Algebraic coding theory

CHAPTER 10 - Special Mathematical Topics

10.1 ALGEBRAIC CODING THEORY

ALGEBRAIC CODING THEORY

J.H. van Lint
Eindhoven University of Technology
Eindhoven, Netherlands

Introduction

One of the purposes of the mini-conferences at ths meeting is to introduce new areas of mathematics which have found their way into the curriculum at some universities and colleges but deserve more attention. At the same time one can consider the question whether it concerns a development in mathematics that could in some way be incorporated in the high school curriculum. In the present case the topic is algebraic coding theory, an area between information theory, combinatories and applied algebra, which has only been around for about 30 years. More than likely at most of the world's universities there has never been a course in this subject. On the other hand at a few it has been taught for at least 15 years and it is usually received with enthusiasm by the participating students.

I am going to assume that the reader has either no knowledge or only a vague idea of what coding theory is. We shall start by showing the origin of coding theory by using a few examples. This introduction will also serve as an example of how the subject can be introduced to students. The introduction is also designed to show that the subject can be introduced at the high school level.

By no means do I want to advocate the introduction as a subject at this level. However, I do believe that for little groups of students with an interest in mathematics, or for special projects, or even as a part of a general science course, certain parts of the subject are quite suitable. At a somewhat higher level I believe the subject is extremely suitable as an illustration of an area which is important for practical reasons, where all kinds of mathematics (which the students by then have learned in introductory courses) are applied. One of the things coding theory can illustrate is that a mathematician needs a very broad basic education and that he will often be surprised about the places where he can apply his knowledge.

The second part of the paper sketches a possible course content and serves as a continuation of the introduction. We stress the algebraic aspects.

Most teachers of combinatorics are well aware of the possibilities of coding theory and often have a section on the subject in their course. For this reason we shall hardly go into this part of the subject.

In the final part we look at the possibility of using certain tipics from coding theory in courses on other parts of mathematics, showing connections or applications.

The main reference is "The Theory of Error-Correcting Codes" by F.J. MacWilliams and N.J.A. Sloane, which contains just about everything there is to know about algebraic and combinatorial coding theory and has a list of nearly 1500 references. The authors are preparing a short introductory text book which will appear in 1981 (Springer-Verlag). For some subjects this book is given as a reference. The second part of the References will mention suitable introductions into this area.

Error-correcting codes

We illustrate error-correcting codes by means of a well known recent example. Many readers will have seen the excellent pictures which were taken of Mars, Saturn and other planets by satellites such as the Mariners, Voyager, etc. In order to transmit these pictures to earth a fine grid is placed on the picture and for each square of the grid the degree of blackness is measured in a scale of 0 to 63. These numbers are expressed in the binary system, i.e. each square produces a string of six 0's and 1's. The 0's and 1's are transmitted as two different signals to the receiver station on earth (the Jet Propulsion Laboratory of the California Institute of Technology). On arrival the signal is very weak and it must be amplified. Due to the effect of thermal noise it happens occasionally that a signal which was transmitted as a 0 is interpreted by the receiveer as a 1 and vice versa. If one simply transmits the sextuples mentioned above, the errors would have great effect on the quality of the pictures. In order to prevent this, so-called redundancy is built into the signal, i.e. the transmitted sequence consists of more than the necessary information. We are all familiar with the principle of redundancy from everyday language. The words of our language form a small part of all possible strings of letters (symbols). Consequently a misprint in a long (!) word is recognized because the word is changed into something which resembles the correct word more than it resembles any other word we know. This is the essence of the theory of error-correcting codes. In the example of the Mariner 1969 the sextuples of 0's and 1's were mapped into strings of 32 symbols. If a received string contained less than 8 errors the decoding device interpreted it correctly. We illustrate the idea by a smaller example.

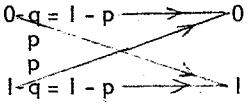
0 = 000	000	00000
1 = 001	001	10110
2 = 010	010	10101
3 = 011	011	00011
4 = 100	100	10011
5 = 101	101	00101
6 = 110	110	00110
7 = 111	111	10000

Here the integers from 0 to 7 are written in the binary system and then mapped into codewords of length 8 (i.e. strings of eight 0's and 1's).

