Error correcting codes in secondary mathematics education, a teaching experiment

Presenting author **Johan Deprez**
Universiteit Anwerpen
Hogeschool-Universiteit Brussel
Katholieke Universiteit Leuven

Co-authors Kris Annaert
Moretus Katholiek Onderwijscentrum Ekeren
Dirk Janssens
Katholieke Universiteit Leuven
Joos Vandewalle
Katholieke Universiteit Leuven

Abstract Making industrial mathematics more visible makes people better aware of the importance of mathematics in modern technologies. We present a teaching experiment in which secondary school students are introduced to a topic in industrial mathematics, namely error correcting codes. The experiment started in 2007 and is still going on. The teaching sequence, designed by students from teacher education, has been tested on a small scale in the academic year 2008-2009. It was revised during the first semester of 2009-2010 and is tested again on a larger scale during the second semester. It connects abstract algebra (examples of finite fields) to interesting applications in technology: the (7,4) Hamming code and the (6,4) Reed Solomon code.

Introduction

The discussion document for the ICMI/ICIAM International Study concerning Educational Interfaces between Mathematics and Industry (Damlamian, & Sträßer, 2009) mentions that "people are often not aware of the importance of the role of mathematics in modern technologies. Many people have a restricted view of what mathematics is and does" and that "[w]e need to make the use of mathematics in modern society more visible" (p. 527). In fact quite often the mathematics is hidden inside and is used in practice in an unnoticed way, e.g. typically $1/4^{th}$ of the bits stored on a CD disk are there for the mathematics and not for the music or the software. In question 1 of paragraph 2 (p. 527) the discussion document asks "[h]ow [...] mathematics, especially industrial mathematics, [can] be made more visible to the public at large". This paper presents a teaching experiment in which a topic in industrial mathematics, namely error correcting codes, is made clear for various applications in information and communication technology (ICT). Examples of applications of mathematics in industry are introduced more often in higher education than in secondary education. But there is no obvious reason for not introducing such applications in the secondary school. So our teaching experiment was carried out with secondary school students.

The teaching experiment was set up not only to make industrial mathematics visible to students, it also intended to make mathematics more appealing and exciting to students, which refers to question 2 in paragraph 2 of the discussion document (p. 527). In fact, the subject of error correcting codes provides opportunities to connect abstract and modern mathematics to exciting applications, thus simultaneously opening two completely new worlds for secondary school students.

The official curriculum in Flanders (Belgium), where the teaching experiment was held, stresses the importance of applications. More and more applications of mathematics are found in school books and in class rooms nowadays. However, many of the problem situations are not 'really real'. One of the reasons to construct the teaching sequence presented in this paper, was to raise the level of authenticity of problems used in the class room.

Not only the subject is new to the secondary school students (and their teachers) but to a certain extent also the teaching methods and the assessment system. Students had to study the subject in groups guided by a text and they received not too much help from the teacher. The assessment was based on a test of students' knowledge of the subject at the end of the teaching sequence and on the process of studying the text in groups.

The teaching experiment was and still is prepared and carried out by student teachers, first of the KU Leuven and later of the Universiteit Antwerpen. The second named author is

one of the student teachers involved in the experiment. The other authors supervise the experiment.

We would like to thank here the teachers who were prepared to let the experiment happen in their class rooms, especially Rita De Ridder and Ann Vanderkimpen, who were engaged in the small scale testing of the first version, and Isabelle Poels, who drafted a preliminary version of the text used during the experiment.

Description of the teaching experiment

The experiment started in the academic year 2007–2008 when a student teacher from the KU Leuven wrote a first draft of a text for secondary school students about error correcting codes. This text was based on Rousseau and Saint-Aubin (2008), a book which is intended for tertiary education. During the academic year 2008-2009 another student teacher of the KU Leuven revised the text, in collaboration with two secondary school teachers who used the text in their classes. During the academic year 2009–2010, a third student teacher, now from the Universiteit Antwerpen, continues the experiment. During the first semester, she revised the text again, based on the experiences with secondary school students in the previous year. A group of 12 teachers will use the revised text in their classes during the second semester.

