarize our preliminary observations above:

- The word with all zeros is always a codeword.
- If we have a quad among the cards, then the corresponding word with four ones is a codeword.
- We cannot have a codeword with two ones.

Example 4 Suppose that we have the cards \vec{a}_1 through \vec{a}_8 of the EvenQuads-8 deck. Because the EvenQuads-8 deck is a complete set, there are $\frac{\binom{8}{3}}{4} = 14$ quads. In addition, 00000000 and 11111111 are also codewords. Thus, there are 16 codewords, and the error correcting code is the same as the extended Hamming code of length 8, in other words, the Hamming code with an additional parity bit added.

In general, the code associated to the EvenQuads- 2^n deck is the extended Hamming code of length 2^n .

It is important to note that c_i s are scalars and \vec{a}_i s are vectors. The cards are in \mathbb{Z}_2^n , while the codeword \mathbf{b} can also be viewed as a vector in the space \mathbb{Z}_2^ℓ . We use the vector symbols for the vectors corresponding to cards and bold symbols for the vectors corresponding to codewords to emphasize that they belong to different spaces.

Definition 1 When we get code C from a sequence of cards $\vec{a}_1, \dots, \vec{a}_\ell$, we say that the sequence of cards *realizes* code C . When the cards belong to the EvenQuads- 2^n deck, we say that the code C *can be realizable* in the deck of size 2^n .

Linear Algebra to Help

From Cards to Codes

We can represent the cards as an n -by- ℓ matrix A , consisting of zeros and ones. Each column is a card, and the i -th row is a set of values of the i th coordinate in each card. Then, the codewords are the vectors belonging to the nullspace of the matrix A . If we denote transpose by T , then the codeword \mathbf{c} satisfies the equation

$$A\mathbf{c}^T = \mathbf{0}.$$

The condition that a codeword \mathbf{c} has to have an even number of ones is equivalent to \mathbf{c} being orthogonal to the vector consisting of all ones in \mathbb{Z}_2^ℓ . Let us denote by $\mathbf{1}$ the vector consisting of all 1s and by $\mathbf{0}$ the vector consisting of all 0s in \mathbb{Z}_2^ℓ .

For every vector \vec{a}_i , where $1 \leq i \leq \ell$, let us consider the augmented vector \vec{a}_i' , where we add one more coordinate with the value 1.

Let us consider matrix A' , where we add a row of all ones to the bottom of matrix A . In other words, the columns of A' are formed by augmented vectors \vec{a}_i' . Then the code C

is exactly the nullspace of A' . This is because the vectors orthogonal to $\mathbf{1}$ are exactly the vectors that contain an even number of ones. In other words, the space of codewords C is the dual space to the space of rows of A' . Sometimes, we denote our code C as C_A to emphasize that it is built by the set of cards corresponding to the matrix A .

Let \mathbf{u} and \mathbf{v} be two vectors in \mathbb{Z}_2^ℓ . The dot product $\mathbf{u} \cdot \mathbf{v}$ is the parity of the number of places where \mathbf{u} and \mathbf{v} are both 1. Let C be a code. Then the dual code C^* is the set of vectors \mathbf{u} such that $\mathbf{u} \cdot \mathbf{v} = 0$ for all vectors \mathbf{v} of C . From standard linear algebra [1], we get the following proposition.

Proposition 1 *Let C be a linear binary code of length ℓ that contains 2^k codewords. Then C^* is a linear binary code with $2^{\ell-k}$ codewords. Furthermore, $C^{**} = C$.*

It follows that given the code C_A , its dual code is a span of rows of A' .

Note that a linear binary code has the Hamming distance of at least 4 between codewords if and only if the minimum number of ones in a nonzero codeword is at least 4. Let us call a linear binary code a *quad* code if each codeword has an even number of ones, and the minimum number of ones in a nonzero codeword is at least 4. In particular, code C_A described above is a quad code.

From Codes to Cards

Suppose we have a quad code C of length ℓ and dimension k . We want to construct a set of cards corresponding to it. We start with the known connection between the number of codewords and code length [12].

Lemma 2 *Let C be a binary code of length ℓ with a minimal Hamming distance of at least 4. Then C contains at most $\frac{2^{\ell-1}}{\ell}$ codewords.*

In other words,

$$k \leq \ell - 1 - \lceil \log_2 \ell \rceil.$$

We are ready to prove our theorem that describes when we can find cards that realize the given code.

Theorem 3 *Let C be a quad code of length ℓ containing 2^k codewords, where $\ell > 0$. Then, the code is realizable in an *EvenQuads-2ⁿ deck* if and only if*

$$n \geq \ell - k - 1.$$

Proof We start with the dual space C^* . This space contains vector $\mathbf{1}$ and has dimension $\ell - k$. We pick $\ell - k - 1$ independent vectors that together with $\mathbf{1}$ span C^* .

Suppose $n < \ell - k - 1$, then the matrix A' for any set of cards has fewer than $\ell - k$ rows. Thus, its nullspace has a dimension greater than k , implying that C cannot equal its nullspace.

For the other direction, suppose $n \geq \ell - k - 1$, then in addition $n \geq \lceil \log_2 \ell \rceil$ by Lemma 2. That means we can find n vectors that span C^* by adding more vectors, which can be repeat vectors, to the ones we found. We arrange these n vectors as rows of matrix A . The columns of this matrix are our cards. It is left to show that the columns are different. It follows from the fact that C is the nullspace of matrix A' . If the set of cards had identical cards, then C would have contained a codeword of weight 2. As, by our assumption, this is not the case, all the cards must be different. \square

Codes and Symmetries

There is a natural approach to symmetries of quads using linear algebra. So far we looked at cards as elements of a vector space \mathbb{Z}_2^n . A more conceptual way to do this is to see them as elements of the corresponding affine space (which is the same as a vector space, but without choosing the origin). This view emphasizes that the cards are equivalent to each other. The symmetries of the space are affine transformations.

