DARTBOARD ARRANGEMENTS

G. L. Cohen*
School of Mathematical Sciences
University of Technology, Sydney
PO Box 123, Broadway
NSW 2007, Australia
glc@maths.uts.edu.au

E. Tonkes
School of Information Technology
Bond University
Gold Coast
Queensland 4229, Australia
etonkes@bond.edu.au

Submitted: February 17, 2000; Accepted: March 3, 2000.
AMS Subject Classification: Primary 05A05.

Abstract

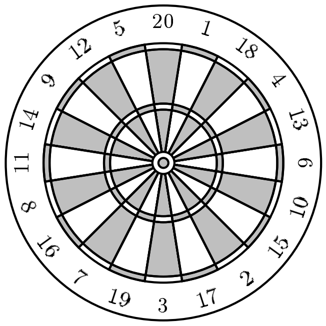
This note considers possible arrangements of the sectors of a generalised dartboard. The sum of the pth powers of the absolute differences of the numbers on adjacent sectors is introduced as a penalty cost function and a string reversal algorithm is used to determine all arrangements that maximise the penalty, for any $p \geq 1$. The maximum value of the penalty function for $p = 1$ is well known in the literature, and has been previously stated without proof for $p = 2$. We determine it also for $p = 3$ and $p = 4$.

Introduction

This note considers a combinatorial optimisation problem arising from the design of the standard dartboard, shown below. The arrangement may be construed as an effort to

*This paper was researched and prepared while a visitor at Bond University. The author is grateful for the hospitality shown by all.

penalise a player's poor shots. By aiming for a large number (other than the inner bull worth 50 or the outer bull worth 25) but hitting a small number to one side of it, the penalty will be incurred. Is it possible for other arrangements to allow greater overall penalties?



Several authors have considered this problem, sometimes generalised to a "dartboard" with n sectors numbered 1, 2, ..., n, for $n \geq 3$, say. A philosophy of maximising some penalty function seems sensible, and two such functions have received most attention: the sum of the absolute differences of all pairs of adjacent numbers, and the sum of the squares of these differences. We shall refer to these criteria as defining the L_1 and L_2 problems, respectively.

Let $\mathcal{A}$ be the set of all permutations of $\{1, 2, \ldots, n\}$ and write

$$D_p(A) = \sum_{j=1}^{n} |i_j - i_{j+1}|^p,$$

where $A = (i_1, i_2, \ldots, i_n) \in \mathcal{A}$ and $i_{n+1} = i_1$. The problems may be given more generally as

$$L_p: \qquad \text{Find } \widehat{A} \in \mathcal{A} \text{ such that } D_p(\widehat{A}) = \max_{A \in \mathcal{A}} D_p(A).$$

For $p = 1$ and $p = 2$, the problem seems to have been discussed first by Selkirk [5]. He makes a number of assertions, all of which are correct but none of which are proved. The concentration of effort by the later writers suggests that the proofs are not as easy to obtain as Selkirk seems to imply. Eiselt and Laporte [2] formulate L_1 and L_2 in the context of a maximum-cost travelling salesman problem, and solve them using a branch-and-bound algorithm. They only consider the case $n = 20$. Everson and Bassom [3] give a direct solution of L_1, but do not consider L_2.

To our knowledge, the only other published discussion of the design problem is Lynch [4], who also related it to the travelling salesman problem. With $n = 4m$, for integer m, he took as his criterion that adjacent numbers be of opposite parity and have absolute differences greater than $2m - 2$.

Our intention here is to use a string reversal algorithm to solve L_p, for any $p \geq 1$.

We refer to the permutation A above as a dartboard arrangement, and write it more usually as the string $A = i_1 i_2 \ldots i_n$. We call an arrangement $B \in \mathcal{A}$ equivalent to A if $B = i_q i_{q+1} \dots i_n i_1 \dots i_{q-1}$ for some q, $1 \le q \le n$ (cyclic permutation of A), or $B = i_r i_{r-1} \dots i_1 i_n \dots i_{r+1}$ for some r, $1 \le r \le n$ (reversed cyclic permutation of A). For an actual dartboard, these imply that it does not matter which number is uppermost or whether the board is reflected in a diameter. In this paper, all arrangements are permutations in $\mathcal{A}$, except for a few references to the original dartboard arrangement, which we refer to as A_d.

