Permutation codes

Thomas Ericson

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

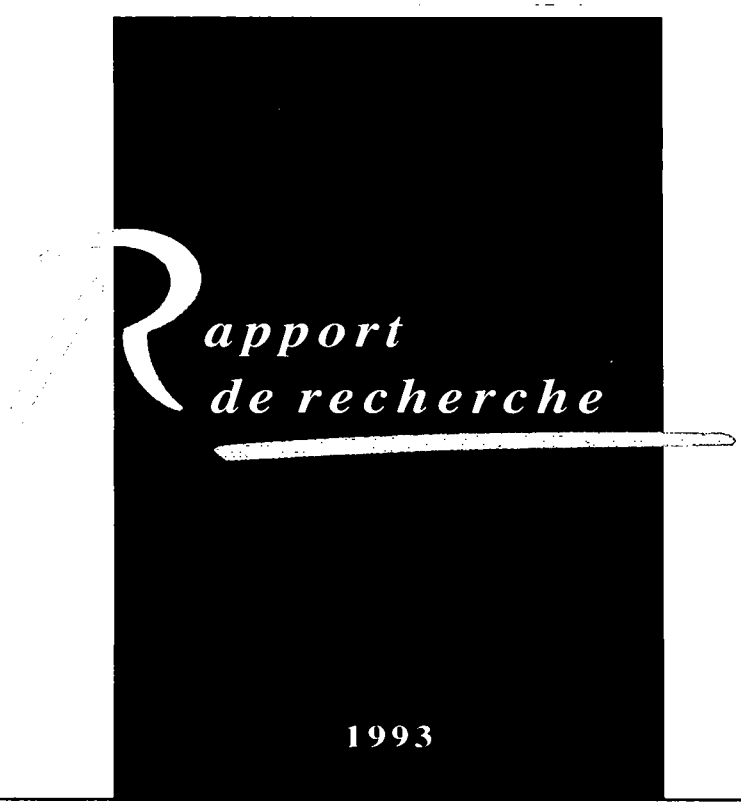
Permutation Codes

Thomas ERICSON

N° 2109
Novembre 1993

PROGRAMME 2

Calcul symbolique,
programmation
et génie logiciel



Codes de permutation
Permutation codes

THOMAS ERICSON [1]

Résumé

Les codes de permutation de Slepian sont examinés en détail. Notamment nous proposons une optimisation du vecteur initial et déduisons tous les codes dominants en dimension $n \leq 6$. A l'exception du code simplex et des codes biorthogonaux – qui sont toujours des cas particuliers de codes de permutation – il n'existe probablement pas de bons codes de dimension supérieure.

Abstract

Slepians permutation codes are investigated in detail. In particular we optimize the initial vector and derive all dominating codes in dimension $n \leq 6$. With the exception of the simplex and biorthogonal codes – which are always included as special cases of permutation codes – there are probably no further good codes in higher dimensions.

keywords: spherical codes, permutation, modulation, optimization of initial vector.

---

[1]Department of Electrical Engineering (ISY), University of Linköping, S-58183 Linköping, SWEDEN. **Invité au projet CODES du 1/9/93 au 30/7/94**

1. Introduction

Permutation modulation is one of the oldest principles for generating spherical codes. We will refer to the resulting codes as permutation codes. They were presented by Slepian in 1965, [1], and have since then been studied by many authors. The basic idea is very simple and comes in two versions. Take an arbitrary realvalued unit norm vector $x_0 = (x_1, x_2, \ldots, x_N)$ and form all possible permutations of it. The set of vectors so generated is obviously a spherical code. The construction is known as permutation modulation variant 1. If in addition all possible sign changes are also allowed the construction is referred to as permutation modulation variant 2. Both versions produce codes with a number of attractive properties: they are easy to characterize, they are easy to analyze and they can easily be implemented. Most remarkable of all: maximum likelihood decoding can be performed with the aid of a very simple algorithm!

For small dimensions the class of permutation codes containes some very good codes. Unfortunately, however, performance generally deteriorates quickly with increasing dimension. For n larger than say 6-8 no interesting codes are produced (with the exception of the simplex and biorthogonal codes, which are always generated as special cases).

We give a detailed discussion of the various versions, paying special attention to the problem of optimizing the initial vector. We also present tables of the best codes generated by this principle.

2. Variant 1

Let the initial vector $\mathbf{x}_0$ be of the form

$$\mathbf{x}_0 = (\mu_1, \mu_1, \ldots, \mu_1, \mu_2, \ldots, \mu_2, \mu_3, \ldots, \mu_K, \ldots, \mu_K)$$

where $\mu_1 < \mu_2 < \ldots < \mu_K$ are given real numbers and where the number μ_1 occurs m_1 times, μ_2 occurs m_2 times etc. We collect the integers m_i in a vector $m = (m_1, m_2, \ldots, m_K)$ which we will refer to as the **distribution**.

We notice that the codewords $\mathbf{x}$ are described as vectors with $N = m_1 + m_2 + \ldots + m_K$ components. We point out right away that this doesn't mean that the dimension is equal to N. As will be seen shortly variant 1 codes normally have dimension n=N-1.

We also notice that all codewords have the same energy given by

$$E = \sum_{i=1}^{K} m_i \mu_i^2 .$$

Hence, by proper scaling the code becomes a spherical code. The number of codewords is

$$M = \frac{(m_1 + m_2 + \ldots + m_K)!}{m_1!\, m_2! \ldots m_K!}$$

and the minimum squared distance is

$$d_E^2 = \min_{i \neq j} \{ 2(\mu_i - \mu_j)^2 : i, j = 1,2, \ldots, K;\ i \neq j \}.$$

The main parameters of a spherical code are n, ρ and M, where n and M are as above and where $\rho \triangleq d_E^2 / E$ is the normalized squared distance. For a given initial vector $\mathbf{x}_0$ the formulas above are sufficient for deriving these parameters. However, without a proper choice of the initial vector $\mathbf{x}_0$ we cannot expect to obtain any good codes. Therefore, let us see how $\mathbf{x}_0$ should be chosen in order to generate codes with favourable parameters.

Without loss of generality we prescribe that the minimum distance between different levels m_j should be 1. That fixes the value of d_E^2 to 2. For a given value of d_E^2 we now want to minimize the energy E. It is clear that we could only loose energy by letting the distance between any pair of adjacent m_j:s exceed the minimum. Hence, without loss of generality we may assume

$$\mu_j = j - a; \quad j=1,2,\ldots,K. \tag{1}$$

It remains to determine the constant a and to choose the best distribution m. We consider first the selection of a and begin with a few definitions.