The reader can easily check that any two codewords differ in four positions (and that it is not difficult to add more codewords to the list which also differ in at least four positions from all the previous ones). If we just use the eight words given above we can drop the fifth symbol since it is zero for all of the words. In the example of the Mariner 1969 a codeword of length 32

consisted of six information symbols and the rest was redundancy. We say that this code has information rate 6/32.

It is not difficult to understand that if we are willing to use extremely low information rate then t is possible to correct many errors in each received word. E.g. if we repeat each symbol (0 or 1) a number of times, say 2n + 1 times, then a received string of 2n + 1 symbols may contain up to n errors and still be interpreted correctly on the basis of maximum likelihood. The information rate is now 1/(2n + 1). The following example illustrates in a simple way why coding theory has become so interesting and important.
We conduct an experiment in which a coin is tossed many times and the results heads (= 0) resp. tails (= 1) are transmitted over a so-called binary symmetric channel:



(The probability that a symbol is received incorrectly is p.) Let us assume that the channel is fairly good, say p = 0.001. If we simply transmit our results then the receiver will have a 0.1% incorrect information at the end of the experiment. Suppose the channel can handle two symbols in the time needed for a toss of the coin. In the terminology used earlier we can transmit with information rate 1/2. The old educational principle of attempting to be better understood by saying everything twice does not work in this example! If we transmit 00 resp. 11 instead of 0 rep. 1 then the receiver will know that a received sequence 01 is incorrect but he cannot correct the error. Now we wait a few seconds until ten tosses have taken place and then start transmitting twenty symbols for every ten tosses of the coin; (again the information rate is 1/2). We need a mapping f from $(0,1)^{10}$ to $(0,1)^{20}$ or in other words we need a subset of 2^{10} elements from $(0,1)^{20}$ which we call a code C consisting of s^{10} codewords of length 20. Since we have 2^{20} words from which to pick it is not so very surprising that it is possible to choose C in such a way that any two codewords from C differ in at least six places. This means that if a received word contains one or two errors these can be corrected. It is not difficult to calculate the probability that a received word contains more than two errors and thus discover that the receiver now has about 0.001% incorrect information.

Shannon's famous theorem which marks the beginning of this area of research states that in fact the probability of error can be made arbitrarily small (in our example with p = 0.001 and information rate 1/2). The important thing to note is that this is achieved without lowering the information rate! The proof of this theorem is based on a probabilistic argument. So the question remains, to this day, how to construct good codes.

This has been a short and sketchy introduction.

If the reader has become interested he can read more about what was explained above in the references (5), (7), (9), (10), (a), (b), (c), (d), (e), (f).

A course on coding theory

A course on coding theory should start with several examples of communication channels affected by some kind of noise, followed by a motivation for working in this area such as the previous paragraph.

Of course, at the university level one actually gives a rigorous proof (and a precise statement) of Shannon's theorem. Before embarking on general thoery it is good to illustrate that usually three different aspects can be distinguished. First of all there is the definition of the code C, i.e. the question of constructing in some systematic way the mapping f from information strings $(a_0, a_1, \ldots, a_{k-1})$ to codewords $(c_0, c_1, \ldots, c_{n-1})$. Then follows a detailed analysis of the properties of the code C, the most important being the so-called minimum distance. Finally one can go into the question of decoding algorithms of different kind, and even complexity questions can be considered. In many courses for mathematics students the third part, which is of a more practical nature, is often hardly treated at all.

Once the setting is clear the actual mathematics starts. We introduce structure. The first thing to do is to spend some time on the theory of finite fields. If the course is for mathematics majors one can go into this extensively or possibly it is already known to the students. I claim that even if one chooses coding theory as a special topic to show (better) high school students some interesting application of mathematics including several things they may have had to learn without realizing where it is used, then one can easily explain F_2, even F_p and maybe also the fields F_4, F_8, etc.

Let us introduce some terminology.

Let F be a finite field. We call F the alphabet. A code C is a subset of F^n. The distance $d(\underline{x}, \underline{y})$ of two codewords $\underline{x} = (x_1, x_2, \ldots, x_n)$ and $\underline{y} = (y_1, y_2, \ldots, y_n)$ is the number of places where they differ (i.e. $x_i \neq y_i$). The weight of $\underline{x}$ is $d(\underline{x}, \underline{0})$, where $\underline{0} := (0,0, \ldots, 0)$. The minimum distance d of the code C is

$$[d(\underline{x}, \underline{y}) : \underline{x} \in C, \underline{y} \in C, \underline{x} \neq \underline{y}]$$

Clearly d is a measure for the error-correcting capability of C. The information rate which we used in a previous section is defined as $n^{-1} \log|C|$, where the base of the logarithm is the size of the alphabet, i.e. $|F|$.

