

Optimal Dartboard Design with Simulated Annealing

Eyüp Ensar IŞIK¹

¹*Yildiz Technical University, İstanbul, Turkey, eeisik@yildiz.edu.tr, ORCID: 0000-0002-9180-0243*

Darts game is a popular game where people compete in competition while having fun. Although most people do not know, the dartboard design used while playing the game is an optimization problem studied by different researchers. A dartboard design that will increase the competition in the game and make the game more complex can be set up as a combinatorial optimization problem. This study used the Simulated Annealing algorithm to obtain an optimal dartboard design. Contrary to the algorithm's classical structure, it aims to achieve better results by using more than one neighborhood structure. By running the designed algorithm with four different objective functions, the best designs were tried to be obtained. As a result, in addition to the existing designs in the literature, two new designs are proposed that provide good solutions for all objective functions.

Keywords: *Dartboard design, Meta-heuristic, Simulated Annealing, Penalty function method.*

© 2022 Published by **AIntelia**

1. Introduction

The game of darts is even more common in the British Isles but is also one of the most popular in the world. Its origins date back to the sixteenth century, and it was played in 1620 by the pilgrims on the Mayflower. The current scoring system was devised in 1896 by Brian Gamlin, of Bury in Lancashire [1]. There are several game variations, but the most common appears to be that players subtract scores to reduce an initial value of 301 or 501 to zero [2].

In a classic darts game, the rules are simple. Each player takes turns throwing three darts and scores points based on the area they hit on the dartboard divided into 20 equal parts. These scores are written on the dartboard, but it is possible to get double or triple points if particular areas on the dartboard are hit. In addition, there is a zone on the dartboard, defined as "bull," and located in the middle of the board; this zone gives the player 25 points. Inside this zone is a smaller circle known as the "bull's eye," which gives 50 points.

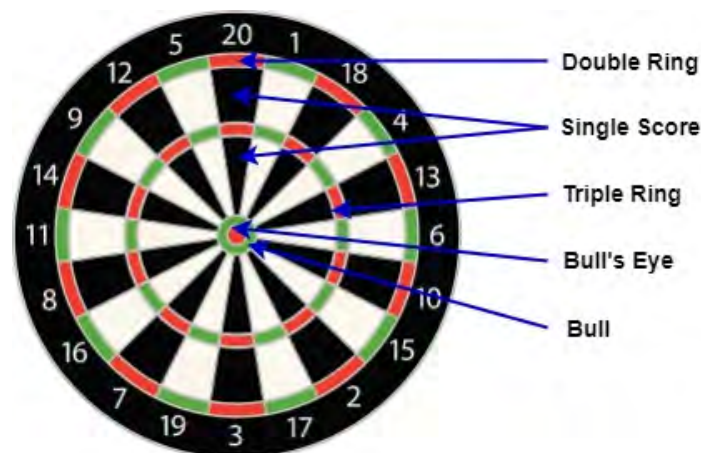


Figure 1. Scoring system of the classic dartboard.

Points scored are subtracted from the starting point 501 for each player. In the beginning, each player has 501 points and the points received are deducted from this number. The last dart thrown must come into the double ring. The object of the game is to reach exactly zero first. If the player stays at exactly "1" point or drops below zero, darts fired in the last round are disregarded.

At the beginning of the game, the goal is to get as many points as possible and get closer to zero. However, the strategy changes since an even number is needed at the end of the game. At this stage, a double of an even number is targeted, ensuring that if a double is missed and a single is hit, the next shot will give an "an out" in darts jargon. For example, a single 18 allows a double 9 with the next shot to win the game [3]. One natural question about the game of dart is whether the numbers on the dartboard are well located with respect to some optimality criterion and, indeed, which criterion should be used to compare various dartboard designs [2].

In a standard dartboard, low numbers are placed next to high ones to penalize players who miss their targets. That's why the 20, for example, is next to the 1 and the 5. Mathematicians have long come up with improved arrangements that maximize the differences between adjacent numbers to penalize mistakes as much as possible.

Mathematicians say the new dartboard will make the most difference at the end of a game when the rules are that a player must finish on the double. Currently, if a player is on an odd number and therefore needs an odd number to leave himself with an even, he can choose from the southwest sector of the board where four odds are adjacent: 7, 19, 3, and 17.