We are particularly interested in alternating dartboard arrangements, for which we must distinguish between even and odd values of n. For $n = 2k$, let $(s_1, s_2, \dots, s_k)$ be any permutation of $\{1, 2, \dots, k\}$, which we refer to as small numbers, and $(l_1, l_2, \dots, l_k)$ be any permutation of $\{k+1, k+2, \dots, 2k\}$, which we refer to as large numbers. The string $s_1 l_1 s_2 l_2 s_3 \dots l_{k-1} s_k l_k$ of alternately small and large numbers is an alternating arrangement. For $n = 2k+1$, let $(s_1, s_2, \dots, s_k)$ be any permutation of $\{1, 2, \dots, k\}$ and $(l_1, l_2, \dots, l_k)$ be any permutation of $\{k+2, k+3, \dots, 2k+1\}$, and put $m = k+1$. The string $m s_1 l_1 s_2 l_2 s_3 \dots l_{k-1} s_k l_k$ is an alternating arrangement.

Notice that A_d is not alternating.

The following result will be required.

Lemma 1 *For any real numbers $a < b < c < d$, and any $p \ge 1$,*

$$(d-a)^p + (c-b)^p \ge (c-a)^p + (d-b)^p,$$

with equality if and only if $p = 1$. Also

$$\min\{(d-a)^p + (c-b)^p, (c-a)^p + (d-b)^p\} > (d-c)^p + (b-a)^p.$$

Proof: Put $\alpha = d - c$, $\beta = c - a$ and $\gamma = c - b$, so that $\alpha > 0$, $\beta > \gamma > 0$ and we are required to prove first that

$$(\alpha + \beta)^p - \beta^p \ge (\alpha + \gamma)^p - \gamma^p.$$

The function f, where $f(x) = (\alpha + x)^p - x^p$ for $x > 0$, is easily seen to be increasing, and strictly so unless $p = 1$. Then, for $p > 1$, $f(\beta) > f(\gamma)$ as we wished to show.

Put $\delta = b - a$, so $\delta > 0$. We must prove also that

$$\min\{(\alpha + \gamma + \delta)^p + \gamma^p, (\alpha + \gamma)^p + (\gamma + \delta)^p\} > \alpha^p + \delta^p.$$

But this formulation makes the result obvious.

Solution of L_p

We begin with

Lemma 2 *For any $p \ge 1$, a necessary property of an arrangement $\widehat{A}$ that is a solution of L_p is that $\widehat{A}$ is alternating.*

Proof: We will describe an algorithm that allows us to transform an arbitrary non-alternating arrangement $A' \in \mathcal{A}$ into an alternating arrangement A in such a manner that the penalty function is increased at each stage of the process. Then $D_p(A) > D_p(A')$, so a non-alternating arrangement could not be maximal. That will prove the theorem.

We assume first that n is even: $n = 2k$. Notice in general that two large numbers appear successively in A' if and only if elsewhere two small numbers appear successively. The algorithm is as follows.

Step 1. From an arbitrary fixed number in the string move to the right, say. Identify the first occurrences of successive small numbers and successive large numbers. Reverse the substring from the second of the first pair to the first of the second pair, to give a new arrangement.

Step 2. If the result is an alternating arrangement, the process is complete. Otherwise, return to Step 1.

Following each application of Step 1, there are two fewer occurrences of successive small or large numbers, so the process must terminate in an alternating arrangement.