We discuss the text in more detail in the following paragraphs. Now we only give a short overview. The text (more specifically: the revised version used during the academic year 2009–2010) consists of six chapters. The first chapter presents examples of situations in which error correcting codes are used and gives some general information concerning error correcting codes. In the second chapter arithmetic modulo 2 is introduced. This is applied in the third chapter, which is dedicated to the (7,4) Hamming code. Chapter four deals with modular arithmetic in general. Examples of finite fields with a prime number of elements are introduced and used in small applications like International Standard Book Numbers (ISBN) and EAN product codes. In chapter five, two examples of finite fields whose number of elements is a power of a prime number, are constructed using modular arithmetic with polynomials. The final chapter uses the knowledge from chapter five to study a simplified version of the (6,4) Reed-Solomon code.

The text is meant to be studied more or less independently in small groups of students with only few interventions of the teacher. Short pieces of information (definitions, examples, ...) alternate with exercises that stimulate students to study the text actively. At the end of each chapter some more challenging problems are given.

Two teachers were involved in the small scale testing during the academic year 2008–2009. Teacher A had a class of 18 students in grade 11 and 12, all of them being relatively good in mathematics (top 20%). The class of teacher B was in the same range of age, but much more mixed in terms of ability in mathematics (top 50%). It contained 10 students. Teacher A and teacher B spent 6, respectively 10, lessons (of 50 minutes) to the topic. Unfortunately, due to a lack of time, it was not possible to deal with the last chapter. The student teacher observed the class and helped the teachers in supervising and giving feedback to the work of the students.

The larger scale testing during the academic year 2009–2010 involves 12 teachers and around 150 students. Again, students are in grade 11 and 12. Most of them belong to the top 20% in terms of ability in mathematics. The student teacher will observe a number of classes. Moreover, she will record conversations of students in groups while working with the text, she will interview some of the teachers and administer a questionnaire to the participating students.

Overview of the text

Here, we give a more detailed description of the text used in the teaching experiment (more specifically: the revised version used in the larger scale teaching experiment during the academic year 2009-2010). In the following paragraphs, then, we show a series of excerpts from the text, giving the reader a more detailed impression of the material covered and the teaching method.

The first of the six chapters of the text starts by an introduction to the binary number system and its use in the transmission of information. Next, it describes two situations in which error correcting codes are used: transmitting of images and sound (CD). This introduction intends to motivate the students. Then the key idea of error correcting codes is introduced. When transmitting information in real life, errors are inevitable. In order to be able to detect and possibly correct certain of these errors after transmission, redundant information is added in some intelligent way. During the whole teaching sequence, this idea will be illustrated by a number of concrete examples. Generalization to a more abstract theory of error correcting codes is not the goal here. In the first chapter, the examples are very simple: the NATO phonetic alphabet, repetition codes and parity codes. Finally, in the first chapter, some terminology is introduced: alphabet, code word, one error detecting code and one error correcting code, information rate, ...

The (very short) second chapter introduces a system for calculations which is new to the students: modular addition and multiplication with the two elements 0 and 1. For future use, vectors and matrices based on this system are introduced as well.

The new mathematics introduced in the second chapter is immediately applied in a technological context in the third chapter, where it is used to explain the principle of the (7,4) Hamming code. In this way, mathematics is connected to the real world right from the start, with the intention of enhancing the motivation of the students to study the abstract mathematics. The Hamming code itself is motivated by remarking that it is a code with a better information rate than the codes in the first chapter. The (7,4) Hamming code is defined and it is proved that it can detect and correct one error. Then it is shown how to use Venn-diagrams and/or matrices (generator matrix, parity-check matrix) to encode and decode information.

The fourth chapter generalizes modular arithmetic with the two elements 0 and 1 to modular arithmetic in general. In some of these new systems for calculations, the basic properties of the traditional systems of calculation are conserved. This leads to the concept of the finite fields $\mathbb{F}_p$ where p denotes a prime number. At the end of this chapter, a number of smaller scale applications are treated: (Belgian) bank account numbers, International Standard Book Numbers (ISBN) and EAN product codes.