Even a bad player can expect to get an odd number. But if the odds and evens alternate, it becomes much more difficult. Also, the most preferred ending double to aim for is double 16 since if the shot is missed the double and hit single 16, it is required double 8 (and if missed the double and hit 8, requires double 4, then double 2, and then double 1.) On a traditional board, 8 is right next to 16, which makes the game easier since players are already aiming for that section of the board [4].

Researchers conduct some studies in recent years to achieve a fairer, more difficult, and optimal dartboard design. Over the standard procedure, they address new constraints and tried to find the optimal dartboard design to overcome the aforementioned disadvantages.

2. Literature Review

In the literature, Selkirk [5] address the dartboard design problem for the first time. In his study, the author first drew attention to the problem's existence and then reveal the structure of the problem. The author says the standard design is not balanced regarding the differences between adjacent numbers. He brings a statistical solution to this situation: mean and standard deviation. The sum of squares of differences of adjacent numbers ordered from 1 to 20 should be maximized. There is only one optimal solution for this situation. The author also finds other possible designs according to "Quadrant sums vary as little as" and "A vector solution."

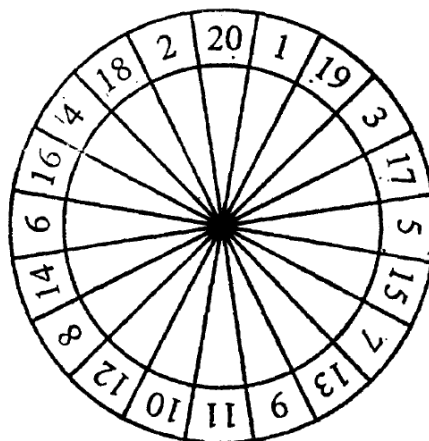


Figure 2. Selkirk's optimal dartboard design [5].

Kohler [6] provides an analysis of the game under some hypotheses:

- Players aim at specific target points on the board rather than at general areas;
- The distribution of the point reached by the dart is a symmetrical bivariate normal of the standard deviation a , where a is inversely related to the accuracy of the player;
- This distribution does not depend on the particular target point.

And the author used dynamic programming to design the dartboard for the darts game version, where the score is subtracted from 301. HA. Eiselt and G. Laporte [2] considered the dartboard design problem as Traveling Salesman Problem (TSP) and Quadratic Assignment Problem and suggested different designs. Similarly, Curtis [7] addressed the dartboard design problem as TSP with maximum costs and proposed a greedy algorithm for both circular dartboards and linear hoopla boards. Liao [8] and Liao et al. [9] generalize the dartboard problem by considering more circles, and the goal is to arrange k_n number on k circles to obtain the maximum risk.

Donkers [3] examined six different criteria for optimal dartboard design. According to the author, although these criteria are based on entirely different ideas and use different mathematical methods, they do not provide a consistent solution. By most criteria, the classic design scores the worst, and Selkirk's design is the best. Therefore, the author used the simulated annealing algorithm with the center of gravity method to find the optimal dartboard design in his work. Even though the author finds good results, it would be interesting to see what results from simulated annealing would find for the other criteria.

Percy [10], [11] has several studies for optimal dartboard design. In the latter, the author emphasizes that the problem has been studied before, but he considers new constraints and different optimality criteria. The author find a new optimal dartboard according to new constraints and optimality criteria, but in conclusion, he says Selkirk's design is still the best.

Eleni & Aristidis [12] and Drosos et al. [13] use ant-based metaheuristics to solve the problem. In [13], Drosos et al. underlined that the dartboard design problem is a combinatorial optimization problem. In their paper, they solve this problem using Ant System (AS) and Max-Min Ant System (MMAS) algorithms. Both algorithms have been proven to be very effective in finding the optimum solution to hard combinatorial optimization problems. AS and MMAS algorithms are used to solve the problem. MMAS algorithms prove to be much faster and provide better solutions.

In this study, a simulated annealing meta-heuristic algorithm is used to find the optimal dartboard design. The constraints Percy [11] used in his last work is be considered for this. Then, the results are compared to the different optimality criteria previously used in the literature.

3. Problem Definition and Formulation

There are $20! \approx 2 \times 10^{18}$ possible arrangements of the numbers 1, 2, . . . , 20 on a dartboard, and $19! / 2 \approx 6 \times 10^{16}$ distinct cycles that allow for reflection and rotation. The objective is to determine a cycle that is optimal in some sense. According to [11], there are three constraints to be imposed to any cycle:

1. Penalize mistakes by overambitious players. This is apparent in the standard dartboard designed by Gamlin (1896), in which large numbers tend to be adjacent to small numbers.
2. Alternate odd and even numbers. This parity criterion is proposed by Eastaway and Haigh [14]. It is particularly appealing because it induces a degree of symmetry and ensures a challenging endgame.
3. Exhibit rotational quasi-symmetry. We propose this criterion to ensure that similar clusters of adjacent sectors around the dartboard offer similar rewards to players.