We will illustrate the algorithm with the string

$$A_1 = s_1 l_1 s_2 l_2 l_3 l_4 s_3 l_5 l_6 s_4 l_7 s_5 s_6 s_7 s_8 l_8 l_9 s_9.$$

As this is written, moving to the right, the first occurrence of a large pair is $l_2 l_3$ and the first occurrence of a small pair is $s_5 s_6$. Applying the algorithm, we obtain the following sequence of strings, where the substrings to be reversed have been indicated within square brackets:

$$\begin{aligned}
A_1 &= s_1 l_1 s_2 l_2 [l_3 l_4 s_3 l_5 l_6 s_4 l_7 s_5] s_6 s_7 s_8 l_8 l_9 s_9,\\
A_2 &= s_1 l_1 s_2 l_2 s_5 l_7 s_4 l_6 [l_5 s_3 l_4 l_3 s_6] s_7 s_8 l_8 l_9 s_9,\\
A_3 &= s_1 l_1 s_2 l_2 s_5 l_7 s_4 l_6 s_6 l_3 [l_4 s_3 l_5 s_7] s_8 l_8 l_9 s_9,\\
A_4 &= s_1 l_1 s_2 l_2 s_5 l_7 s_4 l_6 s_6 l_3 s_7 l_5 s_3 l_4 s_8 l_8 [l_9 s_9],\\
A_5 &= s_1 l_1 s_2 l_2 s_5 l_7 s_4 l_6 s_6 l_3 s_7 l_5 s_3 l_4 s_8 l_8 s_9 l_9,
\end{aligned}$$

and A_5 is alternating. In A_4, note the implied pair $s_9 s_1$.

At each repetition of Step 1, the value of D_p is increased. To see this, consider the typical situation

$$\begin{aligned}
B &= \ldots l_1 [l_2 \ldots s_1] s_2 \ldots,\\
C &= \ldots l_1 \; s_1 \ldots l_2 \; s_2 \ldots,
\end{aligned}$$

in which C results from B by application of Step 1. The calculations for the penalty function take the form

$$\begin{aligned}
D_p(B) &= \Delta + |l_1 - l_2|^p + |s_1 - s_2|^p,\\
D_p(C) &= \Delta + (l_1 - s_1)^p + (l_2 - s_2)^p,
\end{aligned}$$

respectively, where Δ is precisely the same calculation in both sums. In each of the four possibilities, namely $s_1 < s_2 < l_1 < l_2$, $s_2 < s_1 < l_2 < l_1$, $s_2 < s_1 < l_1 < l_2$ and $s_1 < s_2 < l_2 < l_1$, the second part of Lemma 1 implies that

$$|l_1 - l_2|^p + |s_1 - s_2|^p < (l_1 - s_1)^p + (l_2 - s_2)^p.$$

Hence $D_p(B) < D_p(C)$, so we conclude that D_p is maximal for an alternating arrangement.

When n is odd, the median number is treated as a small number if it lies directly between two small numbers and as a large number if it lies directly between two large numbers. This of course includes situations such as the string $ms_1 \dots s_2$ (in which m is treated as a small number). The algorithm may then be applied to arrive at an alternating arrangement, and the penalty function may be shown to increase, as before.

That completes the proof of Lemma 2.

We can now give a particular alternating arrangement that solves L_p. It is presented in a manner suggested by Brown [1].

Theorem 1 *A solution of L_p is given for each value of n (mod 4) as follows. Reading each as a single string from left to right, we have*

$$\begin{array}{cccccccccccccccccccc} 1 & & 3 & & 5 & & \dots & & 2k-1 & & 2k & & 2k-2 & & \dots & & 4 & & 2 & \\ & 4k-1 & & 4k-3 & & \dots & & 2k+3 & & 2k+1 & & 2k+2 & & 2k+4 & & \dots & & 4k-2 & & 4k \end{array}$$

when $n = 4k$;

$$\begin{array}{cccccccccccccccccccc} 1 & & 3 & & 5 & & \dots & & 2k+1 & & 2k & & \dots & & 4 & & 2 & \\ & 4k+1 & & 4k-1 & & \dots & & 2k+3 & & 2k+2 & & 2k+4 & & \dots & & 4k-2 & & 4k+2 \end{array}$$

when $n = 4k + 2$; and

$$\begin{array}{ccccccccccc} 1 & & 3 & & 5 & & \dots & & 2k-1 & & 2k+1 \\ & 2k & & 2k-2 & & \dots & & 4 & & 2 & \end{array}$$

when $n = 2k + 1$.