Affine transformations are transformations of a vector space that consist of parallel translations and invertible linear transformations. Affine transformations preserve lines and parallelism, but may not preserve distances and angles. Affine transformations can be decomposed into translations, reflection, scaling, rotation, and shearing.

We can view an affine transformation as a pair (M, \vec{t}) , where M is an invertible n -by- n matrix over the field \mathbb{F}_2 and \vec{t} is a translation vector in \mathbb{Z}_2^n . The pair acts on vector \vec{a} as $\vec{y} = M\vec{a} + \vec{t}$. We can simplify this description using augmented vectors. Indeed, $\vec{y} = M\vec{a} + \vec{t}$ is equivalent to the following

$$\begin{bmatrix} \vec{y}^T \\ 1 \end{bmatrix} = \begin{bmatrix} M & \vec{t}^T \\ 0 \dots 0 & 1 \end{bmatrix} \begin{bmatrix} \vec{a}^T \\ 1 \end{bmatrix}.$$

Suppose \vec{a}' , \vec{t}' , and \vec{y}' are augmented vectors for \vec{a} , \vec{t} , and \vec{y} . Then, we can express the affine transformation as

$$\vec{y}'^T = P\vec{a}'^T,$$

where P is an invertible $(n + 1)$ -by- $(n + 1)$ matrix above that we got from M by adding a zero row at the bottom and then attaching \vec{t}'^T as the last column.

Affine transformations preserve quads. Indeed if \vec{a}_i , for $1 \leq i \leq 4$ form a quad then the transformation produces four vectors $\vec{b}_i = M\vec{a}_i + \vec{t}$. We have

$$\begin{aligned} \sum_{i=1}^4 \vec{b}_i &= \sum_{i=1}^4 (M\vec{a}_i + \vec{t}) = \sum_{i=1}^4 M\vec{a}_i + \sum_{i=1}^4 \vec{t} \\ &= M \sum_{i=1}^4 \vec{a}_i + 4\vec{t} = 0, \end{aligned}$$

proving that vectors \vec{b}_i form a quad too.

Theorem 4 *If we change a set of cards according to an affine transformation, the corresponding code does not change. Moreover, if two sets of cards correspond to the same code, then one of the sets can be achieved from the other by an affine transformation.*

Proof Consider ℓ cards \vec{a}_i , for $1 \leq i \leq \ell$, and the corresponding matrix A' and code C_A , where C_A is the nullspace of A' . Applying an affine transformation (M, \vec{t}) to vectors \vec{a}_i , we get vectors $\vec{b}_i = M\vec{a}_i + \vec{t}$. We denote the new augmented matrix as B' . We know that $B' = PA'$. Thus, the nullspaces of A' and B' are the same. It follows that the new code C_B is the code C_A before the transformation.

For the second part, consider two sets of cards corresponding to the same code C . That means the augmented matrices A' and B' for both sets have the same nullspace C . It follows that the rows of A' and B' span the same space. That means we can represent the rows of A' as a linear combination of rows of B' . Thus, there exists a matrix M , such that $A' = PB'$. Moreover, the last rows of A' and B' are the same. Thus, we can find P , such that all the elements of the last row are zero, except for the last element, which is 1. Therefore, P represents the affine transformation in our space. \square

It follows that if two sets of cards correspond to the same code, the number of quads in each set is the same.

Possible Number of Quads for a Small Number of Cards

We are interested in the number of possible quads given ℓ cards. Suppose that there exists a set of cards that produces q quads with ℓ cards from an EvenQuads- 2^n deck. By adding more cards to the deck, we can conclude that any larger deck can also have ℓ cards with q quads. Thus, it is enough to find the smallest such deck. Denote by $D(\ell, q)$ the smallest deck size 2^n such that there exist ℓ cards from the EvenQuads- 2^n deck that form exactly q quads. If such a deck does not exist, we define $D(\ell, q)$ as ∞ .

Number $A(\ell, 4)$ describes the maximum number of codewords in a binary linear code of length ℓ with the Hamming distance 4. This number is described by sequence A005864 in the OEIS [14], where the sequence starts with index 1:

1, 1, 1, 2, 2, 4, 8, 16, 20, 40, 72, 144, ...

As each quad gives us a codeword, and there is also an all-zero codeword, the maximum number of quads among ℓ cards is bounded by $A005864(\ell) - 1$. Thus, for q greater than $A005864(\ell) - 1$, we have $D(\ell, q) = \infty$.

We start by considering small examples without using the machinery of error-correcting codes. We cannot have a quad if we have fewer than 4 cards. The next two examples cover cases for 4 and 5 cards.

Example 5 Suppose we have 4 cards. If the deck is of size 4, then we have 1 quad. If it is a bigger deck, we can have 0 quads by choosing any three cards, and then for the fourth card, we can pick any other card that avoids the quad. As $A005864(4) = 2$, the maximum number of quads is not more than 1, thus we have covered all cases.

Example 6 Suppose we have 5 cards. It follows that the deck size is at least 8. As $A005864(5) = 2$, the maximum number of quads is not more than 1. We can prove directly that more than 1 quad is not possible. Indeed, if there are two quads, their intersection has 3 cards. But 3 cards define a quad uniquely. We can pick any quad and an extra card for an example of 1 quad among 5 cards; for example, 0, 1, 2, 3, and 4. If the deck size is 8, then 5 cards always contain a quad: we leave it for the reader to check this. For a larger deck, we can use cards 0, 1, 2, 4, and 8 for an example of 5 cards without a quad.