Def.: $A_m = \sum_{i=1}^{K} m_i\, i^2$

$$B_m = \sum_{i=1}^{K} m_i\, i.$$

We get

$$E = \sum_{i=1}^{K} m_i\,(i-a)^2 = A_m - 2aB_m + Na^2$$

$$= N\left(a - \frac{B_m}{N}\right)^2 + A_m - \frac{B_m^2}{N}.$$

Let us define

$$a_m = \frac{B_m}{N};$$

$$V(m) = A_m - \frac{B_m^2}{N}.$$

Choosing $a=a_m$ gives us

$$E = V(m) = A_m - \frac{B_m^2}{N}.$$

It is clear that this is the choice of a that minimizes E. The remaining problem is how to select the distribution vector $m = (m_1, m_2, \ldots, m_K)$.

We notice that any permutation of the components m_i in m only affects the energy E but leaves the cardinality M unchanged. As a first step in our optimization with respect to m it is therefore natural to consider optimization over the set of all permutations of a given distribution $m = (m_1, m_2, \ldots, m_K)$. A set T of that kind is usually referred to as a **type**. Let a type T be given and let us consider the problem of finding the best vector m in T. The key to the solution of this problem is given by the following lemma.

Lemma 1: Let a type T be given and suppose that $m \in T$ minimizes V(m), i.e.

$$V(m) = \min \{ V(m') : m' \in T \}. \tag{2}$$

Then the following implication holds:

$$(i-a_m)^2 > (j-a_m)^2 \implies m_i \leq m_j . \tag{3}$$

Proof: Suppose $m \in T$ doesn't satisfy (3), i.e. suppose there exists a pair i, j of indices such that

$$\varepsilon = (i-a_m)^2 - (j-a_m)^2 > 0$$

$$\Delta = m_i - m_j > 0 .$$

Define $m^* \in T$ as follows:

$$m_k^* = \begin{cases} m_k & k \neq i, j \\ m_i & k = j \\ m_j & k = i \end{cases} .$$

We have

$$D(m^*, a_m) = \sum_{k=1}^{K} m_k^* (k-a_m)^2$$

$$= \sum_{k=1}^{K} m_k (k-a_m)^2 + (m_i^* - m_i)(i - a_m)^2 + (m_j^* - m_j)(j - a_m)^2$$

$$= V(m) - \Delta \varepsilon .$$

But we also have

$$D(m^*,a_m) = V(m^*) + N(a_m - a_{m^*})^2.$$

If follows that we must have $V(m^*) < V(m)$, so (2) cannot be satisfied.

■

We notice next that (3) essentially determines the $m \in T$ minimizing V . Let T and $t = (t_1, t_2, \ldots, t_K) \in T$ with $t_1 \le t_2 \le \ldots \le t_K$ be given and let m be a distribution minimizing V over T. The corresponding mean is denoted a_m . Let us consider how t should be permuted in order to generate m.

The obvious procedure is illustrated in Fig.1. It is clear that t_K should be closest to a_m. Without loss of generality we assume that it is to the left of a_m. The next symbol t_{K-1} is then immediately to the right of a_m. Continuing the argument we find that t_{K-2} must be just to the left of t_K , t_{K-3} must be just to the right of t_{K-1} etc. In summary we have that

$$m = \begin{cases} (t_1,t_3,\ldots,t_K,t_{K-1},\ldots,t_4,t_2) & K \text{ odd} \\ (t_2,t_4,\ldots,t_K,t_{K-1},\ldots,t_3,t_1) & K \text{ even} \end{cases} \qquad (4)$$

is a possible choice for an $m \in T$ minimizing V.

The choice in not unique. We could equally well have put t_K to the right of a_m . Further, whenever an optimal distribution happens to be such that a_m is just halfway between t_K and t_{K-1} there are further alternative choices. However, it is easy to see that they all lead to the same value of V. In the sequel we will therefore refer to (4) as the optimal distribution, implicitly understanding that it is never unique but that all alternative optimal distributions differ only in a trivial way.

It is worth emphasizing that for a given type T the above argument leads to the optimal distribution m without prior knowledge of the mean a_m . This is important, because in order to compute a_m we need of course know the distribution m, i.e. the quantity we are looking for.

It turns out that for small dimensions n different types T produce different spherical codes and that they are all incompatible, i.e. none of them dominates any other. Therfore, for small dimensions we have actually carried the optimization as far as it is possible to go. In other words: in order to produce all dominating codes we need to consider all different types T.

For large dimensions this is no longer true: for large dimensions only some of the types produce dominating codes. The optimal distributions are those approximating Gaussian distributions, see Slepian [1] and Ingemarsson [3]. However, as we have already pointed out, for large dimensions the permutation codes unfortunately do not offer any good performance.

The first example of a type not producing a dominating code is the one containing $m = (1,4,1,1)$, which is dominated by the one containing $m = (2,3,2)$. This occurs in dimension $n=6$.

All optimal codes satisfy (1) with $a=a_m$, i.e. each codeword $\mathbf{x}$ satisfies

$$\sum_{i=1}^{N} x_i = \sum_{k=1}^{K} m_k (k - a_m) = B_m - Na_m = 0.$$

It follows that although specified by N coordinates the optimal codes are actually located in an (N-1)-dimensional subspace. It turns out that the dimension is always exactly N-1, i.e there is no even smaller subspace that contains any of these codes. For individual codes this is usually easy to establish. A general proof is given in Appendix 1.

We summarize the properties we have derived for the variant 1 permutation codes:

Dimension: $n = m_1 + m_2 + \ldots + m_K - 1$

Normalized distance: $\rho = \dfrac{2N}{NA_m - B_m^2}$

Cardinality: $M = \dfrac{(m_1 + m_2 + \ldots + m_K)\,!}{m_1!\, m_2! \ \ldots \ m_K!}$.

The parameters A_m , B_m and N are obtained as

$$A_m = \sum_{i=1}^{K} m_i \, i^2$$

$$B_m = \sum_{i=1}^{K} m_i \, i$$

$$N = m_1 + m_2 + \ldots + m_K \, .$$

Finally, the alphabet is

$$\mathcal{L}_q = \{ \, i - a_m : i=1, 2, \ldots, q \, \}$$

with q=K. We notice that in general the alphabet is unsymmetric.