Now that students have seen that new systems of calculation are very useful in a technological context, they are introduced to yet another (and more complicated) new system for calculations in chapter five. Now, fields whose number of elements is a power of a prime number are constructed. The elements of a field with p^n elements are polynomials with coefficients in the field $\mathbb{F}_p$ and degree at most $n-1$. Addition is the normal addition of polynomials, adding its coefficients modulo p. Multiplication is more complicated. The normal multiplication of polynomials may yield a polynomial having degree n or more. Therefore, the product of two polynomials in the new sense is the remainder of the 'normal' product after division by a given irreducible polynomial. We do not study finite fields in general, but treat two examples: $\mathbb{F}_8$ and $\mathbb{F}_{16}$. The elements of the fields are represented in different forms: as a polynomial with restricted degree, a number in decimal form, a number in binary form or a power of a primitive element. Some representations lend themselves better to do additions, whereas others are better suited for multiplications.

Again, the abstract mathematical constructions and calculations are applied in a technological context as soon as possible. The final chapter deals with a simplified version of the (6,4) Reed-Solomon code, which makes use of calculations in the field $\mathbb{F}_{16}$. Encoding and decoding are discussed in detail using concrete examples of messages in which errors are introduced during transmission. The techniques that are needed give the occasion to enlarge and revise the meaning of classical mathematical objects (numbers, polynomials, vectors, matrices ...) and calculations by using these objects in decoding information that is stored or sent.

Examples from chapter 3: The Hamming code

The (7,4) Hamming code adds to a set $a_1a_2a_3a_4$ of four information bits three control bits c_5, c_6 and c_7 defined by

$$c_5 = a_1 + a_3 + a_4,\ c_6 = a_1 + a_2 + a_4,\ \text{and}\ c_7 = a_1 + a_2 + a_3,$$

(where addition is modulo 2) resulting in code words of seven bits. The text introduces students to this definition and they do some exercises (e.g. calculating the control bits for a message, finding the number of possible code words) to make sure that they understand the rule. Then it is shown that the Hamming code is able to detect and correct one error, based on a property from the first chapter: if different code words differ in at least three positions, the code is one error correcting. The case of messages differing in one position, giving rise to code words differing in three or four positions, is demonstrated in the text. Our first excerpt (translated from Dutch to English) from the text is problem 27, in which the students are asked to deal more independently with the case of messages differing in two positions.

> Problem 27. Assume that two messages differ in two positions. Show that the corresponding code words differ in at least three positions by considering the following cases:
> a. a_1 and a_2 are different;
> [...]
> f. a_3 and a_4 are different.
> Hint: Some of the cases are analogous!

Finally, the text finishes the proof.

We show a second, more practically oriented example from this chapter. In the part concerning decoding, students receive the following (simple) problem.

> Problem 34. Use the definition of the Hamming code to find the error in the received word 1010000.

Students can consult model solutions for the problems. Here, the model solution goes as follows.

> We verify the defining equations of the Hamming code:
>
> $$1 + 1 + 0 = 0?\ ,\ 1 + 0 + 0 = 0?,\ 1 + 0 + 1 = 0?$$
>
> and conclude from this that the second equation is not fulfilled. Changing the second control bit from 0 to 1, gives three fulfilled equations. Hence, the code word is 1010010.

Students also learn to encode and decode using Venn diagrams. If they have already studied matrices in their mathematics course, they are also introduced to encoding and decoding using matrices.

The final problem in this chapter gives the students the opportunity to run through the whole process of encoding and decoding.

> Problem 39. Draft a message and translate it into a sequence of bits using the ASCII-code in table 1.1. Encode this message using the (7,4) Hamming code and send it to one of your fellow students after inserting a few errors. You will receive a message from one of your fellow students. Decode this message.

Examples from chapter 4: Prime fields

The start of this chapter is purely mathematical in nature. Students meet examples of calculation systems based on modular addition and multiplication. They experience that this leads to a system sharing the basic properties of the real number system in some cases (i.e. arithmetic modulo a prime number), whereas the resulting system lacks some of these properties in other cases. This is shown in the following problem, which comes after the text has treated the non-existence of a multiplicative inverse for the number 4 in $\mathbb{Z}_6$.

> Problem 49. Fill in the table below

Number	Calculation	Conclusion The multiplicative inverse in $\mathbb{Z}_6$
1	...	...
2	...	...
3	...	...
4	...	does not exist
5	...	...
6	...	...

At the end of this chapter, we added a number of smaller applications in the realm of error correcting codes. We show one example, concerning the (10-digit) ISBN.

