NEW MATHEMATICAL MODELS OF N-QUEENS PROBLEM AND ITS SOLUTION BY A DERIVATIVE-FREE METHOD OF OPTIMIZATION

Ali J.[1], M.Saeed[1], N. A. Chaudhry[2] M.F.Tabassum[1]*
[1]Department of Mathematics, University of Management and Technology, Lahore, Pakistan.
[2]Department of Mathematics, UCEST, Lahore Leads University, Lahore, Pakistan.
* Corresponding Author: Muhammad Farhan Tabassum, farhanuet12@gmail.com, +92-321-4280420

Abstract: *Most of the optimization methods do not inherit convergence proofs. One of the measures to rank them is their potential to solve challenging problems specially formulated for this purpose. In this paper the problem involves two main issues. Firstly we present new formulations of the N queens' configuration problem as optimization problems and secondly we modify a derivative free method so that it may be able to find the optimal configuration of the chessboard.*

Keywords: N-Queen Problem, Optimization, Non-linear Programming Problem, Hooke's and Jeeve's Method

1. INTRODUCTION

The eight queens is a well-known NP-complete problem proposed by C. F. Gauss in 1850 [11]. A number of techniques have been developed for solving N-queens chessboard problem. For example, a backtracking search technique [4, 5] is a search based algorithm which systematically generates all possible solutions for the given $N \times N$ board. Many methods, like search heuristic methods, local search and conflict minimization techniques, neural networks, Hopfield networks, integer programming of N queens problem as an assignment problem scheme have been reported in [2]. Another method is the method of dancing links which was made popular by Knuth [10]. A DNA sticker algorithm for solving N-Queen problem is given in [12] which can solve the problem in $O(N^2)$ time. Particle Swarm Optimization technique has been applied and the efficiency measures have also been tested in [8]. In spite of all these temptations the applications of derivative free deterministic methods to such type of configuration problems can rarely be seen.

2. MATERIAL AND METHODS

2.1 Hooke and Jeeves Method (HJ)

Robert Hooke and T.A. Jeeves coined the phrase "direct search" in a paper that appeared in 1961 in the Journal of the Association of Computing Machinery [14]. This method is the discrete form of the classical *coordinate descent algorithm* [19]. As it uses the coordinate search directions without considering the behavior of the function therefore it may require high number of function evaluations. But it is less likely to fail through numerical ill-conditioning [16]. The mathematical model of N-Queen considered by us appears to be discrete, non-differentiable and an integer programming problem. We are interested in investigating the capability of HJ method but not efficiency. Therefore we do not bother about the computational cost. The method requires an initial guess and a set of linearly independent search directions spanning the whole search space. For an n-dimensional optimization problem we will use a standard basis of n-dimensional Euclidean space following the implementations in [20]. The method applies two types of actions, termed as moves in [21], to improve the initial solution by considering it as current base point (x_b). First move is the exploratory move. An exploratory move can be categorized as Type I exploratory move and a Type II exploratory move as described in [5]. Type I exploratory move performs a local search in the vicinity of x_b [20] by stepping forward and backward along each search direction. The success, achievement of a better point x_E over x_b, of such an exploratory move leads to the next move called pattern move where as its failure results in reduction in the step lengths and hence leads to termination criteria. HJ *pattern move* is an aggressive one and tries to move further rapidly [15] by rotating x_b about x_E through an angle of 180^o. The new point found by pattern move is $x_b + 2(x_E - x_b)$ [21]. For further reference we will denote it by x_p. This new point x_p acts as a temporary base point in the sense that it becomes the current iterate according as its value is less than $f(x_E)$ or not [17]. The next phase of the HJ method is to perform Type II exploratory move [22] (Explore-II) on x_p. The outcome of Explore- II is labeled as x_e here. Such a move tries to improve x_p in its neighborhood. The pattern move is a success if $f(x_e)<f(x_E)$ otherwise it is a failure [18]. In case of success x_e is updated as current base point and the algorithm repeats the pattern move. Failure of the pattern move invokes ExploreI with x_E as the base point. The process is repeated until a preset termination status is reached. The maximum number of function evaluations, maximum number of iterations to be executed or sufficiently small step lengths can be used.