3. Best variant 1 codes

In Table 1 we have listed all dominating permutation codes of variant 1 for dimensions 1 $\leq n \leq 6$. That actually covers all cases of interest: for $n > 6$ we do generally obtain better codes from other constructions.

The table gives dimension n, normalized squared distance ρ, and cardinality M. For each code we have also indicated the vector m generating the code. As can be seen, for each n in the range considered all partitions of N=n+1 produce dominating codes, with the distribution m = (1,4,1,1) mentioned above as the only exception. On the other hand, the discussion in Section 2 assures that by considering one representative in each possible partition we can be sure of obtaining all dominating codes.

We have indicated with a star (*) all cases where the codes are known to be optimal and with (B) all other cases where we aren't aware of any better code.

For n=1 we have just a fancy representation of the binary antipodal code. In dimension n=2 we have the equilateral triange and the regular hexagon. All of these are optimal, although they represent rather trivial codes. We notice that both the triangle and the hexagon are described without using radicals, but to the expence of using one extra coordinate.

In tree dimensions m = (2,2) gives a nice description of the biorthogonal code (which in general is not included in this construction) and m = (1,2,1) offers an alternative description of the cubeoctahedron. The distribution m = (1,1,1,1) generates the semiregular polyhedron with 6 squares and 8 hexagons as faces. It is not in itself a particularly good spherical code. However, adding one point just outside the center of each hexagon produces a spherical code with parameters (n, ρ, M) = (3, 2/5, 32), which although not optimum is reasonably good (for n=3, M=32 the best known code has ρ=0.4125).

The distribution m = (n,1) provides the standard representation of the simplex code. Thus, all simplex codes are included in the construction. These codes, of course, are optimal for any value of n .

Five codes are marked with a B, indicating that they represent best constructions known to us. They occur in dimensions 4, 5 and 6. The three codes in dimensions 4 and 6 have unsymmetric alphabets, while those in dimension 5 have symmetric alphabets. The two

binary codes - m = (3,2) and m = (3,3) - offer, of course, efficient representations on the unit sphere of corresponding Johnson spaces.

Apart from the cases mentioned there seem to be very few good ones among the variant 1 codes. We have performed a complete search in the range $n \leq 24$. In that range there is none. Except for the simplex codes it is very unlikely that there are any further good codes for larger n.

The problem of finding the best vector $\mathbf{x}_0$ is usually referred to as the initial value problem (Slepian [6], p 582, see also Ingemarsson [3] and Karlof [4]) . The first few observations in Section 2 reduce this problem to that of finding the best distribution m. Lemma 1 addresses this latter part of the problem. It is worthwhile emphasizing that although the lemma doesn't fully solve the problem it quite substantially reduces the number of distributions we need to consider. Suppose we want to find all dominating codes in dimension $n=N-1$. The total number of distributions $m = (m_1, m_2, \ldots, m_K)$ with $m_1 + m_2 + \ldots + m_K = N$ is 2^{N-1} , while the number of types equals the number $p(N)$ of partitions of $N=n+1$. That number is much smaller. Hall [8] gives a table (p 38) containing the first 100 values of $p(N)$ and also offers the following asymptotic formula (p 44):

$$p(N) \approx \frac{1}{4N\sqrt{3}} e^{A\sqrt{N}}$$

where A is a constant. As an example, for $n=24$ we have $p(N) = p(25) = 1958$, which should be compared to $2^{24} = 16\,777\,216$.

The computation of the list of dominating codes was based on the results from the previous section, which means that for each type we need to consider just one distribution. However, for each dimension n we have to generate all partitions of $N=n+1$. An efficient method for this is presented in the appendix.

For n larger than 24 the above method becomes rather time consuming and more efficient methods are desirable. Karloff [4] has developed an algorithm for directly determining an optimal initial vector $\mathbf{x}_0$. According to Ingemarsson [3] the best distributions m are approximately Gaussian, a result apparently anticipated already by Slepian [1]. However, as we have already mentioned - and as was observed by Slepian ([1], p 233) - the resulting codes usually have poor performance.

4. Variant 2a

In variant 2 of the permutation codes we allow, in addition to all permutations, also all possible sign changes in the components of the initial vector $\mathbf{x}_0 = (\mu_1, \mu_1, \ldots, \mu_1, \mu_2, \ldots, \mu_2, \mu_3, \ldots, \mu_K, \ldots, \mu_K)$. Clearly we may assume, without loss of generality, that all components μ_i are non-negative. It turns out to be convenient to distinguish between two cases: the case $\mu_1 = 0$ - which we will refer to as variant 2a - and the case $\mu_1 \neq 0$ - which we will call variant 2b. We consider case 2a first.

It is convenient to introduce an additional notation so as to distinguish between the zero symbols and the other symbols. Let the distribution m be of the form m=(r,p), where $r=m_1$ and $p = (p_1, p_2, \ldots, p_J) = (m_2, m_3, \ldots, m_K)$. The size of a variant 2a code with distribution m=(r,p) is (with J = K-1)

$$M = \frac{(r+ p_1+ p_2+ \ldots + p_J)!}{r!\ p_1!\ p_2! \ldots p_J!}\ 2^{(p_1+ p_2 + \ldots + p_J)}. \tag{5}$$

Again without loss of generality we prescribe that the minimum distance between any pair of adjacent letters μ_j and μ_{j+1} should be 1. As before this fixes the minimum squared distance to 2. As we have already prescribed the condition $\mu_1 = 0$ we obviously have in this case

$$\mu_j = j - 1; \quad j = 1, 2, \ldots, K;$$

corresponding to an alphabet $\mathcal{L}_q$ of the form

$$\mathcal{L}_q = \{ 0, \pm 1, \pm 2, \ldots, \frac{q-1}{2} \}.$$

The size q of the alphabet is always odd: q = 2J+1 = 2K-1. By Appendix 1 the dimension is

$$n = r + p_1+ p_2 + \ldots + p_J .$$

The energy E , finally, is given by the formula

$$E = \sum_{j=1}^{J} p_j\, j^2. \tag{6}$$

All that remains is to determine the distributions m=(r,p) that lead to dominating codes.

From (5) and (6) we see that for fixed r the size M is constant over the type T(p) while the energy is not. It follows immediately that p should be chosen such that $p_1 \geq p_2 \geq \ldots \geq p_J$.

The selection of r is less obvious. It is easy to see that for fixed $p = (p_1, p_2, \ldots, p_J)$ the size M is an increasing function of r. However, the dimension $n = r + p_1 + p_2 + \ldots + p_J$ increases also with r, so different values of r give rise to codes of different dimensions.