The next chapter in the course is a very nice application of linear algebra: one requires that C is a linear subspace of F^n, say of dimension k. Such a code is called an $[n,k]$ code. By just mentioning some key words we hope to give an impression of what comes up: basis, generator matrix, standard form, cosets, linear equations, inner product. A very important concept is the dual code $C^\perp$ defined by $C^\perp := \{\underline{x} \in F^n : (\underline{x}, \underline{y}) = 0$ for all $\underline{y} \in C\}$.

We illustrate this by treating the $[7,4]$ binary Hamming code. We wish to construct a linear code over the alphabet $F_2 = \{0,1\}$ which can correct one error. Therefore d must be at least 3. We do this by constructing a matrix H which has as its rows a set of basis vectors for the dual code $C^\perp$. By definition every codeword $\underline{c}$ has inner product $\underline{0}$ with every row of H, i.e. $\underline{c}\, H^T = \underline{0}$. If we change the i-th symbol of $\underline{c}$ we obtain a word $\underline{c}'$ with one error. Clearly $\underline{c}'\, H^T$ is the transpose of the i-th column of H. Hence we can correct the error if all columns of H are different!

Now, let the i-th column of H be the binary representation of the integer i ($1 \le i \le 7$):

$$H = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix}$$

We then have a very easy decoding algorithm. If $\underline{c}'$ contains one error in position i then $\underline{c}'H^T$ is the integer i in binary and $\underline{c}$ is obtained by changing the i-th received symbol back into what it should be. The three rows of H are linearly independent; so the code we have just defined is the set of solutions of three independent linear equation in seven variables, a $[7,4]$ code with d = 3. Finally we could analyse this code and show connections to several interesting areas of combinatorics.

This simple but important example was chosen to make it clear that one can teach this subject quite clearly. To really get an idea of the most important codes used in practice today one must use more algebra. Again, we mention the minimum which is necessary to be able to treat much interesting material. We need the concept of a ring, ideals, principal ideals, polynomial rings. It is useful to know something about a group algebra but we can do without this for a while. A cyclic code is a linear code C such that for every $(c_0, c_1, \ldots, c_{n-1}) \in C$ the word $(c_{n-1}, c_0, c_1, \ldots, c_{n-2})$ is also in C. The mapping $(c_0, c_1, \ldots, c_{n-1}) \rightarrow c_0 + c_1x + \ldots + c_{n-1}x^{n-1}$ maps C 1 - 1 onto a principal ideal in the ring of polynomials mod $(x^n - 1)$. This opens a wealth of possibilities and an advanced course would now go into the connections between the two representations, zeros of polynomials in extension fields of F and a discussion of the important class of BCH-codes. If "applications of algebra" is the main theme one can include much more which we shall mention below. The reader mainly interested in high school or first-year college education should not stop reading! The subject of arithmetic with polynomials is certainly taught in high school. Multiplying polynomials is not so interesting, division even more distasteful. Now study the following example and think about its possibilities. We are going to work with polynomials $c_0 = c_1x + \ldots + c_6x^6$ in which every c_i is 0 or 1. We first add, subtract, multiply in the normal way and then reduce mod 2 (so 1 + 1 = 0). Now some division: polynomials of degree greater than 6 are reduced by taking the remainder after dividing by $x^7 - 1$. Let C be the set of $(c_0, c_1, \ldots, c_6)$ such that $c_0 + c_1x + \ldots + c_6x^6$ is divisible by $x^3 + x + 1$. Then C is a $[7,4]$ cyclic code with minimum distance 3, in fact it is a cyclic representation of the Hamming code (easy to decode!) treated earlier. A lot of things that require proof, with motivation provided gratis. If one wants more: let (a_0, a_1, a_2, a_3) be a string of four information symbols; map this into $(c_0, \ldots, c_6)$ where $c_0 + c_1x + \ldots + c_6x^6 = (a_0 + a_1x + a_2x + a_3x^3)(x^3 + x + 1)$, an extremely simple encoding scheme for our code. My experience is that at this stage many students like to hear something about the actual circuitry (shift registers) which are used to do the encoding and decoding (see 1, 9). We now turn to a connectin with geometry. We introduce the idea of Reed-Muller codes by looking at a simple example. The code for Mariner 1969 which was mentioned in the introduction is of this type.