For ours implementations we consider that:

(i) $\underline{D}$ as a vector of step lengths whose i^{th} component d_i represents increment along i^{th} coordinates direction.

(ii) The reduction parameter for i^{th} increment is $\alpha_i = \frac{d_i}{d_i\text{-}1}$ i. e; each failed exploratory move results in a new $\underline{D}$ with each $d_i = \frac{1}{\alpha_i} d_i$.

(iii) The termination criteria $\max_{i=1}^{n} \{d_i\} < \varepsilon$ where ε is tolerance parameter.

(iv) Type II Exploratory move is called on x_P only if $f(x_P) \le f(x_E)$ i.e; a local search is applied on x_P only if it proves to be a better point. We expect this choice will reduce the computational cost.

(v) The penalty function:

$$\text{Penalty}(X) = R \times \max_{j=1}^{N} \{0, X_j - N^2, -X_j\},$$

where R is penalty factor.

The algorithm generates only one or several solutions but not necessarily all, thus falls in third category mentioned in [13].

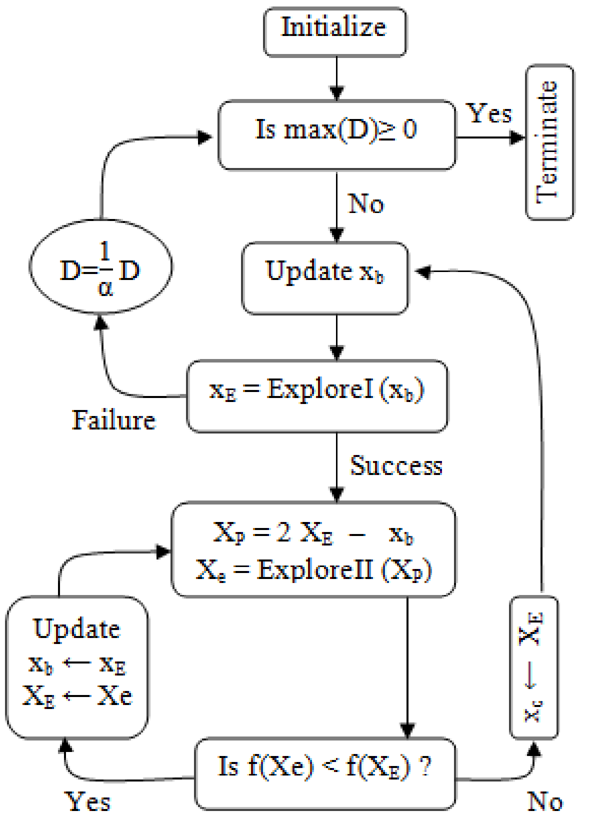

Figure 1: Flow chart of the modified HJ method

2.2 N-Queens Problem-Problem Statement

In an N queens chessboard problem there are N×N square boxes. The strategic moves of each queen are horizontal, vertical, left diagonal and the right diagonal across the chessboard. Queens are said to attack each other if they appear on the same strategic path. The objective is to place N queens on an N x N chess board, such that no pair of queens can attack each other according to the rules of standard chess [1]. If N=4 then the chessboard and the tracks followed by a queen are shown in the adjoining figure.

For an N×N chessboard problem there is a set of $\binom{N^2}{N}$ possible configurations [1] of N queens, which is actually a subset of the collection of $(N^2)^N$ total permutations. This collection constitutes the search space for the chessboard configuration problem when it is considered as an optimization problem until a special strategy is adopted to drop the identical or symmetric permutations. But empirical observations for smaller-size problems reflect that there is an exponential increase in the number of solutions when N increases [3].

