Irish Math. Soc. Bulletin
Number 95, Summer 2025, 3– 21
ISSN 0791-5578

Coding Theory: The story of how an engineering problem evolved into a branch of pure mathematics

STEVEN T. DOUGHERTY

Abstract. We describe how a very practical engineering problem involving the transmission of electronic information evolved into beautiful, interesting, pure mathematics with branches in algebra, combinatorics, and number theory.

1. Introduction

Throughout the development of mathematics, interesting questions from other disciplines have sparked new mathematics to emerge. For example, when Newton discovered calculus [27], he was doing so largely to answer questions about motion and gravity (in contrast to Leibniz's motivation, which was quite different [23]). However, no one doing real analysis today would think that what they were doing was completely in service to questions of motion or gravity. The mathematics of calculus developed on its own as pure mathematics after it was born to answer applied questions. Similarly, Newton's minimal resistance problem [27] gave rise to the calculus of variations, which sparked a great deal of mathematics. Modern physics has also given mathematicians plenty to study arising from topics like string theory and the theory of relativity. After all, where would differential equations and differential geometry be without a constant flow of problems from physics and engineering. Outside of analysis, Euler developed graph theory to answer the fairly easy recreational question about the bridges of Königsberg [10]. This simple problem gave rise to one of the most useful and interesting branches of discrete mathematics. Moreover, numerous applied problems have been solved using the techniques developed in graph theory. In a similar way, Euler developed the study of Latin squares from the much more difficult recreational problem of arranging 36 officers in a square [11] (the problem took over 100 years to solve after Euler started the approach). In this paper, we are going to describe another situation where a mathematical discipline arose from an engineering problem in the twentieth century to become a highly active and interesting area of pure mathematics. In fact, one could argue that several highly interesting pure branches of coding theory have emerged, such as algebraic geometry codes and codes over rings. In both of these areas, an existing branch of pure mathematics, specifically algebraic geometry and ring theory, was first used in the service of practical coding theory and then was highly enriched as pure mathematics from the techniques that were used in the application. In this paper, while we do talk about algebraic geometry, our main focus will be the development of the study of codes over rings and other alphabets as pure mathematical objects.

1.1. **Coding Theory is Born.** In the late 1940s, coding theory was born as a very practical solution to an engineering problem. At Bell Labs, when the information

2020 *Mathematics Subject Classification.* 01A60,11T71, 94B05 .
Key words and phrases. Coding Theory, Rings, Pure Mathematics.
Received on 28-02-2025; revised on 15/06/2025.
DOI: 10.33232/BIMS.0095.3.21.

age was in its infancy, scientists and engineers worked on practical problems for the monopolistic telephone giant. Basically, the idea arose that if the electronic device could realize that something was, in fact, an error, then the device should also be able to change the error so that with a very high probability the error could be changed to what it should be. In other words, if an error can be detected, it should be possible that the error can be corrected.

The first major step in this direction was made by Claude Shannon in 1948 in the paper "A mathematical theory of communication" [36]. It is impossible to overemphasize the profound effect this paper had on the emerging science of information theory. It is for this work and what follows that Shannon is often called the "Father of Information Theory". In this setting, communication generally means taking information, encoding it, passing it along a channel, decoding it, and receiving it. One can think of information being a message sent by an electronic device, which is encoded into a sequence of 0s and 1s. Then the information can be sent through a wire or over the airways, then it is decoded and received by another device. As an example, one might think of a telephone conversation. In this setting, noise may effect the message while it is in a channel. Certainly, anyone who has had telephone conversations has experienced a noisy channel. What Shannon showed was that effective communication can be conducted in virtually all situations. That is, no matter how noisy the channel is, one can still effectively communicate over it. However, he did not show exactly how this should be done, but rather that it was always possible. Thus began the long search for the mathematics that could make this happen.

2. Nomenclature and Early History

At this point, it might benefit the reader to give the definitions of the names of the disciplines (confusing as they are) that are being discussed. Information theory is the science of communicating, storing, and quantifying information. In general, this overarching term houses three different subjects. The first is coding theory, which studies how to communicate effectively, that is, how to correct errors with high probability that occur when transmitting data. The second is cryptology, which is the science of keeping information secret from bad actors as well as decrypting secret information made secret by someone else. Making messages secret is called cryptography with the obvious etymology of writing secrets and cryptanalysis is the branch which seeks to uncover what someone else has made secret. Here is where we meet our first difficulty in nomenclature, as we admit that often the term cryptography replaces the term cryptology to refer to the entire discipline. The third is called information theory, (which is our next difficulty in nomenclature). While information theory often refers to the entire subject, it also refers to the very practical engineering problem of the quantification of information and its transmission through a channel.

While the three topics are necessarily related and are often applied to the same information, the techniques used in these three disciplines are quite different. The mathematics behind coding theory is largely linear algebra with some help from combinatorics and abstract algebra. The mathematics behind modern cryptography is usually number theory (for example with the RSA crypto-system and the discrete log crypto-system) in non-symmetric key cryptosystems and finite field arithmetic, boolean functions, and other techniques from discrete mathematics in symmetric key cryptosystems. However, as we approach a post-quantum world, the techniques of cryptography are expanding quite rapidly to encompass numerous mathematical fields. The mathematics behind information theory is largely probability with some help from analysis. In actual transmission of information in our time, certainly, all three sciences are at work, as we desire to make the information correct, secure, and efficiently transmitted.

Shannon laid the foundation for all three of these disciplines. For coding theory there was the 1949 paper "Communication in the presence of noise" [37]; for information theory there was the 1948 paper "A mathematical theory of communication" [36]; and for cryptology there was the 1949 paper "Communication theory of secrecy systems" [38]. The three vitally important disciplines for the information age are based on these three seminal papers. We note that the first and third papers were published in the Bell System Technical Journal and the second was published in the proceedings of the IRE which has been renamed as the proceedings of the IEEE (Institute of Electrical and Electronics Engineers). It is quite clear that these are not mathematical journals, but rather very applied engineering journals. At that point, they were not trying to produce interesting mathematics, but rather using mathematics to solve very important and interesting engineering problems.

In this paper, we shall be concerned with the first of these sciences, which is coding theory. We shall show how it evolved from this application to become a branch of pure mathematics studied for its own sake rather than in the service of other applied disciplines.

Following Shannon's landmark paper [36] in 1948, there came two more important papers. The first by Marcel Golay [16], in 1949, and the second by Richard Hamming [20] in 1950. Within these papers, the Golay codes and the Hamming codes were described. To this day, they remain some of the most discussed codes in coding theory. Moreover, the list of connections between these two families of codes to other interesting mathematical objects seems to never stop growing. It has often been said that if you want to interest a mathematician in coding theory you show them the length 23 Golay code, its connections to the Leech lattice, and its connections to finite groups (see [6] for a description of all of these connections) and any mathematician will quickly become interested in the topic.

3. Techniques of Coding Theory

The techniques that arose from these early papers are essentially the techniques that are still used today. First of all, you define an alphabet A. This alphabet consists of the symbols that one can send. Quite often, in the electronic world, the alphabet is the binary set $A = \{0, 1\}$, but other alphabets are possible. One can even think of the English language as a code over the Latin alphabet. While every discussion of coding theory begins by assuming that A is any set, it very quickly becomes clear that A must have some algebraic structure to make the techniques of coding theory effective. In those days, and in most applications today, it was assumed that A must be a finite field. In practice, A is often the finite field $\mathbb{F}_2$. However, even in early papers, results were often couched in terms of an arbitrary field since the proofs were the same.

