Journal of Advanced Computer Science and Technology, *1 (2) (2012) 57-63*
©Science Publishing Corporation
www.sciencepubco.com/index.php/JACST

Particle Swarm Optimization For N-Queens Problem

Aftab Ahmed, Attique Shah, Kamran Ali Sani and Abdul Hussain Shah Bukhari

Department of Computer Science, BUITEMS, Quetta
aftab.ahmed@buitms.edu.pk
Department of Information Technology, BUITEMS, Quetta
attique.shah@buitms.edu.pk
Department of Computer Science, BUITEMS, Quetta
kamran.ali@butims.edu.pk
Benazir Bhutto Shaheed University, Lyari, Karachi
ahsbukhari@yahoo.com

Abstract

The N-Queen is one of the prominent toy problems for performance assessment in the domain of Computational Intelligence. The problem greatly helps to examine the diverse aspects of solving techniques, counting the potential of obtaining feasible solution and computing time & space complexity. The well-defined set of constraints sketches out all the versions of problem. The Queen has various attacking options over the chessboard. In point of fact, it can walk off across the column, row and both diagonals. The feasible solution of N-Queens highly demands the non-attacking placements over the chessboard. This research inspects the effectiveness of Particle Swarm Optimization in order to produce optimal solution of the problem. In PSO, a particle contains a structured set of N-Queens placements. Current position of the particle is well directed by the *pbest* and *gbest* parameters, each new generation obtains more converged set of particles. The partial but improved outcome promotes and furnishes to succeeding generation while the substandard may be discarded. The investigational results make known the dynamic features of PSO and eventually validate the effectiveness of research direction.

Keywords: *N-Queens problem, PSO, Constraints.*

1 Introduction

The N-Queens is a famous performance evaluator and test problem in the field of intelligent computation to measure the capabilities of solving methods in terms of

various computing aspects. The technique is particularly effective for Constraints Satisfaction Problems (CSP). The CSP is formulated by set of quantifier (q_1, $q_2 \ldots q_n$), their finite domains such as (d_1, $d_2 \ldots d_n$) and the third component is a group of constraints for example (c_1, $c_2 \ldots c_n$). The key objective of CSP and such identical techniques of computing optimization is to acquire the best available outcome on the cost of reasonable time and resources. The inclination remains behind solving this toy problem is to surface out the merits and demerits of computational methods for matching real world problems. This article examines the efficiency of Particle Swarm Optimization in order to achieve feasible results for N-Queens problem.

Particle Swarm Optimization belongs to group of computational intelligence. It imitates the characteristics of birds swarm, looking for best possible food sources. The movement of a single particle is preceded inclination of its last position, its calculated knowledge, and the swarm's collective intelligence. So the entire swarm is converged by information sharing approach. First of all the idea of basic PSO is coined by Kennedy and Eberhart [1] which is illustrated as following.

$$x_k^i = x_k^i + v_{k+1}^i \quad (1.1)$$

$$v_{k+1}^i = \omega v_k^i + c_1 r_1 (p_k^i - x_k^i) + c_2 r_2 (p_k^g - x_k^g) \quad (1.2)$$

Where *x* stand for an individual (particle), *i* places for the identity of the individual, *k* demonstrate iteration, and *v* is the particle's velocity. $c_1 r_1 (p_k^i - x_k^i)$ points the best position of the particle ever visited (Individual experience and Intelligence), and $c_2 r_2 (p_k^g - x_k^g)$ stands for the feasible position of any particle in the swarm ever visited (Collective experience and intelligence) . ω corresponds to weight of inertia which influence the velocity, c_1, c_2 effects over individual (particle) and entire swarm's knowledge respectively. r_1, r_2 are two random constants expedite the process and helps to avoid trapping into local maximum.

2 Related Work

A plenty number of solving technique are implemented over the N-Queen problem. Aftab [2] investigated the two distinguished heuristics algorithms to resolve the N-Queens problem. The work instills the improved version of Backtracking Recursive and Modified Min-Conflicts. The experimental results revealed dynamic features of implemented techniques. Crawford [3] applied Genetic Algorithm with various patterns to resolve the issue. He concluded his research work with optimized outcome obtained. Constraints Logic Programming (CLP) is very suitable option to solve the N-Queens problem. The CLP is defined as *"a straightforward statement of the constraints serves as part of the program.*