Table 1 shows $D(\ell, q)$ for $\ell < 8$. The first column describes the number of cards, and the first row describes the number of quads. If the number of cards is ℓ and the number of quads is q , the entry corresponding to ℓ and q shows $D(\ell, q)$. We replaced the infinity sign with an empty entry so as not to clutter the table.

Proposition 5 *Table 1 shows $D(\ell, q)$ for $\ell < 8$.*

Proof The cases of 1 through 5 cards are discussed above. Given the number of quads, we want to find a corresponding quad code with the maximum possible number of codewords, as by Theorem 3 they are realizable in smaller decks.

Suppose we have 6 cards with 0 quads. The corresponding quad code can have at most two codewords, 000000 and 111111. By Theorem 3, the code is realizable for $n = 4$.

Suppose we have 6 cards with q quads, where $q > 0$. Then, 111111 cannot be a codeword, as it has a distance of 2 from any codeword corresponding to a quad. Since the code is linear, the total number of codewords is a power of two, bounded by $A005864(6) = 4$. So, the total number of

Table 1 $D(\ell, q)$ for $\ell < 8$

# of Cards	0	1	2	3	4	5	6	7
1	1							
2	2							
3	4							
4	8	4						
5	16	8						
6	16	16		8				
7	32	32	16	16				8

codewords, in this case, can be 2 or 4, with the corresponding number of quads being 1 or 3.

The code with 2 codewords can be 000000 and 001111. By Theorem 3, the code is realizable in the deck of size 16. The code with 4 codewords can be the following: 000000, 001111, 110011, and 111100. By Theorem 3, the code is realizable in the deck of size 8.

Suppose we have 7 cards with q quads. The number of codewords is a power of 2; and $A005864(7) = 8$. It means more than 7 quads is impossible. First, we show that 4, 5, or 6 quads are impossible.

We start by showing that we cannot have more than 1 codeword of weight 6. Indeed, two different codewords of length 7 and having six ones have to differ in exactly two places. That means the Hamming distance between them is 2, which is a contradiction.

Suppose we have at least 4 quads, then the total number of codewords is at least 8. As we have 0 or 1 codewords of weight 6, we have either 6 or 7 codewords of weight 4, proving that 4 or 5 quads are impossible.

Suppose we have 6 quads, meaning that we have 6 codewords of weight 4 and one codeword of weight 6. Without loss of generality, we can assume that the last digit of the codeword c of weight 6 is zero. It follows that the codewords of weight 4 have to have 1 as the last digit, as otherwise, they will be at distance 2 from the codeword c . Since the codes are linear, adding two different codewords a_1 and a_2 of weight 4 should give us a working nonzero codeword. However, since the last digit is 1 in both codewords, the sum will have a 0 in the last digit. Thus, it must equal c . If there is another codeword a_3 of weight 4, then again we see that $a_1 + a_3 = c = a_1 + a_2$. Thus, $a_3 = a_2$, creating a contradiction.

The remaining possibility is 7 quads, or, equivalently, 7 codewords of weight 4. An example of these 7 codewords is as follows: 1111000, 1100110, 0011110, 1010011, 0101011, 0110101, and 1001101. This code is realizable in a deck of size 8.

For 0 quads, the largest code contains 2 codewords: 0000000 and 0111111. By Theorem 3, it is realizable in a deck of size 32. For 1 quad, the code up to symmetries contains two codewords, 0000000 and 0001111, and cannot

contain more codewords with weight 4. As we saw before, it cannot contain more than 1 codeword of length 6. Given that the total number of codewords is a power of 2, the code must contain exactly 2 codewords. By Theorem 3, the code is realizable in a deck of size 32. For 2 or 3 quads, the code needs to have 4 codewords. The code 0000000, 1111000, 0001111, and 1110111 corresponds to 2 quads, while the code 0000000, 1111000, 1100110, and 0011110 corresponds to 3 quads. By Theorem 3, these codes are realizable in a deck of size 16. \square

Looking at Table 1, one can notice some patterns.

Proposition 6 *Suppose that $D(\ell, q) = 2^n$ for some values of ℓ, q , and n . Then $D(\ell + 1, q)$ exists and is not greater than 2^{n+1} .*

Proof Suppose that $D(\ell, q) = 2^n$ for some values of ℓ, q , and n . Let us denote the set of ℓ cards in the deck of size 2^n with q quads as S .

First proof. We add an extra card to this set S with a numerical value 2^n , forming a new set S' . The new card cannot form quads with any set of cards in S . Thus, the number of quads in set S' is q , implying that $D(\ell + 1, q) \leq 2^{n+1}$.

Second proof. Consider a quad code corresponding to the set of cards S . We append 0 to each codeword to get a quad code of length $\ell + 1$ and the same dimension as code C . By Theorem 3, this code is realizable in the deck of size at least 2^{n+1} . As this code has the same number of codewords of weight 4 as code C , the cards that realize the code form q quads, implying $D(\ell + 1, q) \leq 2^{n+1}$. \square

Now, we want to provide examples of sets of cards for all nonempty cells in Table 1. We already described the examples for up to 5 cards. In the proof of Proposition 6, we also showed how to build an example for an entry in the table that is twice the entry in the previous row and the same column. Thus, we only need to provide examples for 6 cards and 0 quads, 6 cards and 3 quads, 7 cards and 2 quads, and 7 cards and 7 quads. All these examples are realizable in the EvenQuads-16 deck and shown in Table 2 using quaternary notation.