While one might assume that the field $\mathbb{F}_2$ is the only field used in applications, this is not correct. Even when the information is stored as a 0 or a 1, one can use the field $\mathbb{F}_{2^r}$. For example, one can use $\mathbb{F}_{2^8}$ and deal with bytes rather than bits of information and still have the information stored as binary. The next task is to define the length of the code, usually denoted by n. Then a code of length n is simply any subset of the space A^n. That is, a code is simply a collection of length n vectors with entries from the alphabet A. Essentially, the code consists of all the possible "words" that you can send over the channel. This means that the only things you want the receiver to read are the elements of the code. In this way, if the receiver reads something that is not in the code, an error must have occurred.

One can think of the parallel situation for the English language when you receive something that is not a word. If you are sent, in a message, a collection of letters that are not an English word, you first recognize that it is not a word, then you try to

determine what is the most likely word that was actually sent and had morphed into this collection of letters. This is the same situation as in coding theory. Therefore, the next step is to turn the ambient space, A^n, into a metric space which enables you to determine the distance between a received message and a possible word. This is done by defining the Hamming metric, which is $d_H(\mathbf{v}, \mathbf{w}) = |\{i \mid v_i \neq w_i\}|$ for vectors $\mathbf{v}, \mathbf{w} \in A^n$. It is clear at this point that if you want to define a code that is effective in correcting errors, you want the distance between vectors to be as large as possible. In particular, you want the minimal distance between two distinct vectors to be as large as possible. In that way, if some mistakes are made, it is clear what vector was originally sent.

In terms of the English language analogue, the distance is very small. For example, the words *can*, *cat*, and *car* only differ in one location. Therefore, if someone receives the word *cas*, without any context, it is impossible to know which of these three was the intended message. Mathematically, this is done in the following way. Define the minimum distance of a code C to be $d_C = \min\{d_H(\mathbf{v}, \mathbf{w}) \mid \mathbf{v}, \mathbf{w} \in C, \mathbf{v} \neq \mathbf{w}\}$. Then, using this code, you will be able to detect $d-1$ errors and can correct $\lfloor \frac{d-1}{2} \rfloor$ errors. In terms of the English example above, if you send *cat* and receive *car*, then you would not detect that an error was made since the minimum distance is only 1 in the English language. However, if your language consisted of two words 000 and 111 and you received 110 then you know that a mistake was made in transmission. You would want to correct the vector to 111 since that vector is distance 1 from the received vector, whereas 000 is distance 2 from the received vector. Of course, 2 mistakes might have been made in which case you correct the received vector to the wrong vector.

Already, this setting is sounding like a very mathematical problem. Namely, stated as a mathematical problem, what is the largest set of vectors $C \subseteq A^n$, where A is a set, such that the minimum distance between distinct vectors is d. One can rearrange this question by fixing any two of the parameters and optimizing the third. One can easily imagine this as placing ping pong balls in a room (in which they can float). You want to put as many in the room as possible but keep the distance between any two as far away as possible (two conflicting aims). Therefore, the question becomes how many pingpong balls can you put in the room where they are at least d units away.

3.1. **Bounds and Families of Codes.** Within this very simple combinatorial setting two very important results emerge. The first is the Singleton bound, which states that for a code C of length n over an alphabet of size q with minimum Hamming distance d, we have $\log_q(|C|) \leq n - d + 1$. The proof of this result is very easy and short, however the result has massive consequences. We call any code meeting this bound, that is, $\log_q(|C|) = n - d + 1$, a Maximal Distance Separable (MDS) code. These codes have been shown to be equivalent to many of the most important open questions in combinatorics. For example, a set of k-mutually orthogonal Latin squares is equivalent to certain MDS codes and as such the question of the existence of finite affine and projective planes can be couched in terms of MDS codes. Additionally, the central questions about orthogonal arrays can be phrased in terms of MDS codes. Moreover, there is a well known conjecture about these codes that arose in the study of finite geometry. Specifically, in 1955, Segre posed the MDS conjecture, which is as follows [35]. If $k \leq q$ then $n \leq q + 1$, unless $q = 2h$ and $k = 3$ or $k = q - 1$, in which case $n \leq q + 2$.

The second important result is the sphere packing bound, which states that for a code C of length n over an alphabet of size q with minimum Hamming distance at least $2t + 1$, we have $|C| \left(\sum_{s=0}^{t} C(n, s)(q-1)^s\right) \leq q^n$, where $C(n, s)$ is the binomial coefficient $\frac{n!}{s!(n-s)!}$. The proof of this result is a straightforward counting argument. Namely, $\sum_{s=0}^{t} C(n, s)(q-1)^s$ counts the number of vectors in a ball of radius t around

a codeword, so the number of vectors times the number of vectors in the non-intersecting balls must be less than the number of vectors in the ambient space. A code that meets this bound, that is a code C over an alphabet of size q with minimum Hamming weight $2t+1$ and $|C|(\sum_{s=0}^{t} C(n,s)(q-1)^s) = q^n$, is said to be a perfect code. Important open questions remain about perfect codes, but some of the most important perfect codes were found right at the very beginnings of coding theory. Specifically, they were found by Golay in 1949 [16] and Hamming in 1950 [20].

3.2. **Foundations.** We shall describe the underlying mathematical structures laid forth in these early papers, which remain the foundation for modern coding theory. A generating matrix G is given, which is a k by n matrix with elements from a finite field $\mathbb{F}_q$ whose rows are linearly independent. Then, to encode the vector $\mathbf{v} \in \mathbb{F}_q^k$ from the alphabet $\mathbb{F}_q$, one computes $\mathbf{v}G$ to produce a length n codeword. In general, one encodes $\mathbb{F}_q^k$ to obtain a vector subspace of dimension k in $\mathbb{F}_q^n$, which is known as a linear code of length n and dimension k and denoted as an $[n,k]_q$ code. If, in addition, the minimum Hamming distance is known, it is denoted as an $[n,k,d]_q$ code. The fundamental question of coding theory then becomes: "what is the largest dimension k for which a k-dimensional subspace of $\mathbb{F}_q^n$ exists with minimum Hamming distance d." For linear codes, the Hamming distance turns out to be equal to the minimum Hamming weight where the Hamming weight is the number of non-zero entries in a vector and the minimum Hamming weight is the smallest Hamming weight of any non-zero vector in the code. At this point, one can see that this is fundamentally a mathematical question, which naturally interested mathematicians.

To decode information, the following technique was made. The ambient space $\mathbb{F}_q^n$ is endowed with the standard inner-product $[\mathbf{v},\mathbf{w}] = \sum v_i w_i$ and the orthogonal code is defined as $C^\perp = \{\mathbf{w} \mid [\mathbf{v},\mathbf{w}] = 0, \forall \mathbf{v} \in C\}$. A parity check matrix for C is an $(n-k)\times n$ matrix of rank $n-k$ whose row space is $C^\perp$. It is immediate that $\mathbf{v} \in C$ if and only if $H\mathbf{v}^T = \mathbf{0}$. This setting put the mechanics of coding theory solidly in the field of classical linear algebra. For example, one immediately has that $\dim(C) + \dim(C^\perp) = n$ and that $C^\perp$ must be a linear code. Then, one finds each coset of the code in the ambient space and defines it as $\mathbf{e} + C$ where $\mathbf{e}$ is a vector of (hopefully) small Hamming weight, which serves as a possible error. If the vector $\mathbf{u}$ is received, one computes $H\mathbf{u}^T$ and finds $\mathbf{e}$ satisfying $H\mathbf{u}^T = H\mathbf{e}^T$. This means that $\mathbf{u}$ is in the coset $\mathbf{e} + C$ and one assumes that $\mathbf{e}$ is the error vector. Then one assumes the message that was sent was $\mathbf{u} - \mathbf{e}$. Note that $H(\mathbf{u}-\mathbf{e})^T = \mathbf{0}$ so $\mathbf{u} - \mathbf{e}$ must be in the code. One can see immediately that there are many computational difficulties in this scenario. A code that is useful is one that has a high minimum weight (and hence can correct a high number of erroneous entries of a received word) and where there is an efficient algorithm for decoding the received vectors.