This makes the program easy to modify, which is crucial in real-world problems". A constraint is a logical relation among variables where each variable represents its domain values. In general, a constraint is kind of filtering that refines status of state among a range of available options. A wide range of problems are addressed by CLP approach. A university timetabling problem is successfully solved by CLP algorithms[4]. Their research approach was consisted of two distinguishable phases. First one used to remove all hard constraints and second phase was responsible to tackle and minimize soft constraints. Backtracking algorithm and Min-conflict Algorithm were implemented to acquire the said task. Russell [5] stated that a 8-Queens problem is good benchmark problem for evaluating CSP with *64* squires and maximum eight queens placement availability. The key criterion is as no queen supposed to attack on any other queen. There may have plenty of solution patterns can be comprised with diverse deployment of queens. The problem somewhat imitates the hard constraints satisfaction of scheduling problem[6], but in queens problem no ultimate feasible solution is acceptable. A CSP approach assigns values from relevant domains according to prescribed constraints; the notion is very identical as scheduling problem. Zhang [7] distinguishes; "*there are two approaches to solving CSP. One is using the search algorithms and the other is using the consistency technique. Constraints logic programming deals with plenty of problem including famous NP-hard set of problems. CSP combined different techniques to produce optimal solutions which would be acceptable on some certain measures".*

3 Problem Description

In the chess game Queen is very powerful piece that can approach with multi-directional aspects of horizontal, vertical and diagonal in order that a handful corners remain protected for opponent pieces.

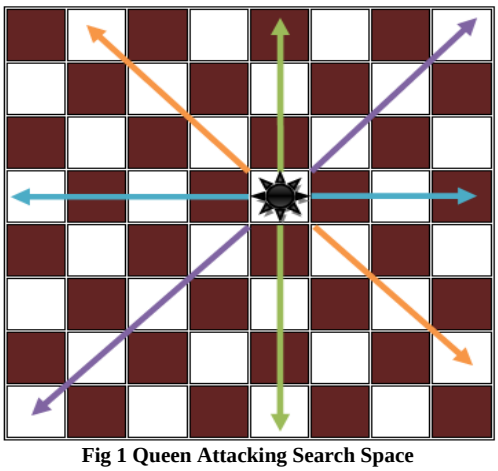


Fig 1 Queen Attacking Search Space

The conventional N-Queen encompasses the assignments of fixed number of Queens over $\boldsymbol{n \times n}$ chessboard in a way no queen access to other. In brief, same

row (horizontally), column (vertically) or diagonal (Crossways) should not be common between two Queens. Figure No. 1 reveal the search-space of the problem. A certain number of Queen Problems can have a wide range of solutions for instance a chessboard of *8x8* provides 92 sets of different solutions.

3.1 Particle Swarm Optimization

Particle Swarm Optimization (PSO) is natural imitation of birds-swarm and population (group of particles) based solution for various optimization problems. The idea was firstly coined by Eberhart and J. Kennedy [1] the metaphor is inspired from individual and cooperative behavior of bird swarm, fish schooling and dust particles swirl. The generally such cooperative social movement takes place for food searching and exhibits as big entity for showing off vigorous look to opponents. The basic resemblance is adapted from the leaning of bird (individual) to arrive at leading position of the swarm. The inclination of each particle is to arrive collision-free closeness with neighbors, every particle tries to improve its current placement with reference to its previous best ever position. Figure 2 briefly exhibits the PSO methodology.

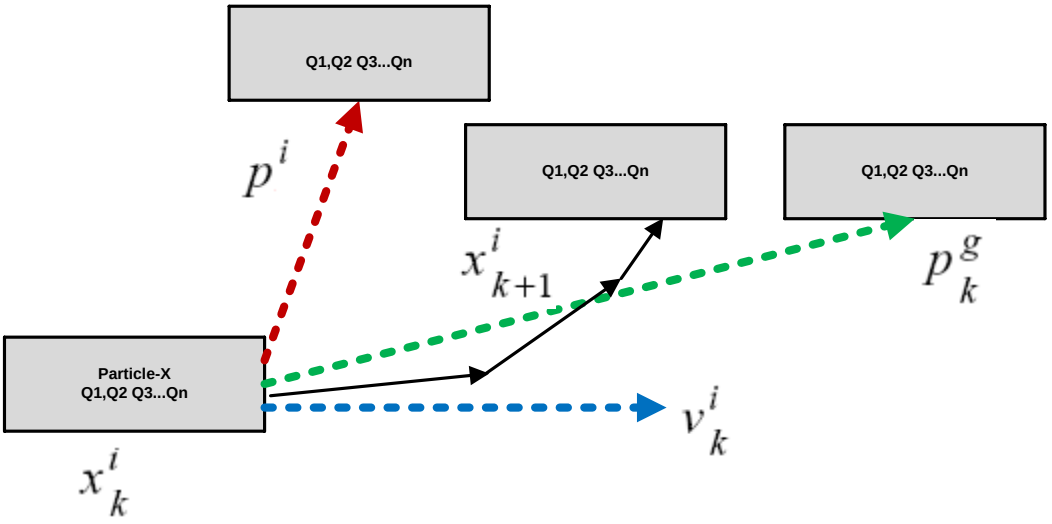


Figure 2 PSO mechanism over N-Queens problem

Algorithm No. 1 demonstrates the typical Particle Swarm Optimization (PSO). First step initializes each practical with random placement over the search space, such assignments of particles over state-space greatly supports PSO not to be trapped into local-maximum dilemma and it expedites the conversion rate as well. The other parameters include population size, communication topology, inertia weight and influence rate of cognitive and social coefficient. Generally, parameters for performance measures varies cast to case, however they can be tuned up with several attempts of slight changes. Every individual (particle) in generation is representing partial solution of N-Queens. Towards the end of each

genome the PBEST and GBEST particles come to surface due to fitness level and provide direction for rest of particles to an optimal solution.