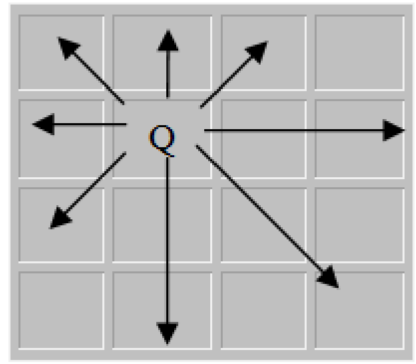

Figure 2: Queen in 4×4 chess-board

2.3 Preparation of N-Queens Optimization Model

N-Queen problem has been formulated in various forms depending on the adopted solution technique. A mathematical model developed in [6] is based on the row and column counters for each cell of the chessboard. The fitness function used in [7] is a matrix in canonical form which exhibits the placements of queens in the cells by 0 or 1 strategy. We develop two different types of mathematical models for an N queen problem in the following lines.

2.4 Single Objective Model (I)

First of all we allot the numbers to the cells from 1 to N^2 starting from any corner and then proceeding either row wise or column wise. For example if N = 4 then the chessboard will be taken of the form:

13	14	15	16
9	10	11	12
5	6	7	8
1	2	3	4

Figure 3: Chess-board for N=4

Secondly we use the variable $X_j : j = 1, 2, 3, \ldots, N.$ which gives the cell number of the chessboard where j^{th} queen is to be placed. This implies the fact that each X_j is a discrete integer variable ranging from 1 to N^2. Next requirement for constructing the model is to identify the strategic directions of each queen. Regarding the above assumptions, and following the idea used in [12], for j^{th} queen we find:

(i) The horizontal attack line number:

$$h_j = h(X_j) = \left\lfloor \frac{X_j + N - 1}{N} \right\rfloor$$

(ii) The vertical attack line number:

$$v_j = v(X_j) = 1 + (X_j - 1) \bmod N$$

(iii) The right diagonal number:

$r_j = N - (v_j - h_j)$

(iv) The left diagonal number:

$\ell_j = h_j + v_j - 1$

Each configuration of the chessboard is an N-dimensional vector $X = (X_1, X_2, X_3, \ldots, X_N) \in S$ where S is the discrete search space containing all possible permutations of N objects taken from a total of N^2 objects.

We define the objective function f(X) as:

$$Minimize\ f(X) = \eta\ (X),$$

where η(X) means the number of identical members of X.

Constraints: Along with the above objective function, the fact that every attack line of every queen should be strictly different from the relevant attack lines of every other queen implies the following constraints:

For all i, j = 1, 2, 3, … , N

Horizontal attack line constraint

$$h_i - h_j \neq 0 \quad \forall\, i \neq j$$

Vertical attack line constraint

$$v_i - v_j \neq 0 \quad \forall\, i \neq j$$

Right diagonal attack line constraint

$$r_i - r_j \neq 0 \quad \forall\, i \neq j$$

Left diagonal attack line constraint

$$\ell_i - \ell_j \neq 0 \quad \forall\, i \neq j$$

Bounds constraints

$$1 \leq X_j \leq N^2$$

And finally the discrete constraints

$$X_j \in Z^+$$

2.5 Multi-Objective Model (II)

The above model alternatively can be formulated as a Multi-Objective Optimization (MOO) model. For this purpose we set:

$$\boldsymbol{h} = [h_1, h_2, h_3, \ldots, h_N],$$
$$\boldsymbol{v} = [v_1, v_2, v_3, \ldots, v_N],$$
$$\boldsymbol{r} = [r_1, r_2, r_3, \ldots, r_{2N-1}]$$

and $$\boldsymbol{l} = [\ell_1, \ell_2, \ell_3, \ldots, \ell_{2N-1}]$$

Now the problem can be stated as:

Minimize $f_k(X) : 1 \leq k \leq 5$

Subject to $1 \leq X_j \leq N^2,$

$X_j \in Z^+,$

$\forall\, j = 1, 2, 3, \ldots, N$

Where $f_1(X) = \eta(X)$, $f_2(X) = \eta(\mathbf{h})$, $f_3(X) = \eta(\mathbf{v})$, $f_4(X) = \eta(\mathbf{r})$, $f_5(X) = \eta(\boldsymbol{l})$