Furthermore, this solution is unique, up to equivalence, when $p > 1$.

Proof: The above strings may be given equivalently, for all n, as the first row of the following array:

$$\begin{array}{cccccccccccc} \dots & 5 & n-3 & 3 & n-1 & 1 & n & 2 & n-2 & 4 & n-4 & \dots \\ \dots & 9 & 7 & 5 & 3 & 1 & 2 & 4 & 6 & 8 & 10 & \dots \end{array}$$

Let this string be A, and let A'' be an arbitrary string in $\mathcal{A}$. By the algorithm in the proof of Lemma 2, we may transform A'' into an alternating string A', and increase the penalty function. The second row of the preceding array indicates the order in which we shall move the numbers of A' to transform it into A, again using string reversal. That is, we fix 1 in A', then move n into place to the right of 1, then $n-1$ to the left of 1, then move 2, then 3, then $n-2$, ... Lemma 1 will imply quickly that the penalty function D_p is increased with each move, and that implies that A is maximal.

To be specific, consider the intermediate string

$$\dots \ l \ \ k \ \dots \ s \ \ n-k+2 \ \ k-2 \ \dots \ 5 \ \ n-3 \ \ 3 \ \ n-1 \ \ 1 \ \ n \ \ 2 \ \ n-2 \ \ 4 \ \dots,$$

where it is required now to move the small number k into place. As before, l denotes a large number and s a small number. Let

$$\begin{aligned} B &= \ldots\ l\ [k\ \ldots\ s]\ n-k+2\ \ldots \\ C &= \ldots\ l\ \ s\ \ldots\ k\ \ n-k+2\ \ldots, \end{aligned}$$

where the substring $k\ \ldots\ s$ in B has been reversed in C. Observe that $k < s < l < n-k+2$, by virtue of the order in which the transformation is taking place. By the first part of Lemma 1,

$$(n-2k+2)^p + (l-s)^p \geq (l-k)^p + (n-k+2-s)^p,$$

so that $D_p(B) \leq D_p(C)$, with strict inequality if $p > 1$. The situation is precisely similar if k is now a large number to be moved into place, as from

$$\ldots\ s\ [k\ \ldots\ l]\ n-k\ k+2\ \ldots \quad \text{to} \quad \ldots\ s\ \ l\ \ldots\ k\ \ n-k\ k+2\ \ldots,$$

where $n-k < s < l < k$. As written, these two cases suggest that k is odd, but the situation is again precisely similar if k is even.

The conclusion follows, as indicated above, and, except for equivalent arrangements, the solution will be unique if $p > 1$.

Further remarks

(1) For L_1, we can prove that an alternating arrangement is also sufficient for a solution. Our method is different from that of Everson and Bassom [3].

Theorem 2 *A dartboard arrangement in $\mathcal{A}$ is a solution of L_1 if and only if it is alternating.*

Proof: By Lemma 2, for the proof we need only show that the penalty function $D_1(A)$ is the same for any alternating arrangement $A \in \mathcal{A}$.

Suppose n is even, $n = 2k$. For $A = s_1 l_1 s_2 l_2 s_3 \ldots l_{k-1} s_k l_k$, we have

$$\begin{aligned} D_1(A) &= (l_1 - s_1) + (l_1 - s_2) + (l_2 - s_2) + \cdots + (l_k - s_k) + (l_k - s_1) \\ &= 2(l_1 + l_2 + \cdots + l_k) - 2(s_1 + s_2 + \cdots + s_k) \\ &= 2\Big(((k+1)-1) + ((k+2)-2) + ((k+3)-3) + \cdots + (2k-k)\Big) \\ &= 2k^2. \end{aligned}$$

For n odd, $n = 2k+1$, and $A = m s_1 l_1 s_2 l_2 s_3 \ldots l_{k-1} s_k l_k$, it is easy to see that, again, $D_1(A) = 2k^2$.