A simple approach with pictures is possible but it is better to use linear algebra since it is fairly common nowadays to use methods from linear algebra in geometry courses. We take F_2 as the alphabet. Consider the m-dimensional space V over this field (if necessary only take m = 3, i.e. look at the unit cube). Let $P_0, P_1, \ldots, P_{n-1}$ be the points of this space, numbered in some way ($n = 2^m$). Define the vector $v^{(i)}$, (i = 1, 2, . . . , m), with n coordinates by $v^{(i)} = 1$ if the i-th coordinate of P_j is 1; = 0 otherwise.

So $v_j^{(i)}$ is simply the i-th coordinate of P_j and hence $v^{(i)}$ is the characteristic function of the hyperplane $(x \in V : x_i = 1)$. The vectors $v^{(i)}$ are the basis of an $[n,m]$ linear code. The code word $\sum a_i v^{(i)}$ is clearly the characteristic function of the hyperplane $\{x : \sum a_i x_i = 1\}$. Observe that if we use only these hyperplanes then more of them will contain the origin; i.e. all words will have a 0 in the corresponding position, which is what happended in the example in the introduction. Therefore every nonzero codeword has 1/2n coordinates equal to 1. Since two hyperplanes intersect in a space of dimension m-2 we see that two different nonzero codewords have a 1 in common in 1/4n positions, i.e. their distance is 1/2n. The example with n = 8 was treated in the introduction. If we add the vector $v^{(0)} := (1,1, \ldots, 1)$ to the basis we increase the dimension of the code by 1. The new codewords are the characteristic functions of the whole space and all hyperplanes through the origin.

Starting from this example one can treat several codes which all show nice applications of both finite geometries and linear algebra.

The course which we are sketching could stop at this point. However, let us look at other topics which can be included. For the purpose of teaching, nonlinear codes are not so suitable. On the other hand, here is an area where the student who has become interested can do a lot of interesting little research projects on construction methods. Of course there is extensive literature on the subject (cf. 3, 9). A much better subject is "bounds on codes". Here the problem is to find inequalities involving $|F|$, n, $|C|$, and d.

Especially at the advanced level some fascinating mathematics comes into the picture (e.g. orthogonal polynomials, application of linear programming). For fixed n, combinatorial techniques of many kinds can be used. More important are the asymptotic results. For all these subjects we must refer to the literature.

To show that there are interesting problems even for a lower level course we consider the following problem, which is left as an exercise for the reader. We wish to construct a binary code C of length n = 6 with minimum distance d = 3. Show that C cannot have more than nine words. Then show that C cannot have nine words. An example with eight words is easily found. My experience with this exercise, with mathematics majors in their third year, is that the solutions which are given usually take a few pages although a few lines suffice. In fact, this kind of problem becomes hard extremely rapidly. If we replace the wordlength in the above problem by n = 8 then the maximal value of /C/ is 20. It is very unlikely that a reader who does not know coding theory will be able to prove this. For large values of n and d it is usually impossible to give exact values.

Special topics that can be included depending on the level of the students and the purpose of the course are e.g. arithmetic codes (used in computers, the alphabet is not a field but a set of integers, addition is normal addition), perfect codes, codes and combinatorics (cf. 3, 9), automorphism groups.

Topics from coding theory as applications

It is a good habit of many teachers of courses in some areas of "pure" mathematics to show that their subject can be applied either in other areas of mathematics of in some other science or in practice. With this purpose in mind we shall consider a number of areas of mathematics and mention possible applications in coding theory. This section is mainly intended to be a collection of suggestions for teachers of such courses. It is clearly impossible to treat the problems of coding theory which are mentioned. We give only references and some indication of the problems and their level. We do not include the obvious area of combinatorics (see 3, 9).

Linear algebra

Here there are many applications at the elementary level. We refer to the previous section and the standard text books on coding theory. As an application of systems of linear equations one can treat the concept of dual code $C^{\perp}$. Here it is worth pointing out that our intuitive idea that C and $C^{\perp}$ are complementary is not true for finite fields. In fact self-dual codes (i.e. $C = C^{\perp}$) are among the most intersting.