In Table 2 we have listed all the dominating codes from variant 2a in the range $2 \leq n \leq 6$. Again this range contains practically all codes of good quality. The case n=1 doesn't occur in variant 2a. In dimension n=2 we have only the square. In dimension n=3 we have alternative representations of the cubeoctahedron (3,1,12) and the semiregular polyhedron (3,2/5,24). Both were generated also in variant 1. For larger n we obtain a number of new codes. Especially in dimension n=5 we have several good codes.

The distrubutions m = (r,p) = (n-1,1) give the standard representation of the biorthogonal codes; they are all included in version 2a.

We notice that in the range $n \leq 6$ all possibe values of r and most of the possible partitions p are represented. As for variant 1 there are very few good codes from variant 2a for $n \geq 6$.

5. Variant 2b

In variant 2b we have $0 < \mu_1 < \mu_2 < \ldots < \mu_K$. Here we have to be a little bit more careful when determining the signal levels μ_i . Because of the sign-changes there are two ways of generating nearest neighbours to a given word: either by permuting a pair of coordinates (as before) or (and this is new) by changing sign in one of the positions containing μ_1. A moment of thought reveals the following formula:

$$d_E^2 = \min \{ 4\mu_1^2, \min_{i \neq j} 2 (\mu_i - \mu_j)^2 \}.$$

A simple calculation reveals that under the constraint $d_E^2 = 2$ the energy is minimized by

$$\mu_i = \frac{1}{\sqrt{2}} + i - 1; \quad i = 1, 2, \ldots, K .$$

The corresponding alphabet is

$$\mathcal{L}_q = \{ \pm \frac{1}{\sqrt{2}} , \pm (\frac{1}{\sqrt{2}} + 1), \ldots, \pm (\frac{1}{\sqrt{2}} + K - 1) \}.$$

Obviously the size is even: $q = 2K$.

For codes in variant 2b the size is

$$M = \frac{(m_1 + m_2 + \ldots + m_K)!}{m_1 ! \, m_2 ! \ldots m_K !} \; 2^{(m_1 + m_2 + \ldots + m_K)}$$

and the dimension n is (again, see Appendix 1)

$$n = m_1 + m_2 + \ldots + m_K .$$

Finding the best distributions is simple. Clearly we should have $m_1 \geq m_2 \geq \ldots \geq m_K$.

In Table 3 we display the dominating codes for $n \leq 6$. As before this list contains practically all interesting codes. It contains several quite good codes, but none for which we are able to prove optimality (except for the trivial codes in dimensions 1 and 2).

In dimension n=3 we notice that we have a better code of size 24 than we had in the two previous versions. The new code is the semi-regular polyhedron with 8 triangles and 18 squares as its faces. The minimum squared distance is $\rho = 0.51$, which is considerably better than we had before and reasonably close to the optimal value of 0.55.

6. Decoding

One of the most fashinating properties of permutation codes is the fact that they can be optimally decoded with a very simple algorithm. Let a permutation code $S \in \Omega_n$ be given. By decoding we understand in this context the following problem.

Problem: for any given vector $\mathbf{r} = (r_1, r_2, \ldots, r_N) \in \Re^N$ find a vector $\mathbf{x} = (x_1, x_2, \ldots, x_N) \in S$ with minimal distance to $\mathbf{r}$.

Alternatively we could, of course, try to find a vector $\mathbf{x}$ having maximal correlation with $\mathbf{r}$. We will use this second formulation.

An algorithm solving the problem is often referred to as a maximun likelihood decoding algorithm. This refers to decoding in the presence of additive white Gaussian noise.

We consider first codes from variant 1. The crucial observation is the following lemma.

Lemma 2: Let S1 be a permutation code of variant 1 and let $\mathbf{x} \in S1$ satisfy

$$(\mathbf{x}, \mathbf{r}) = \max \{(\mathbf{x}', \mathbf{r}) : \mathbf{x}' \in S1\}. \tag{7}$$

Then the following implication holds:

$$r_i < r_j \Rightarrow x_i \leq x_j \tag{8}$$

Proof: The form of the proof is the same as in Lemma 1. Suppose $\mathbf{x} \in S1$ violates (8), i.e. suppose that for some pair i, j we have both $r_i < r_j$ and $x_i > x_j$. Define

$$y_k = \begin{cases} x_k & k \neq i, j \\ x_i & k = j \\ x_j & k = i \end{cases}.$$

Clearly $\mathbf{y} = (y_1, y_2, \ldots, y_N) \in S1$. We have

$$(\mathbf{y}, \mathbf{r}) - (\mathbf{x}, \mathbf{r}) = (y_i - x_i) r_i + (y_j - x_j) r_j$$

$$= (x_i - x_j)(r_j - r_i) > 0.$$

It follows that $\mathbf{x}$ cannot maximize $(\mathbf{x}, \mathbf{r})$.

■

The lemma immediately leads to the following decoding algorithm. Let the initial vector be $\mathbf{x_0} = (a_1, a_2, \ldots, a_N)$ with $a_1 \leq a_2 \leq \ldots \leq a_N$. Decoding can be thought of as assigning a permutation of $\mathbf{x_0}$ to each vector $\mathbf{r} = (r_1, r_2, \ldots, r_N) \in \Re^N$. From the lemma follows that by assigning a_1 to the smallest component in $\mathbf{r}$, a_2 to the second smallest and so on we can be sure to obtain a vector $\mathbf{x} \in S1$ with a minimal distance to the given vector $\mathbf{r}$.

A slight modification of the algorithm allows optimal decoding also of codes from variant 2. In this case we need the following lemma.