In summary, for every alternating arrangement $A \in \mathcal{A}$, $D_1(A) = 2[n/2]^2$, where $[\,\cdot\,]$ is the greatest integer function.

Notice that $2[n/2]^2 = 200$ when $n = 20$, whereas $D_1(A_d) = 198$.

(2) Selkirk [5] gives the result

$$D_2(\widehat{A}) = \begin{cases} \frac{1}{3}n^3 - \frac{4}{3}n + 2, & n \text{ even}, \\ \frac{1}{3}n^3 - \frac{4}{3}n + 1, & n \text{ odd}, \end{cases}$$

which may be confirmed from the maximal arrangements given in Theorem 1.

When $n = 20$, we have $D_2(\widehat{A}) = 2642$, whereas $D_2(A_d) = 2374$.

We have also obtained the following results:

$$D_3(\widehat{A}) = \begin{cases} \frac{1}{4}n^4 - 2n^2 + 3n, & n \text{ even}, \\ [2]\frac{1}{4}n^4 - 2n^2 + 3n - \frac{5}{4}, & n \text{ odd}, \end{cases}$$

$$D_4(\widehat{A}) = \begin{cases} \frac{1}{5}n^5 - \frac{8}{3}n^3 + 6n^2 - \frac{68}{15}n + 2, & n \text{ even}, \\ [2]\frac{1}{5}n^5 - \frac{8}{3}n^3 + 6n^2 - \frac{68}{15}n + 1, & n \text{ odd}. \end{cases}$$

(3) A different criterion has been proposed by Brown [1]. He suggests minimising $\sum_{j=1}^{n}(i_j + i_{j+1})^2$, in the notation of our introduction. Since

$$\sum_{j=1}^{n}(i_j + i_{j+1})^2 = \sum_{j=1}^{n} i_j^2 + \sum_{j=1}^{n} i_{j+1}^2 + 2\sum_{j=1}^{n} i_j i_{j+1} = 2\sum_{j=1}^{n} j^2 + 2\sum_{j=1}^{n} i_j i_{j+1}$$

and

$$\sum_{j=1}^{n}(i_j - i_{j+1})^2 = 2\sum_{j=1}^{n} j^2 - 2\sum_{j=1}^{n} i_j i_{j+1},$$

we have

$$\sum_{j=1}^{n}(i_j + i_{j+1})^2 = \frac{2}{3}n(n+1)(2n+1) - \sum_{j=1}^{n}(i_j - i_{j+1})^2,$$

so Brown's approach is seen to be equivalent to the problem L_2.

(4) It is apparent that our methods rely on the ordering of the natural numbers. They may be applied equally to any set of real numbers: rank the numbers in increasing order, with equal numbers given successive increasing ranks arbitrarily. Then Theorem 1, applied to the rankings and then interpreted in terms of the original numbers, will give an arrangement that solves the analogue of L_p, and, for $p > 1$, the solution will be unique up to equivalence.

(5) In a private communication, Dr Neil Williams has pointed out to us that string reversal has been used on a related problem in the 1983 Australian Mathematical Olympiad. We are grateful for that comment, and also for referring us to the article by Lynch [4].

References

[1] K. S. Brown, "The dartboard sequence", downloaded from the website http://www.seanet.com/~ksbrown/kmath025.htm (posted February 1996).

[2] H. A. Eiselt and G. Laporte, "A combinatorial optimization problem arising in dartboard design", *J. Op. Res. Soc.*, **42** (1991), 113–118.

[3] P. J. Everson and A. P. Bassom, "Optimal arrangements for a dartboard", *Math. Spectrum*, **27** (1994/5), 32–34.

[4] M. A. M. Lynch, "Designing a dartboard — an application of graph modelling", *Teaching Math. and Its Appl.*, **16** (1997), 51–54.

[5] K. Selkirk, "Re-designing the dartboard", *Math. Gaz.*, **60** (1976), 171–178.