Algebra

We have mentioned rings and ideals in the previous section. The Goppa codes provide another nice example of the use of rings. There are several nice applications of the theory of group algebras in coding theory (cf. e.g. 5, 8, 9, 13). These usually also involve some character theory. In a course on semi-simple algebras one can include the idempotents of cyclic codes as an example (9) and for a very nice and important application the association schemes of coding theory are highly recommended (3, 9).

Finite fields

In our discussion of cyclic codes we have already seen that some theory of finite fields is necessary for coding theory. More advanced concepts such as traces turn out to be quite useful, e.g. to describe the Kerdock codes (cf. 9). Sometimes the description of a code makes use of the normal basis theorem (9). Coding theory presents some problems which one would not expect after having learned about finite fields. If C is a code over F_q where $q = p^r$ we can represent F_q as $(F_p)^r$. The field F_q is unique. However, if we substitute the representation for the symbols in the codewords of C then the minimum distance of the resulting code over F_p depends on which representation of F_q we choose. There are open problems in this area.

Group theory

In a course on permutation groups one can include several well-known codes as examples, e.g. the quadratic residue codes with groups PSL (2,n), again an area with open problems. Of course the Golay code is a must in such a course (cf. 3, 9).

Invariant theory

This topic has recently regained popularity. There are several very nice applications in coding theory of which Gleason's theorem on weight enumerators of self-dual codes is probably the most important one (see 9, Ch. 19).

Polynomials

The Reed-Muller codes which were treated in the previous section can also be defined using polynomials. This leads to theorems on the number of zeros of polynomials over finite fields (5). The reader who is familiar with the theorems of Chevalley, Warning and Ax in this area can find an application to RM codes in (6). Questions on irreducible polynomials clearly play a role in the description of finite fields. A very important application is connected to the Justesen codes (7).

Number theory

Elementary number theory is used in the treatment of arithmetic codes (7, 11). There is a proof of the distance properties of RM codes which uses Lucas' theorem (7). Several non-existence theorems for perfect codes are nice applications of number theoretic arguments (cf. 5, 6, 7, 8, Ch. 6). The Carlitz-Uchiyama bound shows an application of A. Weil's famous results on the Riemann Hypothesis in function fields (5, 9). For a course of modular functions the connection with self-dual codes as described by Sloane in (12) is highly recommended.

Orthogonal polynomials

There are several places in coding theory where the Krawtchouk polynomials turn up. Many theorems depend heavily on results which are usually treated in a standard course on orthogonal polynomials. Perhaps the best illustration is the proof by McEliece, Rodemich, Rumsey and Welch of what is the strongest upper bound presently known for codes with prescribed length and distance (cf. 9, Ch. 17). The strongest non-existence results for perfect codes also heavily rely on properties of Krawtchouk polynomials (2, 6, 7). An elementary proof of Lloyd's theorem (cf. 4) is another nice application, in this case of recurrence relations.

Fourier theory

As an application of general ideas from Fourier theory one can treat one of the most famous theorems of coding theory, namely MacWilliams' theorem on weight enumerators. Another nice example is the so-called Mattson-Solomon polynomial which provides very elegant proofs of a number of results on weights (e.g. Goppa codes). These results can be found in the standard text books on coding theory. In our introduction we mentioned the Mariner 1969 mission. The decoding of the RM code which was used shows a good application of the Fast Fourier Transform (Chapter by E.C. Posner in 10).

Finite geometries

Although finite geometries are certainly interesting in their own right I believe that one can say that the increasing popularity of the subject is partly due to many applications in coding theory. We briefly looked at RM codes. For several other applications we refer to (3) and (9). In these references one can also find examples of "applications" in the other direction!

Linear programming

This area clearly does not lack application. We mention it because it may be nice to show how this subject can be applied "theoretically", without any actual

computing. For this we refer to the upper bound mentioned above under orthogonal polynomials.

Most of this section is rather sketchy but that probably does not matter. If I have succeeded in convincing the reader that coding theory is full of elegant applications of many areas of mathematics he will surely consult one of the references and find much more than could be discussed in these pages.