Essentially, the code is a k-dimensional subspace of an n-dimensional space over a finite field. The received message is a vector in this n dimensional space and is decoded by finding the closest vector (with respect to the Hamming metric) in the k dimensional code and decoding to that vector. This is further complicated by the fact that there may not be a unique vector that is closest. Not only does this sound like a problem from pure mathematics, it sounds like a classical geometric problem. These types of problems have been studied in Euclidean space for centuries. This problem simply changes the metric and uses a finite field. For example, one can easily see analogues in terms of the sphere packing problem or in terms of uniform distributions.

The main work done in subsequent decades was to come up with codes that had efficient decoding algorithms. That is, a way of describing the code such that there is an efficient algorithm for finding the closest vector. One of the first was the celebrated

Hamming codes [20]. In these codes, the parity check matrix is constructed by using vector representatives of the distinct points of projective space over a finite field as the columns of the matrix. This means that the minimum distance is 3 and that the code meets the sphere packing bound, that is, they are all perfect codes. Then computing $H\mathbf{u}^T$ gives the exact coordinate where the error is made and what the error is. While these codes have many fascinating properties, they were not often used in applications because the minimum distance is small. However, they have been studied extensively in terms of their connections to combinatorics and group theory. For example, the $[7, 4, 3]$ binary code is intimately related in every imaginable way to the projective plane of order 2, known as the Fano plane.

At about the same time, Golay came up with two other perfect codes, namely a binary $[23, 12, 7]$ code and a ternary $[11, 6, 5]$ code. These codes have been intensely studied and they have fascinating connections to groups, lattices, and designs. For example, the binary Golay code is intimately related to the Leech lattice and it produces 5-designs by its vectors of a given weight. Moreover, its automorphism group is the Mathieu group M23. See [6] for a description of these results.

4. Coding Theory Comes to Life

Within the next decade two families of codes were found that were actually highly useful in practice, namely the Reed-Muller codes [32] and the Reed-Solomon codes [33]. These codes were not only shown to have good parameters but efficient decoding techniques were also produced. Both of these codes rely on very clever applications of abstract algebra. The Reed-Muller codes have a wide array of uses in information transfer and have very interesting connections to finite geometry, see [1] for a detailed description of this connection. The Reed-Solomon codes are extensively used in such technologies as compact discs, digital video discs, blue ray discs, and QR codes. Both families of codes have been used in NASA space probes when sending digital information back to earth.

What should be understood about this period in the history of coding theory is that mathematicians and engineers were using classical results from combinatorics, linear algebra, and abstract algebra to produce codes and algorithms associated with these codes. At this early stage it was very much applied mathematics, namely the results of mathematics were used (along with some very clever innovations) to solve practical engineering problems. Codes were not yet being used to answer questions in mathematics (at least not extensively).

As an example of codes being constructed at this time, consider cyclic codes, which were first introduced by Eugene A. Prange in 1957, see [28], [29], [30], and [31]. These papers, despite their importance are exceedingly difficult to find, and are generally not even in the usual mathematical databases. They were technical reports for a United States Air Force project. A cyclic code is a code C such that if $(c_0, c_1, \ldots, c_{n-1}) \in C$ then $(c_{n-1}, c_0, c_1, \ldots, c_{n-2}) \in C$. This family of codes is one of the most well studied families of codes, not only because they are useful but because they have a canonical algebraic description. Specifically, a linear cyclic code corresponds to an ideal in the ring $\mathbb{F}_q[x]/\langle x^n - 1\rangle$, where the vector $(c_0, c_1, \ldots, c_{n-1})$ corresponds to the polynomial $c_0 + c_1x + c_2x^2 + \cdots + c_{n-1}x^{n-1}$. Notice that multiplication by x corresponds to the cyclic shift.

It follows that a complete classification of cyclic codes can be found by factoring the polynomial $x^n - 1$ over the field $\mathbb{F}_q$. If $p(x)$ is a factor, then the ideal $\langle p(x)\rangle$ corresponds to a cyclic code of length n over $\mathbb{F}_q$. One of the reasons that cyclic codes are so widely studied is that polynomial rings are such a well studied object in abstract algebra. In that sense, the machinery necessary to study these codes was already firmly in place

in the world of pure mathematics. All that was required was to apply this extensive machinery in the setting of algebraic coding theory. This does not mean that interesting and difficult questions did not arise from this application; they certainly did. But the large machinery of algebra was a great help in building the theory of cyclic codes. For example, finding the ideals when the length was not relatively prime to the characteristic of the field is a difficult algebraic problem.

The ideas used to study cyclic codes have been widely generalized from 1957 to the present day. For example, ideals in $\mathbb{F}_q[x]/\langle x^n+1\rangle$ correspond to codes with the property that

$$(c_0, c_1, \ldots, c_{n-1}) \in C$$

implies

$$(-c_{n-1}, c_0, c_1, \ldots, c_{n-2}) \in C$$

and are known as negacyclic codes. Ideals in $\mathbb{F}_q[x]/\langle x^n - \lambda\rangle$ correspond to codes with the property that

$$(c_0, c_1, \ldots, c_{n-1}) \in C$$

implies

$$(\lambda c_{n-1}, c_0, c_1, \ldots, c_{n-2}) \in C$$

and are known as constacyclic codes. Additionally, codes that are ideals in $\mathbb{F}_q[x]/\langle p(x)\rangle$ are known as polycyclic codes. Denoting the cyclic shift by σ, that is

$$\sigma((c_0, c_1, \ldots, c_{n-1})) = (c_{n-1}, c_0, c_1, \ldots, c_{n-2}),$$

then a code that is held invariant by the action of σ^k is known as a quasi-cyclic code of index k.

Finally, one of the more recent generalizations uses an automorphism θ of the alphabet A. If a linear code has the property that if $(c_0, c_1, ..., c_{n-1})$ is in C then $(\theta(c_{n-1}), \theta(c_0), ..., \theta(c_{n-2}))$ is in C, we say that C is a skew-cyclic code. Numerous references for these codes can be found by looking at one of the standard references for coding theory such as [21], [26], or [34].

4.1. **The MacWilliams Relations.** In the early 1960s, Jessie MacWilliams proved two results that were foundational for the study of coding theory, see [24], [25]. The first theorem states that every linear isometry between linear codes for the Hamming distance can be extended to a linear isometry of the ambient space. The second involves weight enumerators and are known as the MacWilliams relations. We shall describe weight enumerators now. Let C be a code of length n over an alphabet A, then the Hamming weight enumerator of C is defined as $W_C(x, y) = \sum_{\mathbf{c}\in C} x^{n-wt_H(\mathbf{c})} y^{wt_H(\mathbf{c})}$, where $wt_H(\mathbf{c})$ indicates the Hamming weight of the vector $\mathbf{c}$. The weight enumerator is expressed as $W_C(x, y) = \sum A_i x^{n-i} y^i$ where there are A_i vectors of Hamming weight i in C. The MacWilliams relations show that the weight enumerator of the orthogonal code can be determined from the weight enumerator of the code. Specifically, if C is a linear code over the finite field $\mathbb{F}_q$, then $W_{C^\perp}(x, y) = \frac{1}{|C|} W_C(x + (q-1)y, x - y)$. One cannot overestimate the many uses this theorem has in coding theory. For example, if C is a self-dual code, that is $C = C^\perp$, then the weight enumerator is held invariant by the action of the MacWilliams relations and therefore invariant theory can be used to determine the set of all possible weight enumerators for self-dual codes. This result is known as Gleason's Theorem. This would be used extensively in the following years in the application of coding theory to design theory and to lattice theory (see [1],[2],[6],[26]). The MacWilliams relations were one of the earliest and most beautiful theorems in coding theory that were both useful in practical applications and stood alone as beautiful theorems of mathematics. This was a case of new mathematics being developed in coding theory rather than existing mathematics being applied to a problem.