The notation is as follows. Define the twenty numbers reading clockwise from the top of a dartboard to be x_i for $i = 1, 2, \dots, 20$. The ordered set $(x_1, x_2, \dots, x_{20})$ forms a cycle and $x_0 = x_{20}$ for convenience. The standard dartboard has the arrangement is; (20, 1, 18, 4, 13, 6, 10, 15, 2, 17, 3, 19, 7, 16, 8, 11, 14, 9, 12, 5) as illustrated in Figure 1.

The objective function is generally taken as maximizing the aggregate penalty measures corresponding to specific forms of the p -norm in terms of the differences between pairs of adjacent numbers. Therefore, as mentioned earlier, some studies have seen the problem as "TSP with maximizing cost."

$$\|d\|_p = \left(\sum_{i=1}^{20} |d_i|^p \right)^{1/p}, d_i = x_i - x_{i-1} \quad (1)$$

Singmaster [15] suggest that differences between adjacent numbers do not necessarily penalize mistakes by overambitious players. For example, the cycle achieves $\max \|d\|_1 = 200$ but the differences on either side of 12 are 11 and 10, whereas the differences on either side of 18 are only 9 and 8. Hence, players fare considerably better on average by aiming for 18 sectors rather than 12. The author proposes that sums of adjacent numbers should be about equal. Instead, this measure does indeed satisfy Constraint (1). To achieve this, he sinks to minimize the variance of these sums and prove algebraically the existence of a unique solution that achieves this optimality, which is equivalent to Figure 2 that maximizes the Euclidean norm $\|d\|_2$. The author also proves that this criterion corresponds to minimizing the cyclic autocorrelation of lag one, which is an excellent interpretation of the requirement implied by Constraint (1). Therefore, different p values get different solutions and advantages. In this study, the objective function taken as Eq. (1) and different solutions is obtained for p values 1 (Manhattan norm) and 2 (Euclidean norm).

Although Percy [11] supports Singmaster's recommendation to consider sums of adjacent numbers rather than differences [15], the author recalls that the solution based on minimizing the variance of these sums fails to satisfy the parity requirement of Constraint (2). Seeking other possible variations on this theme inspired by Constraint (1), which might lead to solutions that satisfy the parity requirement, the author considers optimality criteria that involve minimizing aggregate penalty measures corresponding to specific forms of the p -norm in terms of the central sums for $i = 1, 2, \dots, 20$:

$$\|c\|_p = \left(\sum_{i=1}^{20} |c_i|^p \right)^{1/p}, c_i = s_i - \bar{s}, \quad s_i = x_i + x_{i-1} \quad (2)$$

$$\bar{s} = \frac{1}{20} \sum_{i=1}^{20} s_i = 21 \quad (3)$$

So, with this recommendation, we have four different objective functions or optimality criteria: $\|d\|_1$, $\|d\|_2$, $\|c\|_1$ and $\|c\|_2$. The algorithm is executed more than once with the constraints or not, and the results are interpreted in Chapter 5.

4. Proposed Simulated Annealing Algorithm

Our objective functions and constraints are described in the previous section. Initially, our fitness function is taken as $\|d\|_1$. Since Constraint (1) is also directly proportional to the objective, there is no need for a separate definition within this constraint. Since handling Constraint (2) with penalty function methods may cause the algorithm to consistently produce an unfeasible solution, the solution representation is directly taken in the even-odd order.

A. Encoding the initial solution

The solution representation for dartboard design is a permutation with numbers 1 to 20. But the largest number, 20 in all designs in the literature and practice, is at the top of the dartboard. For this reason, we fix the number 20 to the first position in the solution demonstration. In addition, as mentioned earlier, Constraint (2) is provided by direct solution representation. Since it is fixed to the first position in the number 20, the sequence continues as even-odd-even-odd... An example is shown below:

(20 1 2 11 10 9 12 19 18 3 4 15 6 13 16 7 8 5 14 17)

B. The neighborhood structure

Five different neighborhood structures have been determined for neighborhood search. These are;

- Swap,
- 3-Swap,
- Inverse,
- Circular Shift,
- Insertion.