Lemma 3: Let S2 be a permutation code of variant 2 and let $\mathbf{x} \in S2$ satisfy

$$(\mathbf{x}, \mathbf{r}) = \max \{(\mathbf{x}', \mathbf{r}) : \mathbf{x}' \in S2\}. \tag{9}$$

Then the following implications hold:

$$\text{i)} \quad |r_i| < |r_j| \Rightarrow |x_i| \leq |x_j|, \tag{10}$$

$$\text{ii)} \quad x_i, r_i \neq 0 \Rightarrow \operatorname{sign}(r_i) = \operatorname{sign}(x_i). \tag{11}$$

Proof: First suppose $\mathbf{x}$ violates (11), i.e. assume that for some i we have $x_i, r_i \neq 0$ with $\operatorname{sign}(r_i) \operatorname{sign}(x_i) = -1$. Define

$$y_k = \begin{cases} x_k & k \neq i \\ -x_i & k = i \end{cases}.$$

We obtain

$$(\mathbf{y}, \mathbf{r}) - (\mathbf{x}, \mathbf{r}) = y_i r_i - x_i r_i = -2 x_i r_i > 0.$$

Thus ii) is established. Next suppose $\mathbf{x}$ satisfies ii) but violates i) . That means that we assume that for some pair i, j we have $|r_i| < |r_j|$ and $|x_i| > |x_j|$. First assume $r_i \neq 0$ and define

$$y_k = \begin{cases} x_k & k \neq i, j \\ |x_i| \operatorname{sign}(r_j) & k = j \\ |x_j| \operatorname{sign}(r_i) & k = i \end{cases}.$$

Clearly $y \in S2$. We obtain

$$(\mathbf{y}, \mathbf{r}) - (\mathbf{x}, \mathbf{r}) = (y_i - x_i) r_i + (y_j - x_j) r_j$$

$$= |x_j| \text{sign}(r_i) r_i - |x_i| \text{sign}(x_i) r_i + |x_i| \text{sign}(r_j) r_j - |x_j| \text{sign}(x_j) r_j . \qquad (12)$$

By ii) we have $\text{sign}(x_i) = \text{sign}(r_i)$, which inserted in (12) gives us

$$(\mathbf{y}, \mathbf{r}) - (\mathbf{x}, \mathbf{r}) = |x_j||r_i| - |x_i||r_i| + |x_i||r_j| - |x_j||r_j|$$

$$= (|x_i| - |x_j|)(|r_j| - |r_i|) > 0.$$

If $r_i = 0$ the same inequality is easily established by a slight modification of the argument. Thus i) is established and the theorem is proved.

■

For a code of variant 2 let the initial vector be $\mathbf{x_0} = (a_1, a_2, \ldots, a_N)$ with $0 \le a_1 \le a_2 \le \ldots \le a_N$. By Lemma 3 follows that we obtain optimal decoding by first assigning the values a_i according to increasing values of $|r_i|$ and then choosing $\text{sign}(x_i) = \text{sign}(r_i)$.

We see that in both variants the decoding essentially consists of sorting the components of the vector $\mathbf{r} = (r_1, r_2, \ldots, r_N)$ according to magnitude, an operation that requires $(N-1)N / 2$ comparisons. It is worthwhile noticing that the number of comparisons needed depends only on the dimension n (N-1 in variant 1 and N in variant 2) and **not** on the size M of the code.

7. General comments

It is an ironic fact that one of the nicest properties displayed by the permutation codes - namely the availability of a very simple optimal decoding algorithm - actually is also a property that excludes good asymptotic performance. This follows from an observation by Landau [7]. The observation is the following.

Decoding, as defined in Section 6, of a code X with parameters (n, ρ, M) corresponds to partitioning the Euclidean space $\Re^n$ in M disjoint subregions - one for each codeword . Maximum likelihood decoding corresponds to assigning to each codeword its associated Voronoi region. Clearly the boundaries between the various Voronoi regions are defined by hyperplanes in $\Re^n$. Denote by K the number of hyperplanes needed to specify all the Voronoi regions for the code X and define s = floor (4 / r). Landau showed that as soon as M > 4 the following inequality holds ([7], Th. 2, p160) :

$$K \geq (M / 2)^{1/s}.$$

The simplicity of the decoding algorithms we derived in Section 6 essentially means that the number of hyperplanes involved is small. For codes of variant 1 we have to perform N(N-1) / 2 comparisons. Each determines the position of the given vector **r** with respect to some hyperplane. For codes of variant 2 we also have to determine the sign of each one of the N coordinates in $\mathbf{r} = (r_1, r_2, \ldots, r_N)$. Those decisions also determine the position of **r** with respect to certain hyperplanes, so for variant 2 codes the number of hyperplanes can be estimated as $K \leq N(N-1) / 2 + N$ ([7] p 161). In both cases we have that for fixed ρ - and consequently also fixed s - the size M .of the code X is bounded by $M \leq 2N^{2s}$. Thus, the exponential growth we know is possible for spherical codes in general is excluded for the class of permutation codes.

The above result is well in line with the observations mentioned above regarding the performance for large dimensions and strongly supports our conjecture that essentially no good codes are produced for dimensons larger than 6-8. The simplex and the biorthogonal codes do not provide any counterexamples: although optimal they do not exhibit exponential size.

It is interesting to note that all three of the regular polytopes that generalize to arbitrary dimensions are found among the permutation codes: the simplex codes are all generated by variant 1, the biorthogonal codes are generated by variant 2a, and the hypercubes are generated by variant 2b.

References

[1] D.Slepian, Permutation modulation, Proc. IEEE, Vol. 53, No 3, March 1965, pp 228-236.

[2] D.Slepian, Bounds on communication, Bell Syst. Techn. J., Vol. 42, May 1963, pp 681-707.

[3] I.Ingemarsson, Optimized permutation modulation, IEEE Trans. Inform. Theory, Vol. IT-36, No 5, Sept. 1990, pp 1098-1100.

[4] J.Karlof, Permutation codes for the Gaussian channel, IEEE Trans. Inform. Theory, Vol. IT-35, No 4, July 1989, pp 726-732.

[5] E. Biglieri and M. Elia, Optimum permutation modulation codes and their asymptotic performance, IEEE Trans. Inform. Theory, Vol. IT-22, No 6, Nov. 1976, pp 751-753.

[6] D. Slepian, Group codes for the Gaussian channel, Bell Syst. Techn. J., Vol. 47, No 4, April 1968, pp 575-602.

[7] H.J. Landau, How does a porcupine separate its quills?, IEEE Trans. Inform. Theory, Vol. IT-17, No 2, March 1971, pp 157-161.

[8] M. Hall Jr, Combinatorial theory, Wiley-Interscience Series in discrete Mathematics, Sec. Ed., Wiley 1986.

[9] A.B. Stott, Geometrical deduction of semiregular from regular polytopes and space fillings, Ver. der Koninklijke Akademie van Wetenschappen te Amsterdam (eerstie sectie), Vol. 11, No 1, 1910.

[10] P.H. Schoute, Analytical treatment of the polytopes regularly derived from the regular polytopes, Ver. der Koninklije Akad. van Wetenschappen te Amsterdam (eerstie sectie), Vol 11, No 5, 1913.

[11] I.F. Blake, Permutation codes for discrete channels, IEEE Trans. Inform. Theory, Vol. IT-20, Jan. 1974, pp 138-140.