By this I mean that even if there were no applications of codes to the problem of electronic communication, this was still a particularly beautiful and interesting theorem.

The author of these theorems had a particularly interesting story as well. She had studied in England before raising her children and many years later received her Ph.D. from Harvard University. More will be mentioned about her story and accomplishments later in the article.

5. End of the Beginning

In 1970, N. Levinson published "Coding Theory: A Counterexample to G. H. Hardy's Conception of Applied Mathematics" [22]. In this article, Levinson puts forth the idea that Hardy's well known ideas on the distinction between pure and applied mathematics were contradicted by number theory's use in actual applied coding theory. Hardy saw pure mathematics (in his eyes the superior) as useless and should revel in its uselessness. At the apex of this useless but beautiful mathematics was number theory; it was the queen of mathematics and was (at least at that time) considered completely useless in the outside world. What Levinson shows is that results that Hardy saw as useless but wonderful, were used extensively in the design of codes and decoding algorithms. The purest of the pure of mathematics was now used with a very practical purpose. (As an aside, Hardy believed that one of number theory's greatest attributes was that it had no military applications. How disappointed he would be to find out that in the twenty-first century the world's security services were absolutely filled with number theorists working on cryptographic applications.)

At this point, coding theory was solidly an applied mathematics endeavor. Researchers were using known classical mathematics to solve interesting and important questions in electrical engineering and computer science. That is not to say that there were not first-rate mathematicians and first-rate mathematical theorems emerging, but the main focus was solidly on applications.

In 1971, at a workshop on coding theory, Ned Weldon stated: "Too many equations had been generated with too few consequences... Coding theorist professors had begotten more coding theory Ph.D.s in their own image... no one else cared; it was time to see this perversion for what it was. Give up this fantasy and take up a useful occupation... Coding is dead." [42] The author of the present paper was told by people who were present that this conference was known as the "coding theory is dead" conference. The idea behind this comment was that technology was far behind what they could do in theory. Another concern for those present was that mathematicians were getting interested in the central problem of coding theory, which meant that someone was going to give a complete solution to the fundamental problems of the discipline leaving no room for further research. That is, he thought that all of the work being done by coding theorists was not being applied and served no useful purpose. This did not prove to be the case! There were two enormous reasons for this. The first is that the mathematical part of the theory began to explode and find a myriad of interesting paths to take including numerous connections to combinatorics, finite geometry, and number theory. The second was that over the next three decades electronic devices of all kinds were to expand the uses of coding theory greatly. Few in 1970 would have guessed that by the year 2000 most people would have a computer on their desk at work as well as one in their house, nor would they have guessed that cellphones would be carried by virtually every person on the planet (the author of this article still does not have one, but recognizes their universality). This explosion of electronic communication would find many new and interesting applications for coding theory as well as for information theory as a whole. For example, before the dawn of the information age, cryptography was largely the concern of the military, the government, and large

financial organizations. In 2024, cryptography is the concern of every person on the planet as it allows for internet commerce. One can hardly overstate the importance of keeping personal and financial information secure in the present day.

In 1973, Delsarte published a paper framing coding theory in very pure terms [7]. In this paper, he studied codes in terms of association schemes. This paper has been referenced numerous times and used to produce interesting results about codes in this setting. The paper was the author's thesis at the Université Catholique de Louvain. In this paper, the author describes an association scheme approach to coding theory. One scheme, the Hamming scheme, is defined using a finite set F with cardinality at least 2, and in the space F^n. Then two points x and y are said to be i-th related if the Hamming distance between them is i. One of the important results in this paper was determining the possible group structures in this Hamming scheme. This result would later be understood more completely in the 1990s leading to an explosion in the study of codes over rings. This paper will be discussed later in detail. From our perspective, the importance of this paper is that the author was examining coding theory as a well described mathematical structure. In essence, coding theory was being considered as pure mathematics with solid foundations and viewing the fundamental objects of coding theory in their own right rather than in service to an application. Just two years after Weldon had incorrectly announced the death of coding theory, Delsarte was putting the mathematical discipline of coding theory on a sound foundation within the field of combinatorics.

6. Coding Theory Emerges as a Full Discipline

One of the biggest events in coding theory, just six years after the pronouncement of its death knell, was the publication of the text "The theory of error-correcting codes" by Jessie MacWilliams and Neil Sloane in 1977. This text became the standard reference for algebraic coding theory for at least the next 30 years. The text was steeped both in the engineering applications and the mathematics that underpinned them. One can hardly overestimate the influence this book had on the subject. It was comprehensive and well written and filled with avenues of future research. As a testament to its importance, checking the standard mathematical databases, the text has been cited thousands of times. The major importance of this work was that it brought the known world of coding theory into one place and set a course for the future of coding theory. It also unified notation and terminology for the discipline and served as the standard reference for works in coding theory for the decades that followed.

In the late 1960s and the 1970s, coding theory began to give results in mathematics, as opposed to being used primarily in applications. For example, in 1969 [2], Assmus and Mattson proved a now celebrated theorem that gave a construction for new 5-designs from extremal self-dual codes. Codes such as the extended Golay code of length 24 could be used to construct 5-designs from the codewords of a given weight. Rather than combinatorics giving constructions for codes, codes were now giving constructions of difficult to produce combinatorial objects. In this way, the results of coding theory were now being used to inform questions in pure mathematics, specifically in the field of combinatorics. There were, in fact, numerous connections between combinatorics and coding theory; see [3], published in 1974, for a taste of canonical connections between the two worlds.

During this time there was an active research group in Cambridge Massachusetts centered around Andrew Gleason (a Harvard professor who had solved Hilbert's fifth problem, had done significant work in breaking enemy codes during the second world war, and has two theorems in coding theory named for him, namely the Gleason-Prange Theorem and Gleason's Theorem, which was mentioned earlier), with members

Ed Assmus, Gene Prange, Vera Pless, John N. Pierce, and Skip Mattson, with frequent visits from Elwyn Berlekamp, Jessie MacWilliams, and Neil Sloane. This group was funded by the United States Air Force and was pivotal in making a wide group of mathematicians interested in coding theory. Moreover, this group began to bring the full force of mathematics to questions in coding theory. Some of the most foundational and beautiful theorems in coding theory arose during this time as mathematicians began building a solid mathematical foundation for coding theory, viewing it as a newly born branch of mathematics.

Unfortunately, this group was disbanded when congress passed new legislation directing that the armed services could only fund research that was directly applicable to practical applications. One of the most interesting things that emerged from this group at this time was a presentation in 1970 given by Ed Assmus on the projective plane of order 10, which was a difficult open problem in finite geometry that was of great interest to many combinatorialists and geometers. This would lead to a major event in coding theory and combinatorics beginning with the paper we describe next.

In 1973, "On the existence of a projective plane of order 10", was published by MacWilliams, Sloane, and Thompson [13]. To give an idea of the level of mathematician who was being attracted to coding theory at this time, we shall give a few of the accomplishments of these authors. Sloane was was the winner of a Lester R. Ford Award in 1978, the Chauvenet Prize in 1979, IEEE Richard W. Hamming Medal in 2005, the Mathematical Association of America David P. Robbins Prize in 2008, and the George Pólya Award in 2013. Despite all of these accomplishments, he is perhaps best known for his creation On-Line Encyclopedia of Integer Sequences. Thompson was awarded the Fields Medal in 1970, the Wolf Prize in 1992, and the Abel Prize in 2008 and proved numerous major results in abstract algebra.