3. RESULTS AND DISCUSSIONS

For solving the model II, it can be converted to a single objective optimization problem by the transformation

$$F(X) = \sum \lambda_i \cdot f_i(X)$$

Which is a weighted sum of f_i's and λ_i's are weights. As every f_i (X) has equal priority so we set each $\lambda_i = 1$. Obviously the global minimum value of the function so formed will be 0. To handle the boundary constraints we use an exterior penalty approach. In this way a penalized objective function of the form

$$\Psi(X) = F(X) + Penalty(X)$$

There are multiple optimal solutions for the above formulated problems for N ≥ 4. Especially model II is a new challenging multi-objective optimization problem having many alternative optima in Table. A further study is required to check whether the existing optimization tools have potential to find all optima of model II or not. But in this script we restrict our self to the formulations of the problem and the investigation of applicability of HJ to the prepared models.

Table -1: Alternative total and unique solutions [9]

N	Total Solutions	Unique Solutions
1	1	1
2	0	0
3	0	0
4	2	1
5	10	2
6	4	1
7	40	6
8	92	12
9	352	46
10	724	92
11	2680	341
12	14200	1787
13	73712	9233
14	365596	45752
15	2279184	285053
16	14772512	1846955
17	95815104	11977939
18	666090624	83263591
19	4968057848	621012754
20	39029188884	4878666808
21	314666222712	39333324973
22	2691008701644	336376244042
23	24233937684440	3029242658210
24	227514171973736	28439272956934
25	2207893435808352	275986683743434
26	22317699616364044	2789712466510289

We applied the modified HJ method to the above models. With various initial guesses the modified HJ algorithm found different optimal solutions of both the models for N=8. Figure 4 and 5 present two such solutions.

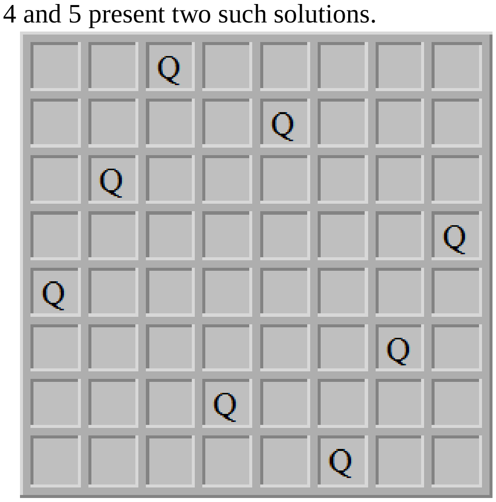

Figure 4: Optimal configuration for 8 queens.

Which means optimal solution is

$$X^* = [6\ 12\ 23\ 25\ 40\ 42\ 53\ 59]$$

With $f(X^*) = 0.$

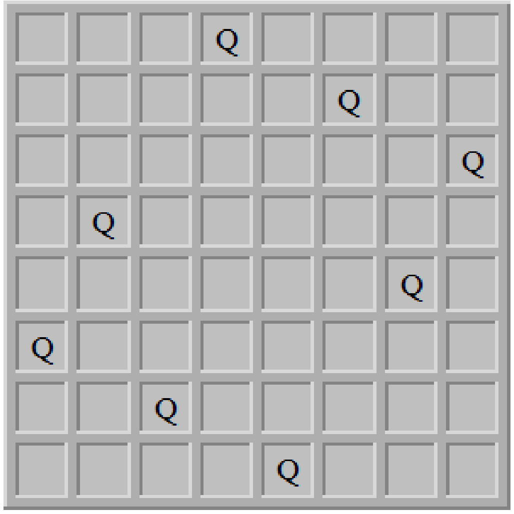

Figure 5: Alternative optimal configuration for 8 queens.

This gives the following optimal solution:

$$X^* = [5\ 11\ 17\ 31\ 34\ 48\ 54\ 60]$$

With $f(X^*) = 0$

4. CONCLUSIONS

A general chessboard configuration problem regarding positions of queens has been formulated in two forms. One formulation falls in the category of single objective optimization problems and the other one belongs to multi-objective optimization problems. According to the best of our knowledge model II is the ever first time proposed MOO test problem which involves all discrete variables, the discrete search space and multiple alternative optima. Moreover, in this study we also modified HJ method to solve the formulated configuration problems. The modified approach successfully produced optimal configurations for both the models.