Appendix 1

We like to establish the dimensionality of the various forms of permutation codes. The crucial result is the following lemma.

Lemma A1: Let $\mathbf{x} = (x_1, x_2, \ldots, x_N) \in \Re^N$ and let $P(\mathbf{x})$ be the linear space spanned by all permutations of $\mathbf{x}$. The dimensionality of $P(\mathbf{x})$ depends on $\mathbf{x}$ in the following way:

$$\begin{aligned}
&\text{A:} \quad \mathbf{x} = 0 && \Rightarrow \quad \dim P(\mathbf{x}) = 0;\\
&\text{B:} \quad \mathbf{x} = \text{const.} \neq 0 && \Rightarrow \quad \dim P(\mathbf{x}) = 1;\\
&\text{C:} \quad \mathbf{x} \neq \text{const.}, \ \sum_{i=1}^{N} x_i = 0 && \Rightarrow \quad \dim P(\mathbf{x}) = N - 1;\\
&\text{D:} \quad \mathbf{x} \neq \text{const.}, \ \sum_{i=1}^{N} x_i \neq 0 && \Rightarrow \quad \dim P(\mathbf{x}) = N .
\end{aligned}$$

Proof: The statement is trivially true when $N = 1$. By induction, suppose it is true for N and consider $\mathbf{x} = (x_1, x_2, \ldots, x_{N+1})$. We will prove the statement by considering each one of the above cases separately. Cases A and B are trivial, so we really only have to consider cases C and D.

Case C: The assuption is that $\mathbf{x} = (x_1, x_2, \ldots, x_{N+1})$ is not constant and that the components sum to zero. Without loss of generality we may assume $x_{N+1} \neq 0$. Consider $\mathbf{x}' = (x_1, x_2, \ldots, x_N)$. It is clear that $\mathbf{x}'$ cannot belong to case A or C. If it belongs to case B we must have that $\mathbf{x}$ is proportional to $(1^N, -N)$. But that means that $P(\mathbf{x})$ is spanned by a simplex, and the dimension is obviously N, as it should.

If $\mathbf{x}'$ belongs to case D by the induction hypothesis $\dim P(\mathbf{x}') = N$. Clearly we have $\dim P(\mathbf{x}) \geq \dim P(\mathbf{x}') = N$. But we also have $\dim P(\mathbf{x}) \leq N$ because of $\sum x'_i = 0$. It follows that we have $\dim P(\mathbf{x}) = N$ in this case too.

Case D: Again without loss of generality we assume $x_{N+1} \neq 0$ and consider $\mathbf{x}' = (x_1, x_2, \ldots, x_N)$. If $\mathbf{x}' = 0$ (belongs to case A) it is obvious that $\dim P(\mathbf{x}) = N+1$. If $\mathbf{x}' =$ const. $\neq 0$ (belongs to case B) we have $\mathbf{x}$ of the form $\mathbf{x} = (a^N, b)$, with $a \neq b$; $a, b \neq 0$; $b \neq -Na$. Alternatively we might write $\mathbf{x}$ in the form $\mathbf{x} = c^{N+1} + ((a-c)^N, b-c)$ and choose c such that $N(a-c) + b - c = Na + b - (N+1)c = 0$, i.e. we may choose

$$c = \frac{Na + b}{N + 1} \quad .$$

Then the vector $((a-c)^N, b-c)$ is proportional to $(1^N, -N)$ and $c \neq 0$. Upon permuting $\mathbf{x}$ the vector $((a-c)^N, b-c)$ will generate a simplex spanning a space of dimension N while the vector c^{N+1} stays constant. By observing that c^{N+1} is orthogonal to all of the vectors in the space spanned by the simplex we conclude that the permutations of $\mathbf{x}$ span a space of dimension N+1.

If $\mathbf{x}'$ belongs to case C we simply delete some other component than x_{N+1} in order to generate $\mathbf{x}'$. There must be some component in $\mathbf{x}$ such that $\sum x'_i \neq 0$, because if all the different sums of this form are zero we might easily conclude $x_1 = x_2 = \ldots = x_{N+1}$ in contradiction to the assumption. Thus, there remains only the case when $\mathbf{x}'$ belongs to case D.

If $\mathbf{x}'$ belongs to case D we may write $\mathbf{x} = (0^N, x_{N+1}) + (\mathbf{x}', 0)$. Consider the subset of all permutations of $\mathbf{x}$ that leave the last component unchanged. By the induction hypothesis those permutations generate a space of dimension N when operating on $(\mathbf{x}', 0)$. Moreover, the vector $(0^N, x_{N+1})$ is unchanged by this subset of permutations and orthogonal to the space generated by $(\mathbf{x}', 0)$. It follows that the space generated in this way has dimension N+1. Clearly the dimension doesn't decrease if we let all permutations operate on $\mathbf{x}$. This observation concludes our argument.

■

It is clear that optimized permutation codes of variant 1 always belong to case C while codes of variant 2 always belong to case D.

Appendix 2

In this appendix we derive an efficient procedure for generating all partitions of a given number n. By a partition of n we understand any representation of the sum $n = x_1 + x_2 + \ldots + x_k$ of k positive integers, where $1 \leq k \leq n$. The number k is called the order of the partition. If $\mathcal{P}(n)$ denotes the set of all partitions of n and if $\mathcal{P}(n,k)$ denotes the set of all partitions of order k it is clear that we have

$$\mathcal{P}(n) = \mathcal{P}(n,1) \cup \mathcal{P}(n,2) \cup \ldots \cup \mathcal{P}(n,n) .$$

We will solve the problem by generating the various subsets $\mathcal{P}(n,k)$. We denote by $P(n,k)$ the matrix having all k-order partitions of n as its rows, taken in reversed lexicographic order. As an example $P(9,4)$ is the matrix

$$P(9,4) = \begin{bmatrix} 6 & 1 & 1 & 1 \\ 5 & 2 & 1 & 1 \\ 4 & 3 & 1 & 1 \\ 4 & 2 & 2 & 1 \\ 3 & 3 & 2 & 1 \\ 3 & 2 & 2 & 2 \end{bmatrix} .$$

By assumption all components in $P(n,k)$ are positive. If we subtract 1 from each component we get a matrix with non-negative components. For the matrix $P(9,4)$ we obtain

$$\begin{bmatrix} 5 & 0 & 0 & 0 \\ 4 & 1 & 0 & 0 \\ 3 & 2 & 0 & 0 \\ 3 & 1 & 1 & 0 \\ 2 & 2 & 1 & 0 \\ 2 & 1 & 1 & 1 \end{bmatrix} .$$