Table 2 Card examples

# of cards	# of quads	Cards
6	0	00, 01, 02, 10, 20, 33
	3	00, 01, 02, 10, 13, 03
7	2	00, 02, 10, 13, 20, 21, 32
	7	00, 01, 02, 03, 10, 11, 12

Weight Enumerator

We calculated possible numbers of quads for a small number of cards. To make more progress, we want to introduce the weight enumerator.

The *weight enumerator* $W(x, y)$ of a binary code C is given by a polynomial

$$W(x, y) = \sum_{c \in C} x^{\ell - w(c)} y^{w(c)},$$

where ℓ is the length of the code, $w(c)$ is the weight of $c \in C$, and the sum runs over all codewords. Recall that the length of a quad code is the number of cards that define it.

Example 7 Suppose four cards form a quad as in Example 2 above. The weight enumerator is $W(x, y) = x^4 + y^4$. Suppose we have 6 cards with q quads. Then, from Proposition 5 and its proof, the associated weight enumerator has to be $x^6 + qx^2y^4$, where q is either 0, 1, or 3.

There is a powerful method that can help us with more advanced examples. Let us define the dual polynomial of W by W^* , where

$$W^* = \frac{1}{W(1, 1)} W(x + y, x - y).$$

MacWilliams’s statement shows how the dual code’s weight enumerator relates to the original code’s weight enumerator.

Lemma 7 (MacWilliams identity) Let C be a linear binary code and W be its weight enumerator. Then the dual code C^* has the weight enumerator W^* .

In particular, W^* is a polynomial with non-negative integer coefficients. Note that $W(1, 1) = |C| = 2^k$ is the number of codewords in C because $W(1, 1)$ is the sum of the coefficients of the polynomial $W(x, y)$. The dual code C^* is a linear binary code where each codeword has the same length ℓ as the original code.

We start with an example of 7 cards. We already covered this example in “Possible Number of Quads for a Small Number of Cards” section. However, here, we want to show how to use the weight enumerator.

Example 8 Suppose we have 7 cards with q quads, where q is 4, 5, or 6. Then, the corresponding code has to have 8 codewords, and the associated weight enumerator has to be $x^7 + qx^3y^4 + (7 - q)xy^6$. Then, the dual weight enumerator is

$$\frac{1}{8}((x + y)^7 + q(x + y)^3(x - y)^4 + (7 - q)(x + y)(x - y)^6).$$

Ignoring the multiplier $\frac{1}{8}$ we get

$$8x^7 + (4q - 28)x^6y + (84 - 12q)x^5y^2 + 8qx^4y^3 + 8qx^3y^4 + (84 - 12q)x^2y^5 + (4q - 28)xy^6 + 8y^7.$$

The coefficient $4q - 28$ is negative for given values of q . Therefore, such codes cannot exist.

Proposition 8 For 8 cards, the possible numbers of quads are 0, 1, 2, 3, 5, 6, 7, and 14, realizable in the smallest decks of sizes 64, 32, 32, 32, 16, 16, 16, and 8, respectively.

Proof We already know that 0, 1, 2, 3, and 7 quads are possible from Proposition 6. We also know that the maximum number of codewords is bounded by A005864(8), which is 16. The coefficient of y^8 in the weight enumerator is at most 1, as only one codeword of weight 8 might exist. Moreover, if 11111111 is a codeword, then there are no codewords of weight 6.

We know that the EvenQuads-8 deck has 14 quads, the maximum possible number, as every three cards complete to a quad.

Suppose the dimension of the corresponding code is 4; that is, there are 16 codewords. We consider possible weight enumerators. The enumerator $x^8 + 15x^4y^4$ is impossible, as 15 quads are impossible. The enumerator $x^8 + 14x^4y^4 + y^8$ corresponds to 14 quads, which we already discussed. We are left to investigate enumerators $x^8 + qx^4y^4 + (15 - q)x^2y^6$, where q is the number of quads. The dual weight enumerator (ignoring the multiplier $1/16$) is

$$\begin{aligned} &(x + y)^8 + q(x + y)^4(x - y)^4 + (15 - q)(x + y)^2(x - y)^6 \\ &= 16x^8 + (-52 + 4q)x^7y + (88 - 8q)x^6y^2 + (116 - 4q)x^5y^3 \\ &\quad + (-80 + 16q)x^4y^4 + (116 - 4q)x^3y^5 \\ &\quad + (88 - 8q)x^2y^6 + (-52 + 4q)xy^7 + 16y^8. \end{aligned}$$

The coefficient $-52 + 4q$ is negative for $q < 13$, and the coefficient $88 - 8q$ is negative for $q > 11$, so the code with such an enumerator does not exist. It follows that we cannot have 8 to 13 quads inclusive.

Suppose the dimension of the corresponding code is 3; that is, there are 8 codewords. We know that it is possible to have 7 quads with 7 cards. That means it is possible to have the same number of quads with 8 cards. We know that the dimension of the corresponding code cannot be 4; therefore,

it has to be 3. It follows that 7 quads with 8 cards are realizable in the EvenQuads-16 deck.

For fewer quads, we need to check the weight enumerators $x^8 + qx^4y^4 + (7 - q)x^2y^6$ and $x^8 + 6x^4y^4 + y^8$.

Consider the dual enumerator for the first one:

$$(x + y)^8 + q(x + y)^4(x - y)^4 + (7 - q)(x + y)^2(x - y)^6$$

$$= 8x^8 + (-20 + 4q)x^7y + (56 - 8q)x^6y^2 + (84 - 4q)x^5y^3 + (16q)x^4y^4 + (84 - 4q)x^3y^5 + (56 - 8q)x^2y^6 + (-20 + 4q)xy^7 + 8y^8.$$

The coefficient $-20 + 4q$ is negative for $q < 5$, so we cannot have a code of dimension 3 with fewer than 5 quads. Now, we show that 5 and 6 quads are possible.