All structures are planned so as not to disturb the even-odd structure mentioned earlier. In this way, we can guarantee that Constraint (2) is provided invariably.

C. Fitness evaluation

The fitness function is calculated according to Constraints (1) and (2). Calculations are made between all adjacent numbers, and finally, the numbers between the first and last positions are calculated. So, for the example given in Section A, the results are as follows:

$$\|d\|_1 = 116, \|d\|_2 = 34.35, \|c\|_1 = 124 \text{ and } \|c\|_2 = 38.47.$$

Aggregate measures such as those used for the optimality criteria in Constraints (1) and (2) do not guarantee rotational quasi-symmetry. To demonstrate this, consider Selkirk's design, which optimizes the norms $\|d\|_1$, $\|d\|_2$, $\|c\|_2$ and possibly $\|c\|_1$. The sum of absolute differences in the semi-circle that surrounds the number 20 is 149, whereas the sum of absolute differences in the opposite semi-circle is 51. Together, these sum to $\max \|d\|_1 = 200$, but this disparity in the subtotals reveals a clear and undesirable lack of rotational quasi-symmetry. This particular cycle does not exhibit a similar disparity when considering central sums evenly distributed around the board. However, it is conceivable that similar disparities might occur more generally for optimal cycles based on aggregate measures corresponding to the p-norms $\|d\|_p$ and $\|c\|_p$.

When we consider Constraint (3), we need a penalty function method. Static penalty functions are simple and popular methods for constrained optimization. These functions do not consider the current generation number in the penalty parameters and impose a fixed penalty on unfeasible solutions. A static penalty approach is proposed in the literature, in which users define some level of violation [16]. This method follows the following steps:

- For each constraint, first, produce l levels of violation,
- For each violation level, calculate a penalty coefficient (larger coefficients correspond to higher violation levels) R_{ij} ($i = 1, \dots, l$; $j = 1, \dots, m$), do this also for each constraint,
- Generate a random set of solutions using both feasible and infeasible solutions,
- Evaluate each solution using the equation $eval(\bar{x}) = f(\bar{x}) + \sum_{j=1}^m R_{ij} \max [0, g_j, (\bar{x})]^2$

Constraints are not always provided at a hundred percent. For this, a tolerance value is determined in the penalty functions. In the article where the quasi-symmetry constraint is introduced for dart board design, the optimum result is the difference between the two semi-circles = 18. Therefore, the tolerance value was taken as 20 in this study. When the difference between the two semi-circles exceeds 20, the fitness function value will be summed with a very large number for the maximization problem and removed from a very large number for the minimization problem. As long as there is improvement in each iteration, fitness function value and solution permutation are updated; some bad solutions are accepted according to the cooling schedule.

5. Computational Study

As said before, the algorithm has been run for four objective functions, depending on whether it is constrained. With the specific parameters of the algorithm given below, the algorithm has been run 20 times for each case, and the results are summarized in the Table 1. The parameters are set to the appropriate values, and the parameters that the

change does not affect the result are taken as the set value. Only the maximum number of iterations was changed. The parameters are;

- Maximum number of iterations = 1000 and 10000
- Length of Markov Chain (Minimum number of accepted solutions) = 75
- Inner loop maximum number of iterations = 150
- Initial temperature = 10000
- Constant for decreasing temperature = 0.999

Table 1. SA Results for Dartboard Design.

SA for Dartboard Design		Max $\ d\ _1$				Max $\ d\ _2$				Min $\ c\ _1$				Min $\ c\ _{12}$			
		MIN	AVG	MAX	STD	MIN	AVG	MAX	STD	MIN	AVG	MAX	STD	MIN	AVG	MAX	STD
Iteration =1000	Unconstrained	182	188.5	196	3.993	46	47.015	47.958	0.600	56	69.2	80	6.879	16.248	20.909	22.627	1.591
	Constrained	178	184.4	194	4.185	44.9	46.574	47.875	0.938	60	72.8	84	6.169	19.799	22.039	24.331	1.243
Iteration =10000	Unconstrained	190	193.3	198	2.452	47.4	48.254	49.437	0.598	36	36	36	0	8.485	8.715	8.943	0.236
	Constrained	190	193.7	198	2.849	47	47.805	48.867	0.538	36	36	36	0	8.485	8.623	8.943	0.216

When the table is examined, it can be seen that the increase in the number of iterations improves the solutions. The best values in Percy's [11] study are obtained for all objective functions when taking the maximum number of iterations to 10,000. Adding constraints to the algorithm does not have a major impact on the results. However, when the constraint is added, it is guaranteed to provide the quasi-symmetry feature. Therefore, running the algorithm as constrained and maximum iteration = 10,000 give the most appropriate results.