It is clear that this new matrix contains all partitions of $n-k = 9-4 = 5$, with a suffix of zeroes in those rows where the partition of $n-k$ is of order less than k. It is also clear that all partitions of $n-k$ are obtained in this way. Thus, if we denote by $p(n,k)$ the number of rows in $P(n,k)$, i.e. if we let $p(n,k)$ denote the number of partitions of n in k parts, we have the following recursion (cf Hall [8], p 32)

$$p(n,k) = p(n-k,1) + p(n-k,2) + \ldots + p(n-k,m) ,$$

where $m = \min\{k, n-k\}$ and where $n = k+1, k+2, \ldots$. The following initial conditions are obvious:

$$p(n,k) = 0, \quad n = 1, 2, \ldots, k-1$$

$$p(n,k) = 1, \quad n = k .$$

It is clear that we can continue the above process and subtract i ones from each one of the rows representing an i-order partition and then further splitting the resulting new matrix. Any row representing a partition of a number r in $s < r$ parts can be split, so the process can be continued until we have only rows with $r = s$.

We can naturally represent the splitting process with a tree, as shown in Fig.2 for the case $(n,k) = (9,4)$. Each node in the tree is indicated with a number (r,s) and represents the set of all partitions of r in s parts. At the root we have the pair (n,k) . Notice that for each branch the sum of the second components in the pairs of numbers representing the nodes add up to precisely n . Clearly, moving from one node to another neighbouring node at the next lower level corresponds to the subtraction of ones in the P(n,k) matrix as described above. If we finally observe that a node of the form (r,r) corresponds to the trivial partition of r as a sum of r ones, we are led to the following procedure for generating all partitions of n in k parts.

i) Generate the tree obtained by putting (n,k) at the root
and successively applying the splitting rule

$$(r,s) \Rightarrow \{(r-s,1), (r-s,2), \ldots, (r-s,t)\}$$

where $t = \min\{s, r-s\}$.

ii) For each node (r,s) add a binary vector of weight s with the ones located at the beginning of the vector.

The procedure is illustrated in Fig. 2 for the case $(n,k) = (9.4)$.

n	ρ	M	m	
1	4	2	1 1	*
2	1	6	1 1 1	*
2	3	3	2 1	*
3	0.4	24	1 1 1 1	
3	1	12	1 2 1	
3	2	6	2 2	*
3	2.67	4	3 1	*
4	0.20	120	1 1 1 1 1	
4	0.38	60	1 2 1 1	
4	0.71	30	1 2 2	B
4	1	20	1 3 1	
4	1.67	10	3 2	B
4	2.5	5	4 1	*
5	0.11	720	1 1 1 1 1 1	
5	0.20	360	1 1 2 1 1	
5	0.36	180	1 2 2 1	B
5	0.38	120	1 3 1 1	
5	0.50	90	2 2 2	
5	0.71	60	1 3 2	
5	1	30	1 4 1	
5	1.33	20	3 3	B
5	1.5	15	4 2	
5	2.4	6	5 1	*
6	0.07	5040	1 1 1 1 1 1 1	
6	0.11	2520	1 1 2 1 1 1	
6	0.18	1260	1 1 2 2 1	
6	0.20	840	1 1 3 1 1	
6	0.27	630	2 2 2 1	
6	0.35	420	1 3 2 1	
6	0.50	210	2 3 2	
6	0.58	140	1 3 3	
6	0.70	105	1 4 2	B
6	1.00	42	1 5 1	
6	1.17	35	4 3	
6	1.40	21	5 2	
6	2.33	7	6 1	*

Table 1 All dominating permutation codes of variant 1, $1 \leq n \leq 6$.

n	ρ	M	r	p	
2	2	4	1	1	*
3	0.40	24	1	1 1	
3	1.00	12	1	2	
3	2.00	6	2	1	*
4	0.14	192	1	1 1 1	
4	0.33	96	1	2 1	
4	0.40	48	2	1 1	
4	0.67	32	1	3	
4	1.00	24	2	2	B
4	2.00	8	3	1	*
5	0.07	1920	1	1 1 1 1	
5	0.13	960	1	2 1 1	
5	0.20	480	1	2 2	B
5	0.29	320	1	3 1	B
5	0.33	240	2	2 1	B
5	0.67	80	2	3	
5	1.00	40	3	2	B
5	2.00	10	4	1	*
6	0.04	23040	1	1 1 1 1 1	
6	0.06	11520	1	2 1 1 1	
6	0.11	5760	1	2 2 1	
6	0.12	3840	1	3 1 1	
6	0.13	2880	2	2 1 1	
6	0.18	1920	1	3 2	
6	0.20	1440	2	2 2	
6	0.29	960	2	3 1	B
6	0.33	480	3	2 1	
6	0.50	240	2	4	
6	0.67	160	3	3	
6	1.00	60	4	2	
6	2.00	12	5	1	*

Table 2. All dominating permutation codes variant 2a, $2 \leq n \leq 6$.

n	ρ	M	m	
1	4	2	1	*
2	0.59	8	1 1	*
2	2.00	4	2	*
3	0.19	48	1 1 1	
3	0.51	24	2 1	
3	1.33	8	3	
4	0.08	384	1 1 1 1	
4	0.18	192	2 1 1	B
4	0.29	96	2 2	
4	0.45	64	3 1	B
4	1.00	16	4	
5	0.04	3840	1 1 1 1 1	B
5	0.08	1920	2 1 1 1	
5	0.14	960	2 2 1	
5	0.17	640	3 1 1	B
5	0.27	320	3 2	
5	0.41	160	4 1	B
5	0.80	32	5	
6	0.03	46080	1 1 1 1 1 1	
6	0.04	2304	2 1 1 1 1	
6	0.07	11520	2 2 1 1	
6	0.08	7680	3 1 1 1	
6	0.09	5760	2 2 2	
6	0.14	3840	3 2 1	B
6	0.16	1920	4 1 1	
6	0.20	1280	3 3	
6	0.26	960	4 2	
6	0.37	384	5 1	B
6	0.67	64	6	

Table 3. All dominating permutation codes of variant 2b, $1 \leq n \leq 6$.