Jessie MacWilliams (mentioned earlier in the article) had a major impact on coding theory and proved one of the foundational results. She received a B.A. from Cambridge in 1938 and an M.A. in 1939 and then moved to the U.S. She raised a family there and later worked at Bell Labs. She completed her Ph.D. in 1962 at Harvard (under the direction of Gleason whose pivotal role was mentioned earlier) where she discovered the MacWilliams relations mentioned earlier. She was also the first Noether Lecturer, and as such gave a talk entitled "A Survey of Coding Theory" in 1980. The accomplishments of these three mathematicians are highlighted just to give an idea of the high quality mathematicians who were being drawn to the discipline at this time. One could also describe the accomplishments of many of the mathematicians that have been cited earlier. The point is that coding theory was being taken seriously by highly able mathematicians from first-rate institutions.

This paper on the projective plane of order 10 was the start of one of the most famous results in which coding theory was able to play a central role. Essentially, one takes a putative projective plane of order 10 and constructs an incidence matrix from its lines and its points. This matrix is used to generate a binary code, which can be extended to a self-dual ($C = C^{\perp}$) $[112, 56, 12]$ code. Using the MacWilliams relations and Gleason's theorem, which is based on this and gives the possible weight enumerators for a self-dual code using invariant theory, the weight enumerator of this putative code can be constructed after determining a few of its parameters with combinatorial (often difficult) arguments.

The existence of the projective plane of order 10 was the first unknown case for finite planes and the problem was often thought to be too difficult for young mathematicians to attempt. In fact, Clement Lam was warned off the problem by his Ph.D. advisor Herbert Ryser as a student at Caltech. Later, after Lam established himself as an accomplished researcher, he took to the problem with full force in the 1980s. He took

the following approach. If the weight enumerator of a putative self-dual code formed from the projective plane of order 10 was known, if one could prove the non-existence of such a code, then the plane would not exist. Lam and his team did significant work in reducing what needed to be done to prove its non-existence, such as showing the code had a trivial automorphism group. Following extensive theoretical work they ran a program on a supercomputer for about one year, which showed that no such code existed and hence the non-existence of the projective plane of order 10 was proven. By doing so, they used the machinery developed in coding theory to produce a significant result in pure mathematics. In essence, this result was an example of coding theory giving results in combinatorics rather than combinatorics providing a framework for applicable results in coding theory. This result, together with the proof of the four color theorem, sparked a debate in the mathematical community about whether a computer proof did, in fact, constitute a proof of a theorem.

In a conversation with the author of this article, Lam offered one possible avenue to take in this regard. Namely, we could take a cue from the physical sciences and only accept a computer proof after an independent team verified the computation. Given the amount of work to do this, it seems unlikely that anyone will try to do this for Lam's proof until they have found something that would streamline the computation.

7. Algebraic-Geometry Brings New Ideas

A major advancement that occurred in the 1980s and 1990s was the use of algebraic geometry in coding theory. It has been widely considered that the Varshamov-Gilbert codes were considered best possible codes in terms of optimization. In the early 1980s, V. D. Goppa discovered a broad class of codes arising from algebraic curves over finite fields, see [17], [18]. In 1983, Tsfasman, Vlădut and others showed the existence of Goppa codes better than Varshamov-Gilbert codes for alphabet size $p^2 \geq 49$, where p is a prime in certain cases, see [40] (whose authors received the 1983 Information Theory Society Award).

This was another example of the application of pure mathematics to produce interesting results in coding theory. In this case, it was the gigantic machinery of algebraic geometry. Algebraic geometry certainly had very pure origins in terms of Hardy's definition of pure mathematics. Moreover, the machinery developed in this discipline was absolutely enormous. Mathematicians, who had been applying these ideas to pure questions in number theory, pivoted and began applying these results to questions in coding theory. Of course, Wiles' phenomenal proof of Fermat's Last Theorem sent algebraic geometers in search of a new holy grail. A very large body of results soon followed. While, in overarching philosophy, this was still another branch of mathematics informing questions in coding theory, there were still a vast number of results in algebraic geometry codes that were produced that would certainly be considered pure mathematics in the sense that they were produced for coding theory (not algebraic geometry) but had no real application in the world of electronic communication.

It is interesting to note that these early papers were published in the Soviet Union, in a time when there was still limited exchange of ideas and results between countries in the Soviet block and countries outside of that block, especially in disciplines such as information theory where knowledge of the results could give a militaristic or technological advantage. The opening up of this block and the subsequent fall of the Soviet Union certainly helped spread the use of algebraic geometry to study codes. Perhaps the greatest impact was that numerous mathematicians from this block emigrated to countries in western Europe and North America (or at least visited them for long periods of time) and brought these ideas with them. Certainly, this explosion of the study of coding theory from the viewpoint of algebraic geometry was a very important move

in the direction to pure mathematics. For a complete description of this connection and an encyclopedic collection of the results see the book by Tsfasman and Vlădut "Algebraic-geometric codes" [39].

8. Rings Spark a Revolution

The next major move toward pure mathematics was the paper "The $\mathbb{Z}_4$-linearity of Kerdock, Preparata, Goethals, and related codes" [12], which was published in 1994. This paper showed that certain families of binary non-linear codes that nonetheless behaved very much like linear codes are, in fact, images of linear codes over the finite ring $\mathbb{Z}_4$ under the Gray map. That is, they were the images of submodules of $\mathbb{Z}_4^n$ mapped to $\mathbb{F}_2^{2n}$ with the proper map. The Gray map is a non-linear map, which is defined as follows: $0 \to 00, 1 \to 01, 2 \to 11, 3 \to 10$. Note that this map is not the one most mathematicians would think of when mapping $\mathbb{Z}_4$ to $\mathbb{F}_2^2$. That is, usually one would think that 2 would map to 10 and 3 would map to 11. However, this Gray map was the key to showing that these binary codes, which were not linear, were actually simply the image of the quaternary linear codes under this map. This is why these codes behaved like linear codes. That is, their group structure came from their being submodules of $\mathbb{Z}_4^n$ and explained why they seemingly obeyed the MacWilliams relations even though they were not linear codes. It should be noted that an understanding of Delsarte's 1973 paper really should have lead the coding theory community to these results much earlier! While these results were very important in their own right, their impact on the study of codes was even greater. It sparked a massive rush to study codes over finite rings.

There were some previous papers about codes over rings, but it was this paper that really sparked interest in this study. These early papers largely showed some things that were used to study codes over finite fields could also be extended to finite rings. They did not really have an application, either practical or theoretical, that sparked people's interest to continue their study. At first, most of the papers justified the study of codes over rings with an application to binary codes, which would tie the results closely to the applications of coding theory. For example, the first rings to be studied were the commutative rings of order 4. Codes over the finite field $\mathbb{F}_4$ had already been studied and codes over $\mathbb{Z}_4$ were widely studied. Then, numerous papers were written about codes over the ring $\mathbb{F}_2[u]/\langle u^2\rangle$ and its associated linear Gray map. Finally, codes over the ring $\mathbb{F}_2[v]/\langle v^2+v\rangle$, which is isomorphic as a ring to $\mathbb{Z}_2 \times \mathbb{Z}_2$, were studied. Codes over these rings were shown to have a canonical connection to lattices in certain spaces. Additionally, motived by the results in Delsarte's 1973 paper, codes were studied over mixed alphabets such as $\mathbb{Z}_4\mathbb{Z}_2$ codes. These were codes that combined the Gray map from the quaternary ring with the identity map from the binary field to construct binary codes. Many interesting and important results were obtained from this viewpoint, see [4] for a complete description of this work.