We start with 5 quads. Our weight enumerator suggests that we have two codewords of weight six. Without loss of generality, we can assume that these are codewords 00111111 and 11001111. We now pick one codeword of weight 4 such that it is distance 4 from both the codewords of weight 6. We use the codeword 01010011. By linearity, the other codewords are 11110000, 01101100, 10011100, and 10100011. By Theorem 3, this code is realizable in the EvenQuads-16 deck.

The 6 quad set does have an elegant associated code, where the codewords are

00000000, 11110000, 00001111, 11001100, 00110011, 00111100, 11000011, 11111111.

Again, this code is realizable in the EvenQuads-16 deck.

Suppose we have 0 quads; then the code can contain codewords of length 6 or 8. Only one codeword of length 8 exists. In addition, if the code has two different codewords of length six, their bitwise XOR has length 2, 4, or 8. Each of these lengths is forbidden. Thus, the maximum dimension of such code is 2. By Theorem 3, this code is realizable in the EvenQuads-64 deck.

Suppose we have 1 quad. Without loss of generality, we can assume that the corresponding codeword is 11110000. We cannot have a codeword of weight 8, as otherwise 00001111 will also be a codeword, thus adding another quad. Then, if there is a weight-6 codeword, its last 4 digits must be 1. Without loss of generality, we can assume that it is 00111111. Then the only other possible weight-6 codeword is 11001111. Thus, the maximum dimension is 4. By Theorem 3, this code is realizable in the EvenQuads-32 deck.

Suppose we have 2 quads. Per our previous discussion, the dimension of the code has to be 2. One example of such code is 00000000, 00001111, 11110000, and 11111111. This code is realizable in the EvenQuads-32 deck.

Suppose we have 3 quads. Again, the code dimension has to be 2. One example of such code is 00000000, 00001111, 00110011, and 00111100. This code is realizable in the EvenQuads-32 deck.

Thus, the only possible numbers of quads in a set of 8 cards are 0, 1, 2, 3, 5, 6, 7, and 14, and they are realizable in the decks of sizes 64, 32, 32, 32, 16, 16, 16, and 8, respec-

tively. □

Example 9 The example of 5 quads among 8 cards in EvenQuads-16 deck is, using quaternary notation, 00, 01, 02, 03, 10, 20, 30, and 33. The example of 6 quads among 8 cards in EvenQuads-16 deck is, using quaternary notation, 03, 11, 12, 13, 21, 30, 31, and 33.

The weight enumerator was very useful to exclude some cases. However, 8 cards are not too many, and it might be possible to prove the proposition above without the weight enumerator. For example, one can prove that the maximum number of codewords of weight 6 is 2. Indeed, the 2 zeros in two different codewords of weight 6 cannot overlap, as otherwise, adding them will give a codeword of weight 2. For the sake of contradiction, we assume there are 3 codewords of weight 6. Without loss of generality, assume these 3 codewords are 00111111, 11001111, and 11110011. Then, we can add the three codewords to get a codeword with 2 ones, which is a contradiction.

No-Quads Sets

Suppose we want to find sets of cards that do not contain any quads. We call such sets *no-quads* sets. These sets are similar to no-sets in the game of SET, which are also called cap sets [3]. Mathematicians are very interested in finding the maximum number of cards in a no-set. Similarly, they are interested in finding the number of cards in a no-quads set. Let $F(n)$ be the maximum number of cards in a no-quads set in the EvenQuads- 2^n deck. For the standard deck, it was shown in [9] that $F(6) = 9$.

Translating this to codes, we can say that we are looking for quad codes of length ℓ that do not contain codewords of length 4. In other words, we are looking for quad codes with a minimum distance of 6. A standard technique gives a bijection between quad codes of length ℓ and distance at least 6 and linear binary codes of length $\ell - 1$ and distance

at least 5. We can remove the last digit from a quad code C to get a linear code of length $\ell - 1$ where the distance decreases by not more than one. On the other hand, given a linear binary code D of length $\ell - 1$ and smallest positive weight 5, we can obtain a quad code by appending a parity-check bit to the end of every codeword in D . Thus, our task is equivalent to finding linear binary codes of length $\ell - 1$ with a minimum distance of 5.

Small Decks

Let $B(\ell)$ be the largest possible number of codewords of length ℓ in a linear binary code with a minimum distance of at least 6. Table 3 shows the values of $B(\ell)$ for $\ell \leq 12$, which we manually calculated.

We give examples of codewords in $B(\ell)$ where ℓ is 1, 6, 9, and 12. Other examples with the same values of $B(\ell)$ can be constructed by appending zeros. When $\ell = 1$, the only codeword is 0. When $\ell = 6$, the 2 codewords are 000000 and 111111. When $\ell = 9$, an example of a code with 4 codewords is 000000000, 111111000, 000111111, and 111000111. When $\ell = 12$, then $B(12) = 8$, and an example is the code with codewords 000000000000, 111111000000, 000000111111, 111111111111, 000111111000, 111000111000, 111000000111, and 000111000111.

Theorem 9 *If $\ell > 0$, we have $F(\ell - \log_2 B(\ell) - 1) \geq \ell$. In addition, if $B(\ell) = B(\ell + 1)$, then $F(\ell - \log_2 B(\ell) - 1) = \ell$.*

Proof Consider a quad code C of length ℓ with $B(\ell)$ codewords. Theorem 3 implies there exists a set of cards in the EvenQuads- $2^{\ell - \log_2 B(\ell) - 1}$ deck that realizes C . Since all codewords of C have at least 6 ones, this set of cards must be a no-quad set. Since C has length ℓ , this set contains ℓ cards, proving the first part of the theorem.