> Project 2.v. A common error when typing in an ISBN is that two successive digits are interchanged, for example: 0–1311–0362–8 is typed in as 0–1311–0326–8. Show that detection and correction of this type of error is possible.

Example from chapter 5: Extension fields

The whole chapter is of a purely mathematical nature and is moreover rather difficult for the students. Two examples are treated. The construction of the field $\mathbb{F}_8$ of eight elements

is explained in detail to the students, while they do a number of smaller exercises to ensure that they understand the construction. Then, the students more or less independently construct the field $\mathbb{F}_{16}$ of sixteen elements, guided by a number of questions, as is shown in the problem below.

Problem 63. Construct the field $\mathbb{F}_{16}$ by copying the construction of $\mathbb{F}_8$.

a. How many elements has the field $\mathbb{F}_{16}$? Describe all elements using their polynomial, binary and decimal notation.

[...]

c. How do we multiply in $\mathbb{F}_{16}$? Like in $\mathbb{F}_8$, we experience the problem that multiplication in the ordinary sense may yield a polynomial of a too high degree. Now, the polynomial used to reduce the degree, is $X^4 + X + 1$. Work out the following multiplications:

1. $(X^2 + 1) \cdot (X + 1)$;
2. $(X^2 + X) \cdot (X^2 + X + 1)$;
3. $(X^3 + X^2 + 1) \cdot (X^3 + 1)$.

d. The polynomial X is a primitive element in $\mathbb{F}_{16}$ as well. Show this by calculating the following powers of X:

[...]

f. Do the following multiplications, rewriting the polynomials as powers of X first:

1. $(X^3 + X^2 + 1) \cdot (X^2 + X)$;
2. $(X^3 + 1) \cdot (X + 1)$.

Examples from chapter 6: Reed Solomon code

The final chapter of the text is dedicated to the study of a simplified version of the Reed Solomon code. Using the knowledge from chapter 5, students are able to understand the principles of coding and decoding following this code. We show an example of the work that we expect students to do while studying the principle of encoding messages using the Reed Solomon code. It is explained that messages now have the form $a_1a_2a_3a_4$, where now each of the four 'symbols' a_i is no longer an individual bit, but an element of $\mathbb{F}_{16}$, i.e. messages consist of four times four bits. To these information symbols, two control symbols c_1 and c_2, also elements of $\mathbb{F}_{16}$, are added such that

$$a_1 + a_2 + a_3 + a_4 + c_5 + c_6 = 0 \tag{6.1}$$

$$a_1X^5 + a_2X^4 + a_3X^3 + a_4X^2 + c_5X + c_6 = 0 \tag{6.2}$$

Next, the text asserts that c_5 and c_6 can be solved from these equations, producing the following expressions:

$$c_5 = a_1X^6 + a_2X^{12} + a_3X^{10} + a_4X^4 \tag{6.3}$$

$$c_6 = a_1X^{13} + a_2X^{11} + a_3X^5 + a_4X \tag{6.4}$$

Then students are asked to verify this, guided by a number of questions, in the following problem.

Problem 65. Show that the expressions for c_5 and c_6 given in 6.3 and 6.4 indeed follow from the equations 6.1 en 6.2. Do this along the following lines:

1. Solve c_6 from equation 6.1.
2. Plug your expression for c_6 into equation 6.2.
3. Use the power of X notation to work this out.
4. Find the multiplicative inverse of X^4 in $\mathbb{F}_{16}$.

HINT: The inverse of X^4 is x if and only if $X^4 \cdot x = 1$.

5. Solve c_5 from the equation found in question 3 while making use of what you obtained in step 4.
6. Plug your expression for c_5 into the expression for c_6.

After having learned also the decoding procedure, students have to do some practical work, as is shown in the following example.

Problem 70. Decode the following message:

0100011001000101110101010001010101010010111111011100.

Use the ASCII-code.

HINT: The message consists of two code words.

Of course, the text ends with a problem in the style of problem 39 discussed above, in which students can run through the whole process of encoding and decoding using their own message.

Experiences

First, we report on the experiences with the small scale testing during the academic year 2008–2009. The version of the text used then differed from the revised text discussed in the previous paragraphs. The main difference is that in 2008–2009 modular arithmetic modulo another number than 2 was treated before Hamming codes. The participating students filled in a questionnaire when the chapter on the Hamming code had been treated in class. Hence, students had not studied extension fields and Reed Solomon codes yet. We discuss some of the results here. Moreover, we give a number of remarks made by the teachers and the authors concerning the text and the experiences in class.