Although there are different objective functions, the problem is not considered multi-objective. However, since a single design should be obtained at the end of the study, the best results for all objective functions should be compared, and the best one should be selected. When the algorithm worked for the $\|d\|_1$ and $\|d\|_2$ objective functions, they found the best results for themselves and each other but could not find the best-known results for the other objective functions.

When the algorithm works for the objective function $\|c\|_1$, it is able to find the best solutions for the $\|d\|_1$ and $\|d\|_2$ objective functions as well. But it couldn't find the best solution for $\|c\|_1$. Finally, when it works for the objective function $\|c\|_2$, the same solution found in Percy's study [11] is found with a success rate of over fifty percent. Percy's design can be seen in Figure 3. This rate increases even more when the problem is handled with constraint.

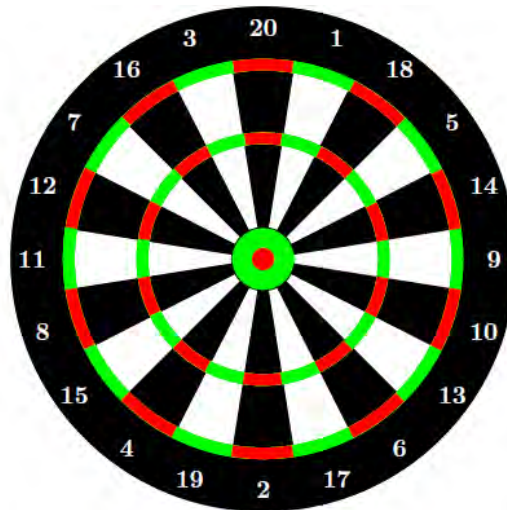


Figure 3. Percy's design.

By looking at these results, it can be said that the objective function $\|c\|_2$ is the most effective objective function. However, with the other objective functions, several alternative solutions have been found that are not dominated. A comparison with other designs in the literature is given below with a table, like in Percy's [11] study. Alternative

designs present in addition to the designs in the study are also given in Table 2. The newly found alternative designs are presented visually in Figure 4 and Figure 5.

Table 2. Dartboard Design Comparison.

Dartboard	$\ d\ _1$	$\ d\ _2$	$\ c\ _1$	$\ c\ _2$
Gamlin's	198	49.8	52	13.5
Selkirk's	200	51.4	18	4.2
Percy's	198	50.9	36	8.5
Alternative 1	198	50.6	36	9.8
Alternative 2	198	50.8	36	8.9

Gamlin's and Selkirk's designs do not match the even-odd order. Therefore, they are more flexible in obtaining the results. Selkirk's design is still best when its even-odd order is not considered.

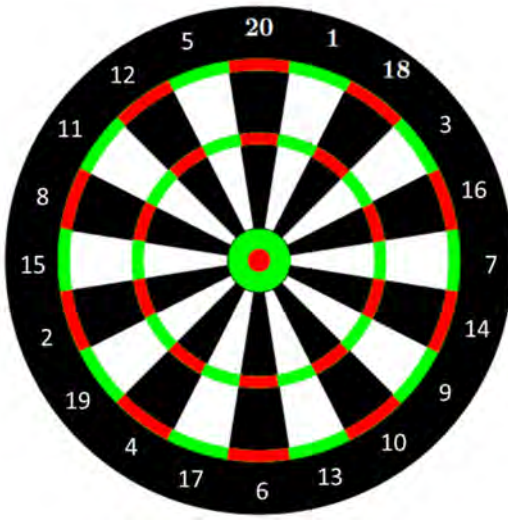


Figure 3. Alternative Design 1.

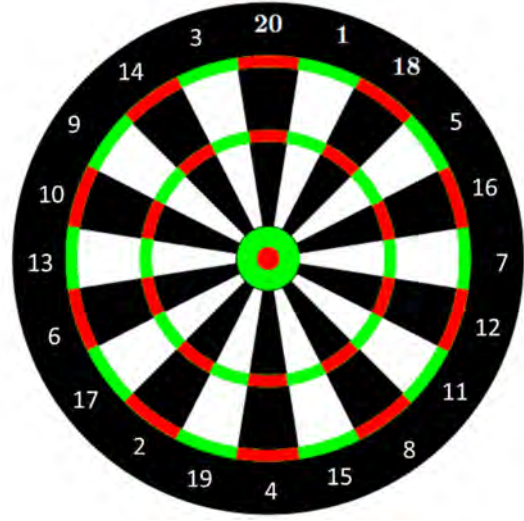


Figure 4. Alternative Design 2.