Assume to the contrary that $B(\ell) = B(\ell + 1)$ but $F(\ell - \log_2 B(\ell) - 1) > \ell$, i.e. the maximum number of cards from a deck with size 2^n in a no-quads set has more than ℓ cards where, from the definition of $F(n)$, we have $n = \ell - \log_2 B(\ell) - 1$. So, there is a code C corresponding to this no-quads set that has length more than ℓ . From the definition of $B(\ell)$, the number of codewords in C is not more than $B(\ell + 1) = B(\ell)$. By

Theorem 3, code C is only realizable in a deck of dimension $n \geq \ell + 1 - \log_2 B(\ell) - 1 = \ell - \log_2 B(\ell)$. This contradicts the fact that $n = \ell - \log_2 B(\ell) - 1$. □

We can use Theorem 9 and Table 3 to calculate the no-quads sets for small decks. Table 4 shows such values, where the first row is n , corresponding to the deck size of 2^n , and for the second row we choose ℓ so that $B(\ell) = B(\ell + 1)$ and $n = \ell - \log_2 B(\ell) - 1$. This allows us to calculate $F(n)$ precisely.

The lemma above confirms that $F(1) = 2$ (when $\ell = 2$), $F(2) = 3$ (when $\ell = 3$), $F(3) = 4$ (when $\ell = 4$), $F(4) = 6$ (when $\ell = 6$), $F(5) = 7$ (when $\ell = 7$), $F(6) = 9$ (when $\ell = 9$), $F(7) = 10$ (when $\ell = 10$), and $F(8) = 12$ (when $\ell = 12$).

Bounds for No-Quads Sets

In [9], the size of a no-quads set was only calculated for a standard deck. Here, we provide bounds for any deck size using bounds from coding theory [8]. In particular, we use the notion of a Hamming bound. Let $A_q(\ell, d)$ denote the maximum possible size of a q -ary block code C of length ℓ and minimum Hamming distance d . Then, the Hamming bound is:

$$A_q(\ell, d) \leq \frac{q^\ell}{\sum_{k=0}^t \binom{\ell}{k} (q-1)^k},$$

where $t = \lfloor \frac{d-1}{2} \rfloor$.

In particular, this bound estimates $B(\ell)$ for larger values: $B(\ell) \leq A_2(\ell - 1, 5)$, which allows us to get a bound for any sized deck.

Theorem 10 *If a no-quad set of ℓ cards exists in some deck, then the number of cards in the deck must be at least $\frac{\ell^2 - \ell + 2}{2}$.*

Proof Consider a no-quads set with ℓ cards. It corresponds to a quad code of length ℓ with a minimal distance of at least 6. Delete any bit, producing a binary code of length $\ell - 1$ with a minimal distance of at least 5. In the Hamming

Table 3 Values of $B(\ell)$ for small ℓ

ℓ	1	2	3	4	5	6	7	8	9	10	11	12
$B(\ell)$	1	1	1	1	1	2	2	2	4	4	4	8

Table 4 Values of $F(\ell)$ for small ℓ

n		1	2	3	4	5	6	7
ℓ and $F(n)$		2	3	4	6	7	9	10

bound above, we substitute $q = 2, d = 5$, and use $\ell - 1$ for the length of the code to get

$$A_2(\ell - 1, 5) \leq \frac{2^{\ell-1}}{\sum_{k=0}^2 \binom{\ell-1}{k} (2-1)^k} = \frac{2^{\ell-1}}{\binom{\ell-1}{0} + \binom{\ell-1}{1} + \binom{\ell-1}{2}} = \frac{2^\ell}{\ell^2 - \ell + 2}.$$

The dimension k of the code satisfies $2^k \leq \frac{2^\ell}{\ell^2 - \ell + 2}$ implying that $2^{\ell-k} \geq \ell^2 - \ell + 2$. Then, by Theorem 3 this code can be realized in the deck with at least $2^{\ell-k-1} = 2^{\ell-1} \cdot \frac{\ell^2 - \ell + 2}{2^\ell} = \frac{\ell^2 - \ell + 2}{2}$ cards. \square

The values of $\frac{\ell^2 - \ell + 2}{2}$ for ℓ from 1 to 20 are
 1, 2, 4, 7, 11, 16, 22, 29, 37, 46, 56, 67,
 79, 92, 106, 121, 137, 154, 172, 191.

They start a famous sequence A000124 in the OEIS database [14] known as the lazy caterer’s numbers. The sequence describes the maximum number of pieces formed when slicing a pizza with ℓ straight cuts. The sequence can also be described as triangular numbers plus 1.

We summarize the lower bounds for small deck dimensions in Table 5, where we put the value in bold when we know the bound is tight. The value in the second row is $\lceil \log_2 A000124(\ell) \rceil$.

On the other hand, we can estimate when the no-quads set of cards has to exist.

Theorem 11 *If $\binom{\ell}{4} + 3 < 2^n$, there exists a no-quad set of ℓ cards in an EvenQuads- 2^n deck.*

Proof We use the probabilistic method. Take a random set of ℓ cards. The number of sets of four cards among these cards is $\binom{\ell}{4}$. Each has a probability of $\frac{1}{2^{n-3}}$ of being a quad. So the probability that a quad exists among our ℓ cards is at most $\frac{\binom{\ell}{4}}{2^{n-3}}$. Consequently, if $\binom{\ell}{4} < 2^n - 3$, then there must exist a no-quad set of k cards. \square

Table 5 Lower bounds for the deck dimension given the size of no-quads set

ℓ	4	5	6	7	8	9	10	11	12	13	14	15
n	3	4	4	5	5	6	6	6	7	7	7	7

We summarize the upper bounds for small sizes of no-quads sets in Table 6, where we put the value in bold when

we know the bound is tight.