The two application oriented chapters, the introductory chapter and certainly the chapter concerning Hamming codes, were considered the most interesting ones by the students. They reported that they had learned a lot in these chapters. Moreover, they appreciated that these chapters were practically oriented, entirely new to them, and that they contained information about real world applications of what they studied. Students also appreciated the fact that these chapters gave them the opportunity to do a lot of exercises. They especially liked to decode coded texts by themselves. They found these chapters not too difficult.

The introduction to modular arithmetic and finite fields having a prime number of elements, was considered the most difficult part of the text. Students found this part the most complex one and had the impression that it might have been explained in a simpler way. Moreover, this part had a high level of abstraction, according to the students.

In general, students appreciated the topic of error correcting codes and more than half of the students reported that it had changed their opinion concerning mathematics.

Concerning the teaching method, students having a better background in mathematics liked working independently in groups. They appreciated having the opportunity to struggle with problems themselves and finding alternative solutions. They reported that they were more concentrated during the lessons. Students whose background in mathematics was poorer, showed considerably less appreciation for the teaching method. One possible explanation, according to the teachers, is that the text may have been too difficult for them.

As to the teachers and the student conducting the experiment, the main point concerning the teaching method, is that even though the text is self-contained, the role of the teacher remains very important. For both classes, the teachers wrote a detailed set of instructions, indicating the parts of the text to be studied, the exercises to be done, the homework to be done, ... Moreover, at the beginning of the teaching sequence and of each chapter, a stimulating introduction by the teacher is needed. Reviewing and summarizing important ele-

ments in the text, dealing with some of the more difficult passages in the text, correcting misconceptions, ... are other important tasks for the teacher.

According to the supervisors, a point that had to be considered when revising the text after the small scale experiment in 2008–2009 was the position of modular arithmetic using other numbers (and more specifically: primes) than 2. In the original text the first part mainly focused on calculations modulo 2, whereas in the second part calculations modulo a polynomial are needed. There is a huge difference in level of difficulty between these two parts. We felt that, for didactical reasons, it was important to give more attention to calculations modulo numbers distinct from 2. This is realized indeed in the revised text used during the academic year 2009-2010 and discussed above. The (limited) material on arithmetic modulo a number distinct from 2 has been removed from chapter 2, where it did not receive enough attention and was not used in the application (Hamming code) that followed, and it has been inserted in a slightly expanded form into a separate chapter (chapter 4) in the new text. This chapter deals with modular arithmetic in general and discusses some short applications. We feel that the text is more coherent now and hope that students are better prepared for calculations modulo a polynomial. At least, the first reactions from the teachers are positive.

Conclusion and discussion

As the teaching experiment is still going on, we can only present provisional conclusions here. One of our aims was to make the role of mathematics in modern technologies more visible. In this respect, a majority of students reported that the teaching experiment had changed their view concerning mathematics. Concerning our focus on the secondary school level, the devised text and the first experiences show that error correcting codes and the necessary (non-trivial and abstract) mathematics can indeed be presented and understood on secondary school level, at least if the students belong to the top 20% range of ability in mathematics. Moreover, students reported that they were really interested in the topic of error correcting codes, showing that this topic may help to make abstract mathematics more appealing and exciting, as was intended. Finally, it was seen that students were able to deal with the topic more or less independently guided by a text, but that the role of the teacher remains very important.

References

Annaert, K. (2009). *Foutenverbeterende codes: een project voor de derde graad van het secundair onderwijs* [Error correcting codes: a teaching experiment for grades 11 and 12 of secondary education]. Unpublished master thesis Katholieke Universiteit Leuven (in Dutch).

Damlamian, A., & Sträßer, R. (2009). ICMI Study 20: educational interfaces between mathematics and industry. Discussion document. *ZDM Mathematics Education*, *41*, 525–533. doi 10.1007/s11858-009-0194-4

Rousseau, C., & Saint-Aubin, Y. (2008). *Mathematics and Technology* (pp.173-207, 291-323). Montreal, Canada: Springer.