6. Conclusion

The optimal Dartboard Design problem is a combinatoric optimization problem that has been studied extensively in the literature for fun. Researchers from different disciplines offer different approaches to solving this problem. One of these methods is to use metaheuristics. Different designs are obtained according to many different objective functions by using several methods. In this study, the objective functions and constraints in Percy's [11] study are considered, and the problem is tried to be solved with a well-known meta-heuristic algorithm, simulated annealing. While the first of the constraints is parallel to the objective function, the second constraint is provided with a solution representation. Depending on whether the third constraint is handled or not, the algorithm is run separately.

When the results obtained are examined, it is seen that objective function $\|c\|_2$ is the most effective and Percy's optimal dartboard design is easily reached by using this objective function. However, according to this solution, two different solutions that are not dominated are obtained, and these solutions are presented as Alternative Design 1 and 2. Thus, in addition to the many designs in the literature, new designs are introduced, contributing to the literature, and the Darts game is made more challenging. Future studies can investigate the performance of different metaheuristic algorithms on the problem. In addition, optimal designs can be tried to be obtained by using metaheuristic algorithms for different problem variations (such as double-layer dartboard [9]).

REFERENCES

- [1] D. McFarlan, "The Guinness Book of Records", Guinness Publishing, Enfield, Middlesex, 1990, p. 242. Ed. 1989.
- [2] H.A. Eiselt, G. Laporte, "A combinatorial optimization problem arising in dartboard design", *1.0per. Res. Soc.*, vol. 42, pp. 113-118, 1991.
- [3] P. W. A. Donkers, "Dartbord arrangement" Delft University of Technology Faculty of Electrical Engineering, Mathematics and Computer Science Delft Institute of Applied Mathematics, 2010.
- [4] A. Bellos, "Optimal dartboard hits the bullseye", *The Guardian*. UK, 2020, June 5. [https:// www.theguardian.com/science/ alexs-adventures-in numberland/2013/jan/09/optimal-dartboard-hits-bullseye](https://www.theguardian.com/science/alexs-adventures-in-numberland/2013/jan/09/optimal-dartboard-hits-bullseye).
- [5] K. Selkirk, "Re-designing the dartboard", *Math. Gazette*, vol. 60, pp. 171-178, 1976.
- [6] D. Kohler, "Optimal strategies for the game of darts", *Journal of the Operational Research Society*, Vol 33, pp. 871-884, 1982.
- [7] S.A. Curtis, "Darts and hoopla board design", *Information Processing Letters*, 92 (2004) 53–56.
- [8] Y. J. Liao, "On the Design of Generalized Dartboard". October 19, 2006.
- [9] Y. J. Liao, M. Z. Shieh, S. C. Tsai, "Arranging numbers on circles to reach maximum total variations", *The Electronic Journal of Combinatorics*, 14 2007, #R47.
- [10] D. F. Percy, "Winning darts!", *Mathematics Today*, vol. 35, pp. 54–57, 1999.
- [11] D. F. Percy, "The Optimal Dartboard?" *Mathematics Today*, vol. 35, pp. 268–270, 2012.
- [12] A. Eleni, V. Aristidis, "A meta-heuristic algorithm is used for dartboard design", *International Journal of Management Science and Engineering Management*. Vol. 4 (2009) No.3, pp. 198-204
- [13] C. Drosos, G. Trichas, A. Vlachos, "A comparative study of meta heuristic algorithms for Dartboard Design" Conference: Fifth International Conference on Information, Intelligence, Systems and Applications IISA, Chania, Greece. 2014. DOI: 10.1109/IISA.2014.6878759.
- [14] R. Eastaway, and J. Haigh, "The Hidden Mathematics of Sport", London. Portico, 2011.
- [15] D. Singmaster, "Arranging a dartboard", *IMA Bulletin*, vol. 16, pp. 93-97, 1980.
- [16] A. Homaifar, S.H.Y. Lai, and X. Qi, "Constrained optimization via genetic algorithms", *Simulation*, 62, 242-254, 1994.