Error-Correcting Codes and Quad Squares

People were fascinated with magic squares for centuries. A classic magic square is a square table of integers where each row, column, and diagonal sum to the same number. Magic squares can naturally be extended to EvenQuads. We call a *magic quad square* is a 4-by-4 square of EvenQuads cards such that each row, column, and diagonal forms a quad. A simpler notion is a semimagic integer square, where the requirement for the diagonals is dropped. Thus, a semimagic integer square has its rows and columns sum to the same number. A *semimagic quad square* as a 4-by-4 square of EvenQuads cards such that each row and column forms a quad. We studied quad squares in our other paper [5].

The notion of a magic quad square is similar to the notion of a magic SET square. A *magic SET square* is a 3-by-3 square of SET cards such that each row, column, and diagonal form a set [7, 10]. Thus, each magic SET square has to contain 8 sets. Interestingly, each magic SET square contains 4 additional sets formed by broken diagonals. The formal rule is that any three cards in a magic SET square such that for each coordinate, the values are the same or different, have to form a set. In other words, the values of each coordinate have to satisfy the rules of SET.

When we move to EvenQuads, magic quad squares are natural generalizations of magic SET squares. However, they do not contain extra quads. Thus, it makes sense to consider a different generalization, called a *strongly magic quad*

Table 6 Upper bounds for the number of cards in a no-quads set given the deck dimension

n	3	4	5	6	7	8	9	10
ℓ	4	5	6	7	8	10	12	14

square. In [5], a *strongly magic quad square* was defined to be a square such that for any four cards, if their x -coordinates are all the same, half and half, or all different, and their y -coordinates are all the same, half and half, or all different, then the four cards form a quad. Equivalently, if the coordinates of four cards form a quad, the cards form a quad.

For each type of square, we construct an error-correcting code. From semimagic to magic to strongly magic quad squares, the number of constraints increases. Thus, the dimensions of the corresponding codes increase too.

We start with semimagic quad squares. We can index the 16 cards in a square by numbers from 1 to 16 going left-to-right and top-to-bottom. Then, the property that the set of cards forms a semimagic square is equivalent to the fact that the corresponding quad code contains the four codewords

1111000000000000, 0000111100000000,
0000000011110000, 0000000000001111

corresponding to rows, and the four codewords

1000100010001000, 0100010001000100,
0010001000100010, 0001000100010001

corresponding to columns. We denote this code as C_s , which is a vector space of dimension 7, as 4 codewords corresponding to rows sum to the same value as 4 codewords corresponding to columns.

Similarly, when we look at a magic quad squares, the corresponding code C_m also must contain two codewords

1000010000100001 and 0001001001001000

corresponding to diagonals. The codewords span the vector space of dimension 8. We add two vectors to C_s , but we have one more dependency: rows 1 and 4 plus columns 2 and 3 plus both diagonals sum to 0.

Both codes contain the codeword consisting of all ones. This means there is a bijection between codewords of weights w and $16 - w$. Two codewords correspond to each other if their bitwise XOR equals all ones. That means the weight enumerator has to be a symmetric function. We wrote a program to calculate the weight enumerator for both codes.

The weight enumerator for C_s is

$$W_{C_s}(x, y) = x^{16} + 8x^{12}y^4 + 16x^{10}y^6 + 78x^8y^8 + 16x^6y^{10} + 8x^4y^{12} + y^{16}.$$

This weight enumerator can be calculated manually, too. For example, the codewords of weight 6 are in a bijection with codewords that can be formed as a bitwise XOR of two codewords that correspond to a row and a column of the square. Thus, there should be 16 of them.

The weight enumerator for C_m is

Table 7 Two extra quads in a magic square

	X	Y	
X			Y
Y			X
	Y	X	

Table 8 The number of codes in each dimension

Dimension	7	8	9	10	11
Semimagic	1	159	2531	2823	112
Magic		1	43	85	10
Strongly magic					1
Deck size	256	128	64	32	16

$$W_{C_m}(x, y) = x^{16} + 12x^{12}y^4 + 64x^{10}y^6 + 102x^8y^8 + 64x^6y^{10} + 12x^4y^{12} + y^{16}.$$

That means every magic quad square contains 12 quads: 4 rows, 4 columns, 2 diagonals, and 2 more. For the two extra quads, we observe that $1000010000100001 + 0100010001000100 + 0000000011110000 + 0001000100010001 + 1111000000000000 = 0010001100001000$. Thus, 0010001100001000 is a quad, and by symmetry, 0100100000010010 is too. We can visualize them in the square as broken diagonals in Table 7, where X marks cards for one quad and Y for the other.

We denote the code for strongly magic squares as C_{sm} . Its dimension is 11 with the weight enumerator

$$W_{C_{sm}}(x, y) = x^{16} + 140x^{12}y^4 + 448x^{10}y^6 + 870x^8y^8 + 448x^6y^{10} + 140x^4y^{12} + y^{16}.$$

The code C_{sm} is special: If we drop any particular bit from all the codewords, we will get a code of length 15 with Hamming distance 3 and dimension 11. That means the result is a perfect code. In particular, it is the Hamming(15,11) code.

Theorem 12 For each type of quad square (semimagic, magic, strongly magic), Table 8 shows how many corresponding codes are in each dimension and the smallest deck size that realizes it.

Proof From Theorem 3, we know that each quad code is realizable in some deck. Moreover, from Theorem 4, we know that the cards corresponding to a code are defined up to an affine transformation.