References

1. E.R. Berlekamp, Algebraic Coding Theory (McGraw-Hill, New York, 1968).
2. M.R. Best, On the existence of perfect codes, Thesis, University of Amsterdam, 1981.
3. P.J. Cameron and J.H. van Lint, Graphs, Codes and Designs, Cambridge University Press, 1980.
4. D.M. Cvetkovic and J.H. van Lint, An Elementary Proof of Lloyd's Theorem, Proc. Kon, Ned. Akad. v. Wet. A 80 (1977).
5. J.H. van Lint, Coding Theory (Springer, New York, 1971).
6. J.H. van Lint (ed.), Inleiding in de Coderingstheorie (M.C. Syllabus 31, Amsterdam 1976).
7. J.H. van Lint, Introduction to Coding Theory (Springer Verlag, 1981).
8. J.H. van Lint and F.J. MacWilliams, Generalized Quadratic Residue Codes, IEEE Trans. on Inf. Theory II 24 (1978), 730-737.
9. F.J. MacWilliams and N.J.A. Sloane, The Theory of Error-correcting codes (North Holland, Amsterdam 1977).
10. H.B. Mann (ed.), Error-correcting Codes (John Wiley & Sons, New YUork, 1968).
11. W.W. Peterson, Error-correcting Codes (M.I.T. Press, Cambridge Press, 1961).
12. N.J.A. Sloane, Binary Codes, Lattices and sphere-packings, in Combinatorial Surveys (P.J. Cameron, ed.), (Academic Press, London, 1977).
13. H.C.A. van Tilborg, Uniformly packed Codes, Thesis, Eindhoven University of Technology (1976).

Introductions to Coding

a. E.F. Assmus, Jr., and H.F. Mattson, Jr., Coding and Combinatorics, SIAM Review 16 (1974), 349-388.
b. E.R. Berlekamp, A Survey of Algebraic Coding Theory, CISM Courses and Lectures 28, Springer Verlag, 1970.
c. N. Levinson, Coding Theory, A Counterexample to G.H. Hardy's Conception of Applied Mathematics, Am. Math. Monthly 77 (1970), 249-258.
d. J.H. van Lint, see References (5), (6), (7)
e. N.J.A. Sloane, A Short Course on Error Correcting Codes, CISM Courses and Lectures 188, Springer Verlag 1975.
f. N.J.A. Sloane, Error-correcting Codes and Invariant Theory: New Applications of a Nineteenth Century Technique, Am. Math. Monthly 84 (1977) 82-107.
g. J. Swoboda, Codierung zur Fehlerkorrektur und Fehlererkennung, Oldenbourg Verlag 1973.

10.2 COMBINATORICS

REFLEXIONS SUR L'INTRODUCTION DE LA COMBINATOIRE DANS L'ENSEIGNEMENT MATHEMATIQUE

Nicolas C. Balacheff
Equipe de Recherche Pedagogique en Mathematique
Grenoble, France

II - Introduction

La combinatoire connait depuis deux decennies une developpement particulierement important. Dans une acception naive elle regroupe les problemes d'enumeration, de denombrement, de classement.

En s'appuyant sur la notion de configuration (disposition d'objets en respectant certaines contraintes) C. Berge (1968) enumere les principaux aspects de la combinatoire: etude d'une configuration connue; recherche de configurations inconnues; enumeration de configurations; denombrement approximatif lorsque la complexite ou la nature des configurations ne permet pas leur denombrement exact (on cherche dans ce cas des inegalites ou des ordres de grandeur); optimisation.

Dans la premiere partie (II et III) de cet expose nous envisagerons le probleme general de l'introduction de la combinatoire dans l'enseignement des mathematiques.

Dans la seconde partie (IV) nous decrirons, a partir de recherches que nous conduisons actuellement, les comportments spontanes d'eleves de 10 et 16 ans pour resoudre des problemes de denombrements.

L'introduction d'un nouvel objet d'enseignement peut s'appuyer sur trois types de considerations: ses apports culturel, formatif, utilitaire a la formation des individues (Kuntzmann 1976).

L'importance croissante des problemes combinatoires dans le monde scientifique et industriel contemporain. Le developpement de la combinatoire abstrate en tant que science (theorie des graphes, recherche operationnelle) donnent un poids evident aux arguments utilitaires et meme culturels.

Dans cet expose, j'aborderai les arguemnts du second type en essayant de montrer ce que la combinatoire peut apporter a la formation mathematique de l'individu.