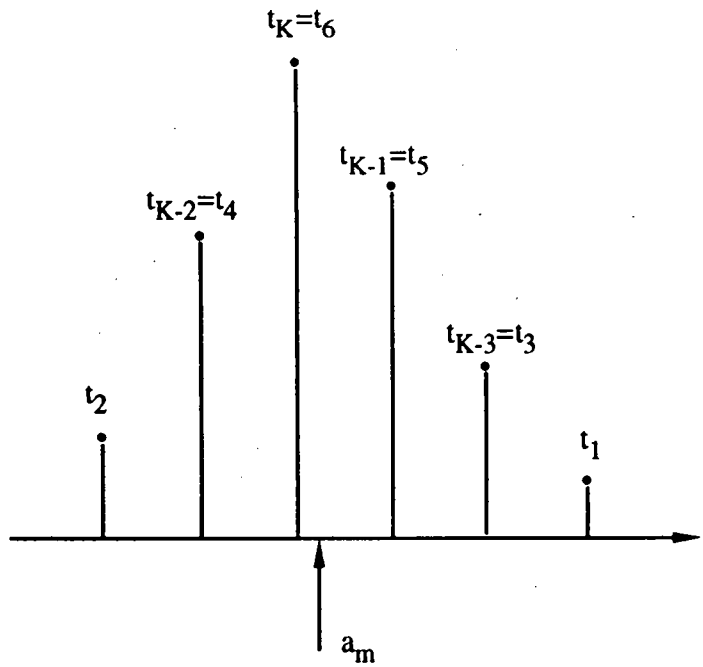


Figure 1 Permuting t = (1, 2, 4, 7, 8, 10) in order to obtain the optimal distribution m = (2, 7, 10, 8, 4, 1).

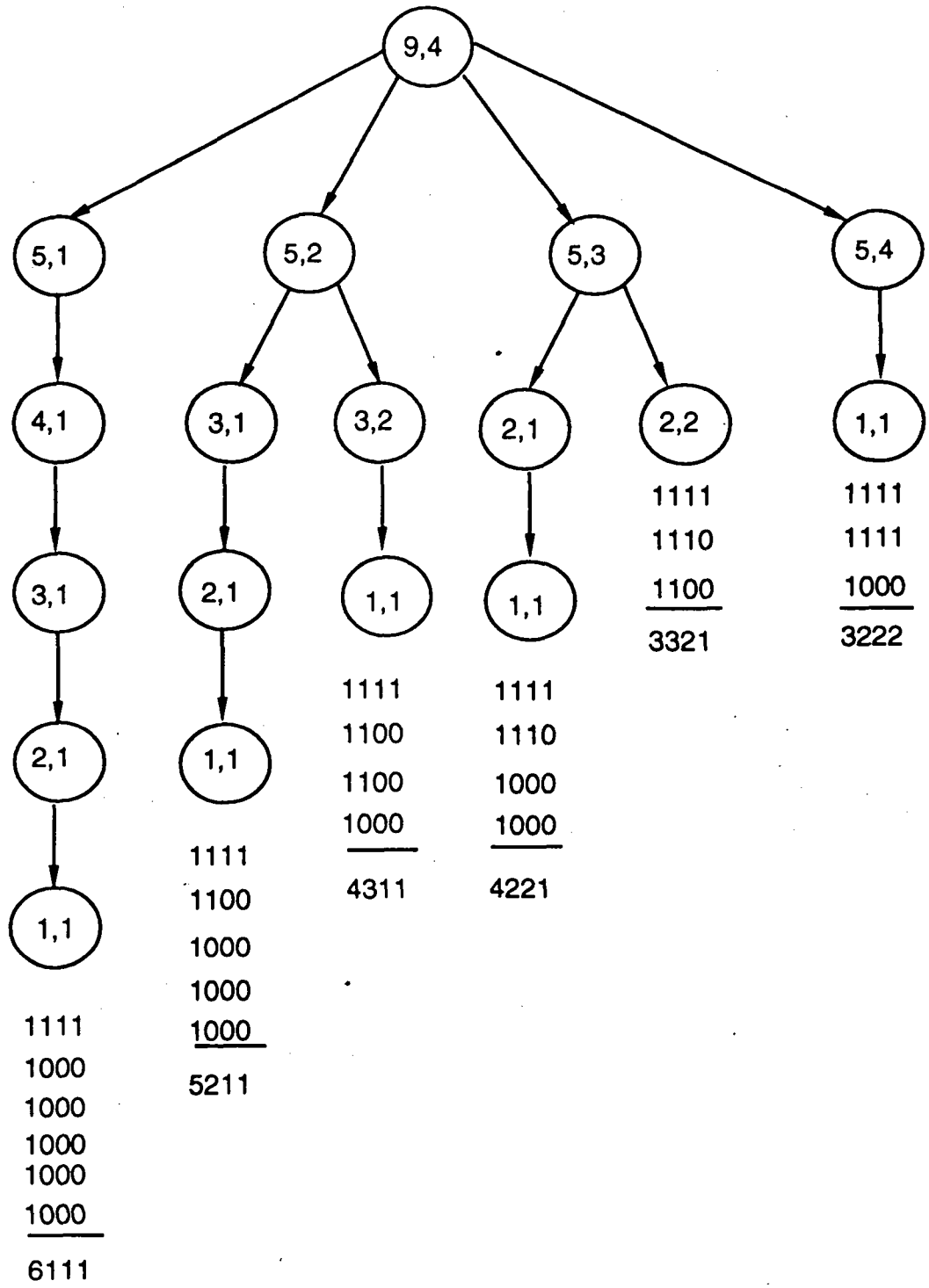


Figure 2 A tree diagram for deriving all 4-partitions of the number 9.

Unité de Recherche INRIA Rocquencourt
Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 LE CHESNAY Cedex (France)

Unité de Recherche INRIA Lorraine Technopôle de Nancy-Brabois - Campus Scientifique
615, rue du Jardin Botanique - B.P. 101 - 54602 VILLERS LES NANCY Cedex (France)
Unité de Recherche INRIA Rennes IRISA, Campus Universitaire de Beaulieu 35042 RENNES Cedex (France)
Unité de Recherche INRIA Rhône-Alpes 46, avenue Félix Viallet - 38031 GRENOBLE Cedex (France)
Unité de Recherche INRIA Sophia Antipolis 2004, route des Lucioles - B.P. 93 - 06902 SOPHIA ANTIPOLIS Cedex (France)

EDITEUR
INRIA - Domaine de Voluceau - Rocquencourt - B.P. 105 - 78153 LE CHESNAY Cedex (France)

ISSN 0249 - 6399