In [5], it was shown that in the EvenQuads-2ⁿ deck, there are

$$2^n(2^n - 1)(2^n - 2)(2^n - 4)(2^n - 8) \cdot \\ (112 + 2823(2^n - 16) + 2531(2^n - 16)(2^n - 32) \\ + 159(2^n - 16)(2^n - 32)(2^n - 64) + \\ (2^n - 16)(2^n - 32)(2^n - 64)(2^n - 128))$$

semimagic quad squares. The proof showed that this formula is a sum of the number of equivalency classes of sets of cards forming a magic square and spanning a particular deck size times the number of elements in such a class. From here, the coefficients in the sum above correspond to the number of quad codes defining semimagic squares in different dimensions.

Similar, in [5], we showed that in the EvenQuads- 2^n deck, there are

$$2^n(2^n - 1)(2^n - 2)(2^n - 4)(2^n - 8) \cdot \\ (10 + 85(2^n - 16) + 43(2^n - 16)(2^n - 32) \\ + (2^n - 16)(2^n - 32)(2^n - 64))$$

magic quad squares. The third row of the table is extracted from the formula in the same way as above.

For the fourth row, we use the theorem from [5] that shows that there are $2^n(2^n - 1)(2^n - 2)(2^n - 4)(2^n - 8)$ strongly magic quad squares that can be made by using the cards from the EvenQuads- 2^n deck.

From Theorem 3, we can find the smallest n , such that the deck size realizing a quad code of dimension k corresponding to one of the squares is 2^n . Given that the length of our codes is 16, such n has to be $16 - k - 1 = 15 - k$. \square

Conclusion and Future Directions

We introduced the EvenQuads game, a generalization of the famous SET game, and connected it to error-correcting codes, which we call quad codes, by associating the set of ℓ cards from the deck with linear binary codes of length ℓ and Hamming distance of 4. These connections allowed us to analyze properties of quad codes such as their dimensions, weight enumerators, and symmetries while extending these ideas to various deck sizes.

With the framework for error-correcting codes, we calculated the possible number of quads for a given number of up to 8 cards. In addition, we computed bounds for no-quads sets. We introduced error-correcting codes related to semimagic, magic, and strongly magic quad squares and calculated the smallest dimension for such codes as well as its corresponding the weight enumerator. We highlighted the mathematical link between magic quads squares and Hamming-(15,11) code.

There is room for further exploration in several directions that are out of scope of our paper.

- Improve the bounds calculated in the paper.
- Extend the notion of quad squares to the third and other dimensions.
- Analyze the effectiveness of suggested codes.
- Explore potential practical applications of these new error-correcting codes.

Acknowledgements We are grateful to the MIT PRIMES STEP program and its director, Slava Gerovitch, for allowing us the opportunity to do this research.

Funding Open Access funding provided by the MIT Libraries.

Data Availability Not applicable.

Declarations

Conflict of interest On behalf of all authors, the corresponding author states that there is no Conflict of interest.

Ethical Approval Moreover, Research Involving Human and/or Animals is not applicable.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Axler S. Linear algebra done right. Springer; 1997.
2. Berlekamp ER. The technology of error-correcting codes. Proc IEEE. 1980;68(5):564–93.
3. Brown TC, Buhler JP. Lines imply spaces in density Ramsey theory. J Combin Theory Ser A. 1984;36(2):214–20.
4. Byrapuram N, Choi H, Ge A, Ge S, Khovanova T, Lee SZ, Liang E, Mandal R, Oki A, Wu D, Yang M. Card games unveiled: exploring the underlying linear algebra; 2023. math.HO. [arXiv:2306.09280](https://arxiv.org/abs/2306.09280).
5. Byrapuram N, Choi H, Ge A, Ge S, Khovanova T, Lee SZ, Liang E, Mandal R, Oki A, Wu D, Yang M. QuadSquares; 2023. math.HO and math.CO. [arXiv:2308.07455](https://arxiv.org/abs/2308.07455).
6. Cary Huffman W, Pless V. Fundamentals of error-correcting codes. Cambridge University Press; 2010.
7. Chen E, Du W, Gupta T, Khovanova T, Li A, Mallajosyula S, Raghavan R, Sinha A, Smith M, Qian M, Wang S. The classification of magic SET squares. Recreat Math Mag. 2020;7(13):71–94.
8. Conway JH, Sloane NJA. Sphere packings, lattices and groups, vol. 290. Springer; 2013.

9. Crager J, Flores F, Goldberg TE, Rose LL, Rose-Levine D, Thornburgh D, Walker R. How many cards should you lay out in a game of EvenQuads? A study of 2-caps in $AG(2, n)$; 2022. math.CO. [arXiv:2212.05353](https://arxiv.org/abs/2212.05353).
10. Gordon G, Gordon H, Gordon R, McMahan L. The joy of SET: the many mathematical dimensions of a seemingly simple card game. Princeton University Press; 2017.
11. Gordon G, McMahan E. Error detection and correction using SET®. In: The mathematics of various entertaining subjects research in recreational mathematics, vol. 14; 2016. p. 199–12.
12. Hamming RW. Error detecting and error correcting codes. Bell Syst Tech J. 1950;29(2):147–60.
13. Rose LL. Quads: a SET-like game with a Twist. <http://sigmaa.maa.org/mcst/QUADS%20-%20SET%20WITH%20A%20TWIST.pdf>
14. OEIS Foundation Inc. The on-line encyclopedia of integer sequences; 2023. <https://oeis.org>.
15. Vinci J. The maximum number of sets for N cards and the total number of internal sets for all partitions of the deck; 2022. <https://www.setgame.com/sites/default/files/teacherscorner/SETPROOF.pdf>.

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