Next, generalizations of these four rings were studied. For example, some studied codes over the finite fields $\mathbb{F}_{2^r}$, the rings $\mathbb{Z}_{2k}$, $R_k = \mathbb{F}_2[u_1, u_2, \ldots, u_k]/\langle u_i^2, u_iu_j + u_ju_i\rangle$, and $A_k = \mathbb{F}_2[u_1, u_2, \ldots, u_k]/\langle v_i^2 + v_i, v_iv_j + v_jv_i\rangle$. These families of rings are also equipped with corresponding Gray maps to the binary space and are just some of the natural generalizations that were studied very quickly after codes over the finite ring $\mathbb{Z}_4$. More exhaustively, codes were studied over chain rings (rings whose ideals form a chain), local rings (rings with a unique maximal ideal), and then to principal ideal rings. However, the study of these rings was carried out in an essentially ad hoc manner, in that foundational results such as the MacWilliams relations were produced for various families of rings as needed. Foundational results for codes over rings in general were not yet well studied or clearly stated. However, numerous fascinating results

for codes over rings were found such as connections to unimodular lattices (see [6] and the massive number of reference therein) and interesting connections to binary codes via Gray maps. While numerous interesting results were found during this time, little thought was given to studying codes over a very broad family of rings. Rather, it was particular applications that fuelled the research.

Throughout the 1990s numerous papers were written on codes over various families of rings and numerous connections were found to number theoretic structures, but still quite often these papers were motivated (at least they were stated as such, if not in reality) by applications in the field of electronic communication. Additionally, old ideas were resurrected because of advances in computing. For example, in 1962 Gallagher wrote a paper [15] on Low Density Parity Check (LDPC) codes. Essentially, these were codes whose parity check matrix had few coordinates containing a non-zero element. Very little was done with these codes in the ensuing years.

In the 1990s and the decades that followed, an explosion of results on these codes appeared. Researchers noticed that these codes could use probabilistic methods, such as belief propagation, for decoding. This gave decoding algorithms of complexity linear in the length of the code. If every coding theorist had a very good computer on their desk in the 1960s, then this would have been noticed decades earlier. By the time of the writing of this article hundreds of papers have been written on this topic. Many techniques from graph theory, algebra, and finite geometry were used to construct these codes and began what is often referred to as modern coding theory to distinguish it from classical coding theory.

9. The Move to Purity

By the 1990s, mathematicians were studying codes in their own right for purely mathematical purposes, but this was still often couched in terms of possible applications, either in engineering or in other branches of mathematics such as combinatorics or number theory. This began to change significantly in this decade. Jay Wood spent the 1989 - 1990 academic year at Lehigh University (during this time he taught the author of this paper algebraic topology) and had numerous productive conversations with E. F. Assmus on coding theory. Recall that Assmus had been part of Gleason's team decades earlier. Assmus certainly viewed coding theory as pure mathematics (at least the part where his interests were) and was largely concerned with studying codes in relation to combinatorics and finite geometry. It was at this time that he wrote a text with Jenny Key on the relationship between codes and designs [1]. Later in the decade, Vera Pless suggested to Wood that it was time to reconsider the MacWilliams Theorems, given the new directions in coding theory. These conversations prompted Wood to write the paper "Duality for modules over finite rings and applications to coding theory" [41], published in 1999. In this paper, he showed that the largest class of rings for which both MacWilliams theorems held was the class of Frobenius rings. It should be noted that, if you were to ask one hundred good coding theorists at the time for a definition of a Frobenius ring, you probably would have received zero correct answers. Asking the same question today to the same group, there would be a marked increase in the number who give the correct answer.

Specifically, he showed the following results.

- If R is a finite Frobenius ring and C is a linear code, then every Hamming isometry $C \to R^n$ can be extended to a monomial transformation.
- If a finite commutative ring R satisfies that all of its Hamming isometries between linear codes allow for monomial extensions, then R is a Frobenius ring.

By an example of Greferath and Schmidt, [19], it was shown that these results do not extend to quasi-Frobenius rings. Therefore, Frobenius rings are the largest class of rings for which the first MacWilliams Theorem hold.

The next result generalizes the MacWilliams relations which we will state in a very general form, that is for the complete weight enumerator rather than just the Hamming weight enumerator. We require some definitions to state it. Let R be a finite ring with r elements and define $\iota : R \to \{0, 1, 2, 3 \ldots, r-1\}$, where the elements of R are $a_0, a_1, \ldots, a_{r-1}$ and $\iota(a_i) = i$. Define the complete weight enumerator of a code C over R as

$$cwe_C(x_0, x_1, \ldots, x_{r-1}) = \sum_{\mathbf{c} \in C} \prod_{i=1}^{n} x_{\iota(c_i)}.$$

Then, if $T_{a,b} = \chi(ab)$ where χ is a generating character for $\widehat{R}$, for a linear code C over a finite commutative Frobenius ring R, we have

$$cwe_{C^\perp}(x_0, x_1, \ldots, x_{r-1}) = \frac{1}{|C|} cwe_C(T \cdot (x_0, x_1, \ldots, x_{r-1})).$$

As a corollary, we get the MacWilliams relations for the Hamming weight enumerator.

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

What these results showed was that many of the foundational techniques of codes over finite fields could be applied in the case where the alphabet is a finite Frobenius ring. For example, when the ring is commutative and the code C is linear, we have $|C||C^\perp| = |R|^n$ and $(C^\perp)^\perp = C$. When the ring is non-commutative you must define orthogonals on the left and the right, then similar results are true depending on whether the code is left linear or right linear.

This paper then sparked an intense interest in codes over Frobenius rings and a flood of papers followed investigating what possible results could be obtained in this new path.

9.1. **Questions and Debates.** At the AMS Special Topic Session at Notre Dame University in April, 2000, a talk was given on codes over Frobenius rings (the author does not recall who was giving the talk). Vera Pless, who was a very important coding theorist (also a member of Gleason's team described earlier), casually raised her hand at the end of the talk and asked "Why should we care about codes over Frobenius rings anyway?" Her question was seeking to find motivation for the study of codes over Frobenius rings. Namely, was there a reason for going down this path. It must be noted that by this time Wood's paper was not widely known. While the author does not remember the answer given, it is certain that Vera Pless was far from satisfied with it. Pless had voiced a question that many were considering at the time. Namely, is it interesting to make such broad general definitions – are we still actually doing coding theory or are we doing generalizations for the sake of generalization?

Years later, at the AMS special session honoring the retirement of Than Ward (another mathematician who wrote numerous important papers in coding theory) at DePaul University, Chicago, October 6, 2007, the author of this article gave a talk about codes over rings with Vera Pless in the first row. The talk began recalling the Pless quote from seven years earlier. Pless then responded, loud enough for everyone to hear, "Why does everyone remember everything I say?"

The talk continued giving reasons for why we should be studying codes over Frobenius rings. The first reason is that we have the two foundational MacWilliams Theorems. There are numerous implications of these theorems in the classical case. It becomes natural to investigate what these implications are in the more general case. The second

reason is that combinatorial bounds such as the Singleton and Sphere-packing bounds mentioned earlier still apply in this case. Moreover, there are analogues for the algebraic bounds based on the types of rings. Additionally, some of the connections to other branches of mathematics were even better than in the classical case. For example, self-dual codes over rings have a much more natural connection to unimodular lattices than codes over fields do; there is no binary code that can produce the extremal unimodular lattice of length 72, but there is a self-dual code over $\mathbb{Z}_4$ that can produce it. Moreover, infinitely many lattices can be produced by self-dual codes over $\mathbb{Z}_{2k}$ that cannot be produced from codes over fields. Moreover, connections to lattices over various other infinite fields were found from codes over rings. Additionally, studying codes over Frobenius rings opened up a flood gate of applications to other algebraic questions. Finally, the results that were being obtained were, in fact, beautiful mathematics. They were natural, interesting, and highly non-trivial. At the end of the talk, the moderator asked for questions, then he said that he had a question but not for the speaker, but rather for Vera Pless. He asked her if she was now convinced that we should be studying codes over Frobenius rings and she wholeheartedly agreed that we should!

