Discrete Applied Mathematics 37/38 (1992) 9–11
North-Holland

Transmitting in the n-dimensional cube

Noga Alon

Department of Mathematics, Sackler Faculty of Exact Sciences, Tel Aviv University, Tel Aviv, Israel; and BellCore, 445 South Street, Morristown, NJ 07960, USA.

Received 1 July 1989

Abstract

Alon, N., Transmitting in the n-dimensional cube, Discrete Applied Mathematics 37/38 (1992) 9–11.

Motivated by a certain communication problem we show that for any integer n and for any sequence $(a_1,\ldots,a_k)$ of $k = \lceil n/2 \rceil$ binary vectors of length n, there is a binary vector z of length n whose Hamming distance from a_i is strictly bigger than $k-i$ for all $1 \le i \le k$.

The n-dimensional cube is the graph whose vertices are all 2^n binary vectors of length n, in which two vertices are adjacent if and only if their Hamming distance is 1, i.e., they differ in precisely one coordinate. Suppose there is a processor on each of these vertices and suppose there is an additional entity, outside the cube, which we call here the *Sender*, that has a message which has to be transmitted to all the processors on the cube. The transmission should be carried out as follows; in each time unit, which we call here a *round*, the Sender can send the message to any single processor on the cube, according to his choice. At the same time, each processor that knows already the message, sends it to his neighbours. Therefore, after the first round, one processor, say p_1, will know the message; after the second round, p_1 and all his neighbours will know the message, as well as some other processor p_2. In general, if in the ith round the Sender sends the message to the processor p_i, then after t such rounds the set of processors that already know the message is precisely the set of all those located on vertices of the cube whose distance from p_i does not exceed $t-i$ for some i, $1 \le i \le t$. The objective is, of course, to minimize the number of rounds until everybody knows the message.

There is a very simple procedure that enables the Sender to achieve this goal in $\lceil n/2 \rceil + 1$ rounds. He simply sends the message to some processor p_1 in the first round, then sends it to its antipodal, i.e., the processor located on the unique vertex of Hamming distance n from the location of p_1, in the second round, and then waits. Such a scheme may sound rather tempting for lazy Senders; they have to work only during two rounds, and then simply wait and rest. A natural question that arises is whether a Sender which is willing to work harder can do better, namely, is there any way in which the Sender may communicate the message to all 2^n processors in less than $\lceil n/2 \rceil + 1$ rounds? This problem, which was communicated to me by Ernie Brickell, originated from Intel because it describes the way they actually built some of their machines. The problem was first described explicitly by Joe Brandenburg and David S. Scott, and was popularized by Vance Faber at Los Alamos National Labs.

Here we show that the strategy of the lazy Sender, described above, is, in fact, optimal. That is: there is no scheme that enables a Sender (obeying the transmission rules mentioned) to transmit the message to all processors in less than $\lceil n/2 \rceil + 1$ rounds. As it turns out, this is a rather simple consequence of a lemma about matrices with $+1$, -1 entries, proved by Beck and Spencer in [2], which is the following.

Lemma [2]. *Let $A = (a_{i,j})$, $1 \le i \le n$, $1 \le j \le n$ be a matrix with n rows and n columns, where each $a_{i,j}$ is either a* 1 *or a* -1. *Then there is a vector y of length n with* 1, -1 *entries, such that for each i, the scalar product of y with the ith row of A is, in absolute value, strictly less than* $2i$.

The short (and clever) proof of this lemma, which appears in [2] uses a linear algebra method first applied in [1] (see also [3]). For the sake of completeness we repeat it here. For each i, consider $r_i = \sum_{1 \le j \le n} a_{i,j} x_j$ as a linear combination of the n real variables $x_1, \ldots, x_n$. These variables will assume, during the argument, values in the closed interval $[-1, 1]$, and in the end they will all be 1 or -1. At the beginning we set the value of each x_j to be 0. Let us call a variable x_j *floating* if its value belongs to the open interval $(-1, 1)$, otherwise call it *fixed*. We now describe a process for changing floating variables to fixed ones. Suppose that during the process there are still l floating variables. Consider the set of equations $r_i = 0$ for $1 \le i \le l-1$, as a linear system of equations in the l floating variables left, where the fixed variables are treated as constants. Since there are more variables than equations there is a line of solutions and we can choose a point of this line that intersects the boundary of the unit cube, i.e., a point in which at least one of the floating variables will become fixed. Initially, there are n floating variables. After n iterations of the above process, they all become fixed. We claim that the vector $y = (x_1, \ldots, x_n)$ satisfies the assertion of the lemma. Indeed, consider the scalar product of the vector $(x_1, \ldots, x_n)$ with the ith row of A. By the description of the process it is obvious that this scalar product was 0 as long as there were at least i floating variables. But afterwards, each of the remaining i iterations changed the absolute value of this scalar product by less than 2. This proves our claim and completes the proof of the lemma.

Now let us return to our cube problem. Denote the vertices of the n-cube by 1, -1 vectors of length n, where two such vectors are adjacent if and only if their

Hamming distance is 1. Suppose that for $1 \leq i \leq \lceil n/2 \rceil$ the Sender sends the message in the ith round to the processor p_i, and let $a_i = (a_{i,1}, \ldots, a_{i,n})$ be the 1, -1 vector describing the location of p_i. We now apply the Beck–Spencer Lemma stated above, to the matrix whose first $\lceil n/2 \rceil - 1$ rows are $a_2, a_3, \ldots, a_{\lceil n/2 \rceil}$ (the other rows are arbitrary). Note that a_1 is not one of these first rows. By the lemma, there is a 1, -1 vector y, whose scalar product with a_i is, in absolute value, strictly smaller than $2i-2$ for all $2 \leq i \leq \lceil n/2 \rceil$. Let $d(y, a_i)$ denote the Hamming distance between y and a_i. Clearly, the scalar product of y and a_i is simply $n - 2d(y, a_i)$. Therefore, for all $2 \leq i \leq \lceil n/2 \rceil$, $-2i+2 < n - 2d(y, a_i) < 2i-2$, i.e., $d(y, a_i) > n/2 - i + 1$, and also $d(-y, a_i) = d(y, -a_i) = n - d(y, a_i) > n/2 - i + 1$, where here $-y$ is the vector antipodal to y in the cube. As the distances are integers this gives that for all $2 \leq i \leq \lceil n/2 \rceil$, both $d(y, a_i)$ and $d(-y, a_i)$ are at least $\lceil n/2 \rceil - i + 1$. (For even n we get an even stronger statement.) Now clearly, either $d(y, a_1)$ or $d(-y, a_1)$ is at least $\lceil n/2 \rceil$. Thus, there is a vector z (which is either y or $-y$) such that $d(z, a_i) \geq \lceil n/2 \rceil - i + 1 > \lceil n/2 \rceil - i$ for all $1 \leq i \leq \lceil n/2 \rceil$. But this means that after the completion of these $\lceil n/2 \rceil$ rounds, the processor located on the vertex z has not yet received the message, completing the proof.

It is worth noting that the proof can be easily modified to show that for even n it is impossible to complete the task in $n/2$ rounds even if the Sender is allowed to send the message in each round to two processors located on antipodal vertices of the cube.

References

[1] J. Beck and T. Fiala, Integer making theorems, Discrete Appl. Math. 3 (1981) 1–8.

[2] J. Beck and J. Spencer, Balancing matrices with line shifts, Combinatorica 3 (1983) 299–304.

[3] V. Sevastyanov, On approximate solutions of scheduling problems, Metody Discret. Analiz. 32 (1978) 66–75 (in Russian).