REFERENCES

[1] N–Queen Problem in Cell: Project Report Arghya Dasgupta, Srinivasan Kannan. http://mc2.umbc.edu/docs/srini-dasgupta.pdf.

[2] Letavec, C. and Ruggiero, J. "The n-queen problem", *Informs Transaction on Education*, **2** (3): 101-103 (2002)

[3] Sosi, R. and J. Gu "Efficient local search with conflict minimization: A case study of the n queens problem," *IEEE Transactions on Knowledge and Data Engineering*, **6**(5): 61-68 (1994).

[4] Bitner, J.R. and E.M. Reingold "Backtracking programming techniques," *Communications of the ACM*, **18**(11): 651-56 (1975).

[5] Purdom, P.W. and C.A. Brown "An analysis of backtracking with search rearrangement," *SIAM Journal of Computing*, **12** (4): 717-33(1983).

[6] Hoffman, E.J., J.C. Loessi, and R.C. Moore "Constructions for the solution of the n queens problem," "*Mathematics Magazine*, **42**: 66-72. (1969).

[7] Amer D., S. Meshoul, H. Talbi, and M. Batouche, "A Quantum-Inspired Differential Evolution Algorithm for Solving the N-Queens Problem" *The International Arab Journal of Information Technology*, **7**(1): 21-28 (2010).

[8] Aftab A., A. Shah, K. A. Sani and A. H. Shah "Particle Swarm Optimization For N-QueensProblem", *Journal of Advanced Computer Science and Technology*, **1**(2): 57-63 (2012).

[9] Durango Bill's "The N-Queens Problem", http://www.durangobill.com /N_Queens.html

[10] Knuth D.E., "Dancing links, in Millenial Perspectives in Computer Science", *Proceedings of the 1999 Oxford-Microsoft Symposium in Honour of Sir Tony Hoare*, Palgrave (2000).

[11] Wirth, N. "Algorithms + Data Structures = Programs", *Prentice Hall*, Englewood- Cliffs (1976).

[12] Ahrabian H. "A DNA Sticker Algorithm For Solving N-queen Problem", *International Journal of Computer Science and Applications*, 5(2): 12–22 (2002).

[13] Cengiz E., S. Sarkeshik and M. M. Tanik "Different perspectives of the N-Queens problem", *Proceeding CSC '92 Proceedings of the ACM annual conference on Communications* 99 – 108 (1992).

[14] Robert M. L., V. Torczon and M. W. Trosset "Direct search methods: then and now", *Journal of Computational and Applied Mathematics* **124**: 191-207 (2000)

[15] Kelley C.T. "Iterative Methods for Optimization", *SIAM, Society for Industrial and Applied Mathematics* Philadelphia, Carolina, (1998).

[16] Robert J. G. "Interior Ballistics Optimization", ***A** thesis Department of Mechanical Engineering, College of Engineering*, Kansas Statz University, Manhattan, Kansas, (1990).

[17] Elizabeth D. D. "Pattern Search Behavior in Nonlinear Optimization", *A thesis College of William & Mary in Virginia*, http://www.cs.wm.edu/~va/CS495/dolan.pdf

[18] Lahouaria B., A. Belmadani and M. Rahli, "Hooke-Jeeves' Method Applied to a New Economic Dispatch Problem Formulation" *Journal of Information Science and Engineering*, 24: 907-917 (2008).

[19] Ong C. G., "Shaking and balance of a convertible one and two-cylinder reciprocating compressor," Master Thesis, *Faculty of the Virginia Polytechnic and State University*, Blacksburg, Virginia, (2000).

[20] Himmelblau D. M., "Applied Non Linear Programming", *McGraw-Hill*, New York, (1972).

[21] Luenberger D. G., "Linear and Nonlinear Programming", *Addison–Wesley*, London, (1984).

[22] Tabassum, M.F., M.Saeed, Nazir Ahmad and A. Sana, "Solution of War Planning Problem Using Derivative Free Methods". Sci.Int., 27(1): 395-398 (2015).