Essentially, if we have the foundational results in such a broad setting, it is quite natural to study these objects to determine what can be said about them. What has happened in these past 25 years is that mathematicians began to study codes as an interesting mathematical object in themselves. Like any objects in pure mathematics, practical applications may be found later for these results, but the drive to study these objects no longer came solely from a particular application.

As Alexander Barg said at an AMS conference at the University of Cincinnati in October 2006: "We do not have to pretend that what we are doing has anything to do with information transfer any more." At this point, codes are algebraic objects just like groups, rings, and fields and should be studied for their own sake whether or not they had an application in engineering or another branch of mathematics. In 2017 the text "Algebraic Coding Theory over Finite Commutative Rings" was published by the author of this paper. This text gives many of the foundational results of codes over rings. The author in no way would compare this book to the other major works described in this article, but mentions it because it is a high level book about coding theory that does not mention any applications at all. The book treats coding theory as a branch of algebra and gives no motivation from any applications. By this time, this approach was perfectly natural.

Within the time between these conferences and the present day, there has been a flood of research done on codes over rings and their applications in algebra and combinatorics. There have been so many written that some journals have had to restrict the number of coding theory papers that they are willing to accept for fear of being overwhelmed by them and losing their original desired focus whether algebraic or combinatorial. Likewise, engineering journals have put restrictions on coding theory papers as well, demanding that the papers they publish should have an immediate application that was the focus of the paper. Their fear was that their engineering journal was quickly becoming a journal of pure mathematics.

10. New Avenues from New Applications

During this time, coding theory was given another avenue of pursuit via an important engineering application. As physicists were rushing headlong into the quantum world, questions arose as to how errors in quantum communication could be detected. In this application, linear codes were not the codes that were interesting. Rather, it was additive codes (codes closed under addition but possibly not under scalar multiplication)

that were useful and instead of the standard Euclidean inner-product, different inner-products were used. See [5] for a complete description. When studying additive codes, it was also shown that a variety of inner-products could be used coming from the character group of the underlying additive group. In this way, one can think of a code over a group rather than over a ring. This idea sparked a great deal of research in loosening the algebraic conditions of the alphabet. Namely, a great deal of research was put forth looking at codes that were additive over some group (possible the additive group of a field or a ring). It can be shown that MacWilliams relations hold in this case in a very broad sense as well; see [9] for a complete description of the MacWilliams relations in this case.

Another new idea came to coding theory around this time as well, namely DNA codes. It is well known, and incredibly important scientifically especially in biology, that DNA contains a genetic program for the development of life. It consists of two strands, which are linked by the Watson-Crick pairing. Essentially, DNA is a sequence of information with an alphabet of A,C,G, and T. As such, it is interesting to study this information from the point of view of coding theory, namely as a code over an alphabet of size 4.

Unlike the quantum case, there does not seem to be a canonical connection to one of the rings of order 4, but rather any ring of order 4 can give results in this setting. Interestingly, at this point, it is not uncommon to hear geneticists talking about Hamming distance when they are talking about DNA.

From the point of view of this article, the interesting thing here is that scientists took coding theory as a body of pure mathematics, and then applied it to a scientific situation. In essence, this is what is at the heart of mathematics, scientists can take an already developed, abstract, body of results and apply them to questions in their own discipline.

As a branch of pure mathematics, coding theory can now be viewed in the following way. A code is a subset of A^n where A is a set with an associated metric. In this setting, the questions of coding theory are largely combinatorial (for example the connection between MDS codes and mutually orthogonal Latin squares). We then consider A to have some algebraic structure. In its most general form we assume A is a finite commutative group and linear codes are subgroups of the ambient space. Attached to the ambient space is a function that acts like an inner-product, which defines an orthogonal code. The MacWilliams relations hold for weight enumerators in this setting and we have the double annihilator condition $(C^{\perp})^{\perp} = C$ and the cardinality condition that $|C||C^{\perp}| = |A|^n$.

Restricting the alphabet, we can make A a finite Frobenius ring (either commutative or non-commutative). Then, with an associated inner-product, the MacWilliams relations, the double-annihilator condition, and the cardinality condition hold. Within this framework, we have an algebraic structure, a metric, and a notion of duality. The essential question then becomes what is the largest code one can construct for a given length with a given minimum distance under the metric. This is certainly a question of pure mathematics and can be stated in a very general abstract manner. Given this setting, various types of codes can be studied, for example codes equal to their duals, codes contained in their orthogonal, codes held invariant by the action of a group, codes meeting a bound, and a host of others. It is then possible that these objects can have a variety of applications both within mathematics and outside of mathematics, which will depend on the choice of alphabet, metric, and inner-product.

11. Conclusion

As it is now, coding theory is the name for two distinct things. The first is a branch of engineering that has applications in electronic communication, information retrieval and storage, quantum computing, cryptographic applications, secret sharing, and a host of other applications. Given the explosion of computer innovations in the past 40 years, each new innovation seems to bring with it a new use for the techniques developed in coding theory. Secondly, coding theory is a branch of pure mathematics. One might think of this as a three petaled flower. The first petal is the connection between codes and topics in combinatorics such as mutually orthogonal latin squares, finite planes, orthogonal arrays, and t-designs. The second petal is number-theoretic, with the fascinating connection between codes and lattices, forms, and the geometry of numbers. The third petal is a branch of abstract algebra that shares concepts with group theory, ring theory, field theory, module theory, and linear algebra in its broadest definition. These three flowers share a common center, which is the classical theory of codes. While the three petals can seem distinct, results in one petal can sometimes be reflected in results in one or both of the other petals. These petals are still very highly connected.

Many of the results in the pure mathematical part of coding theory still have applications in the vast array of applications in the engineering world of codes. However, there are numerous results that would make Hardy laugh with glee at their utter uselessness in the outside world. These are results which, at present, are so removed from any application that no engineer would find any interest in them at all. However, if the history of the discipline has taught us anything, it may only be a matter of time before they find application in some unlikely location.

There is also a strong bridge between these worlds. New applications bring new questions into the pure branch of coding theory. For example, the rush to build quantum technology has provided numerous questions for coding theorists, which has brought them into dialog with theoretical physicists as well.

In a department talk at Lehigh University in 1991, Ed Assmus began his talk on coding theory with a statement that read: "The purpose of applied mathematics is to enrich pure mathematics". The statement caused a great amount of good natured boos and cheers. From his perspective, the applications of coding were important because they enriched pure mathematics. This deliberately provocative statement might not have the complete support of the author of this present work, but he is certainly grateful that these applications were able to spark such interesting and compelling pure mathematics. From the point of view of the author, coding theory is a paradigmatic example of the healthy relationship between pure and applied mathematics in which both benefit highly from the other and spark interesting avenues of research in both.

References

[1] E.F. Assmus, J.D. Key, Designs and their codes, Cambridge Tracts in Mathematics, **103**, Cambridge: Cambridge University Press. x, 352 p., 1992.

[2] E.F. Assmus, H.F. Mattson, New 5-designs, J. Comb. Theory, **6**, 122-151, 1969.

[3] E.F. Assmus, H.F. Mattson, Coding and combinatorics, SIAM Rev., **16**, 349-388, 1974.

[4] J. Borges, C. Fernández-Córdoba, J. Pujol, J. Rifà, M. Villanueva, $\mathbb{Z}_4\mathbb{Z}_2$-linear codes, Cham: Springer (ISBN 978-3-031-05440-2/hbk; 978-3-031-05443-3/pbk; 978-3-031-05441-9/ebook). xii, 245 p., 2022.

[5] A.R. Calderbank, E.M. Rains, P.W. Shor, N.J.A. Sloane, Quantum error correction via codes over $GF(4)$, IEEE Trans. Inf. Theory 44, No. 4, 1369-1387, 1998.

[6] J.H. Conway, N.J.A. Sloane, [E. Bannai, R.E. Borcherds, J. Leech, S.P. Norton, A.M. Odlyzko, R.A. Parker, L. Queen, B.B. Venkov] Sphere packings, lattices and groups. With additional contributions by E. Bannai, R. E. Borcherds, J. Leech, S. P. Norton, A. M. Odlyzko, R. A. Parker, L. Queen

and B. B. Venkov. 3rd ed. Grundlehren der Mathematischen Wissenschaften. 290. New York, NY: Springer. lxxiv, 703 p., 1999.
[7] P. Delsarte, An algebraic approach to the association schemes of coding theory, Philips Research Reports. Supplements **10**, Ann Arbor, MI: Historical Jrl. vi, 1973.
[8] S.T. Dougherty, Algebraic Coding Theory over Finite Commutative Rings, Springer Briefs in Mathematics. Springer, 2017.
[9] S.T. Dougherty, Dualities for Codes over Finite Abelian Groups, Advance in Mathematics of Communication, Volume 18, Issue 6, 1827-1841, 2024, Doi: 10.3934/amc.2023023, 2023.
[10] L. Euler, Commentarii Academiae Scientarum Imperialis Petropolitanae, 8, 128-140, 1736.
[11] L. Euler, Recherches sur une nouvelle espèce des quarrés magiques, Leonardi Euleri Opera Omina Ser. I, Vol 7, 291-392, 1923, Tuebner, Berlin-Leipzig.
[12] A. R. Hammons, P.V. Kumar, A.R. Calderbank, N.J.A. Sloane, P. Solé, The $\mathbb{Z}_4$-linearity of Kerdock, Preparata, Goethals, and related codes, IEEE Trans. Inf. Theory, **40**, no. 2, 301-319, 1994.
[13] F.J. MacWilliams, N.J.A. Sloane, J.G. Thompson, On the existence of a projective plane of order 10, J. Comb. Theory, Ser. A 14, 66-78, 1973.
[14] C.W.H. Lam, L. Thiel, S. Swiercz, The non-existence of finite projective planes of order 10. Can. J. Math. 41, No. 6, 1117-1123, 1989.
[15] R.G. Gallager, Low-density parity-check codes. IRE Trans. Inform. Theory IT, **8**, 21-28, 1962.
[16] M.J.E. Golay, Notes on Digital Coding, Proc. IRE. 37: 657, 1949.
[17] V.D. Goppa, Algebraic-geometric codes, Math. USSR, Izv. 21, 75-91, 1983; translation from Izv. Akad. Nauk SSSR, Ser. Mat. 46, 762-781, 1982.
[18] V.D. Goppa, Codes on algebraic curves, Sov. Math., Dokl. 24, 170-172, 1981; translation from Dokl. Akad. Nauk SSSR 259, 1289-1290, 1981.
[19] M. Greferath, S.E. Schmidt, Finite-ring combinatorics and MacWilliams equivalence theorem, J. Combin. Theory A, **92**, 17-28, 2000.
[20] R.W. Hamming, Error detecting and error correcting codes, Bell Syst. Tech. J., **29**, no. 2, 147-160, 1950.
[21] Huffman, W. Cary; Pless, Vera Fundamentals of error-correcting codes. Cambridge: Cambridge University Press (ISBN 0-521-78280-5/hbk). xvii, 646 p. (2003).
[22] N. Levinson, N., Coding Theory: A Counterexample to G. H. HardyÕs Conception of Applied Mathematics. The American Mathematical Monthly, 77(3), 249-258, 1970, https://doi.org/10.1080/00029890.1970.11992464.
[23] G. Leibniz, Nova Methodus pro Maximis et Minimis". Acta Eruditorum 3: 467Ð473, 1684.
[24] F.J. MacWilliams, Combinatorial Problems of Elementary Group Theory, Ph.D. thesis, Harvard University, 1961.
[25] F.J. MacWilliams, A theorem on the distribution of weights in a systematic code, Bell System Tech. J., **42**, 79 - 94, 1963.
[26] F.J. MacWilliams, N.J.A. Sloane, The theory of error-correcting codes. Parts I, II. North-Holland Mathematical Library. Vol. 16. Amsterdam-New York-Oxford: North-Holland Publishing Company. Part I: xv, 369, 1977.
[27] I. Newton, Philosophae Naturalis Principia Mathematica, 1687.
[28] E. Prange, Cyclic error correcting codes, TN-57-103, 1957.
[29] E. Prange, Some cyclic error-correcting codes withs simple decoding algorithms, TN-58-156, 1958.
[30] E. Prange, The use of code equivalence in the analysis of decoding group codes, TN-59-164, 1959.
[31] E. Prange, An alogrism for factoring $x^n - 1$ over a finite field. TN-59-164, 1959.
[32] I.S. Reed, A class of multiple-error-correcting codes and the decoding scheme, Transactions of the IRE Professional Group on Information Theory, **4**, no. 4, 38-49, 1954, doi:10.1109/tit.1954.1057465.
[33] I.S. Reed, G. Solomon, Polynomial Codes over Certain Finite Fields, Journal of the Society for Industrial and Applied Mathematics, **8**, no. 2, 300-304, 1960, doi:10.1137/0108018.
[34] V.S. Pless, V. S., W.C. Huffman, W. C., Handbook of coding theory. Vol. 1. Part 1: Algebraic coding. Vol. 2. Part 2: Connections, Part 3: Applications. Amsterdam: Elsevier., 1998.
[35] B. Segre, Ovals in a finite projective plane, Can. J. Math. 7, 414-416, 1955.
[36] C.E. Shannon, A mathematical theory of communication, Bell Syst. Tech. J. 27, 379-423, 623-656, 1948.
[37] C.E. Shannon, Communication in the Presence of Noise, in Proceedings of the IRE, **37**, no. 1, 10-21, Jan. 1949, doi: 10.1109/JRPROC.1949.232969.
[38] C.E. Shannon, Communication theory of secrecy systems, Bell Syst. Tech. J., **28**, 656-715, 1949.
[39] M.A. Tsfasman, S.G. Vlădut, Algebraic-geometric codes. Transl. from the Russian, Mathematics and Its Applications, Soviet Series, 58. Dordrecht etc.: Kluwer Academic Publishers. xxiv, 1991.

[40] M.A. Tsfasman; S.G. Vlădut, Th. Zink, Modular curves, Shimura curves, and Goppa codes, better than Varshamov-Gilbert bound, Math. Nachrichten, **109**, 21-28, 1982.
[41] J. Wood, Duality for modules over finite rings and applications to coding theory, Amer. J. Math., **121**, no. 3, 555 - 575, 1999.
[42] N. Weldon, Comment made at the first IEEE Communications Theory Workshop, St. Petersburg, 1970.

Steven T. Dougherty is a Professor of Mathematics at the University of Scranton, Pennsylvania. He is the author of two books and over 140 academic papers in information theory, number theory, combinatorics, abstract algebra, and the history of mathematics. He was awarded the 2005 Hasse prize by the Mathematical Association of America for a paper in game theory.

(Steven T. Dougherty) University of Scranton, Scranton PA, USA
E-mail address: prof.steven.dougherty@gmail.com