Algorithm 1: Standard Particle Swarm Optimization

1. ***Initialization:***
    a. Generate First Swarm by random position $\forall X_i$
    b. Initialize the velocity V_i with smallest random number
2. ***Evaluate Function:*** Calculate the fitness Level of Particles
3. ***Loop Until*** (Termination Criteria)
4. ***For i range (1,PopulationSize)***
    a. $V_i^{k+1} \leftarrow \omega V_i^k + \alpha R_i (Pbest_i - X_i^k) + \beta R_i (Gbest_i - X_i^k)$
    b. $X_i^{k+1} \leftarrow X_i^k + V_i^{k+1}$
5. ***Revaluate Function:***
    a. If $f(X_i^k) < f(Pbest_i)$ than $X_i^k \leftarrow Pbest_i$
    b. If $f(Pbest_i) < f(Gbest_i)$ than $Pbest_i \leftarrow Gbest_i$
6. ***End Loop***

- W represents weight of inertia usually contain value between 0.4 to 1.4
- V^{ik} is the velocity of practical
- C^1 is constant, used to expedite individual movement, 1.5 to 2 is recommended value.
- C2 is recognized as second constant which can have value between 2 to 2.5, this speeds up the swarm influence (Collective movement)
- *PBest* is best ever recorded position of individual in swarm
- *GBest* is best ever recorded position of group which influence each particle to swarm heart.

4 Main Results

The computational approach is implemented over reasonable size of chessboard and the quality results revealed the positive impact and capability of adapted technique. The script was written on Intel® Core(TM) i3 CPU, M 350 @ 2.27 GHz, 2.0 GB RAM. The Python language version 2.6 was chosen to compile the program coding. Table 1 specifically shows the adapted parameters of PSO that were found suitable for this case study. The dataset initialization deliberately scatters up the particles over entire search-space.

Table 1 PSO parameters

	Parameter	Value
1	topology	Ring topology
2	Neighborhood size	20
3	Max Evaluations	1000
4	Number of elites	0
5	Population Size	10
6	evaluator	Binary
7	Termination Criteria	Converging of Population

Table 2 is providing precise details of outcome on each 10^{th} generation. The partially satisfied outcome is showing off the obvious direction to quality improvement. At the end of each generation the tentative outcome moves to succeeding generation. Each step makes some finer changes over chessboard layout.

Table 2 Outcome of 20x20 chessboard using GA

Generations	Fitness *f(x)*	Time
10	52.00	0:00:00
100	45.00	0:00:06
200	39.00	0:00:12
300	33.00	0:00:18
400	31.00	0:00:24
500	22.00	0:00:31
600	18.00	0:00:39
700	11.00	0:01:04
800	09.00	0:01:22
900	05.00	0:01:41
1000	0.00	0:02:02

Table 2 demonstrate the PSO generated outcome, it can be observed the results converged to zero conflict in reasonable number of populations (iterations) and time.

5 Conclusion

The research paper examined the Particle Swarm Optimization (PSO) approach that set up the stunning results at the end, and also provided evidence to be very capable of producing feasible/optimal solutions over N-Queens Problem. The key objective of the study is to reveal the effectiveness and efficiency of PSO so it can be projected to other identical real-world problems such university timetabling etc.

Acknowledgements

We would like to convey our gratitude to the competent authority of BUITEMS for providing the financial means, great research environment and computational and infrastructural facilities.

References

[1] J. Kennedy and R. C. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, 1995, pp. 1942-1948.

[2] Aftab Ahmed, *et al.*, "Constraint Logic Programing For Solving N-Queens Problem," in *Second International Conference on Computer & Emerging Technologies (ICCET 2012)*, Shah Abdul Latif University, 2012, pp. 15-19.

[3] K. D. Crawford, "Solving the N-Queens Problem Using Genetic Algorithms," in *ACM/SIGAPP Symposium on Applied Computing*, 1992, pp. 039-1047.

[4] Aftab Ahmed, *et al.*, "Solving University Scheduling Problem Using Hybrid Approach," *Mehran University Research Journal of Engineering & Technology* vol. 30, pp. 613-624, October 2011.

[5] S. Russell and P. Norvig, *Artificial Intelligence A Modern Approach* Prentice Hall, 2003.

[6] Aftab Ahmed, *et al.*, "Hyper-GA for Solving Benchmark Scheduling Problems," *Australian Journal of Basic and Applied Sciences*, vol. 5, pp. 1657-1667, June 2011.

[7] Lixi Zhang and S. Lau, "Constructing university timetable using constraint satisfaction programming approach," in *International Conference on Computational Intelligence for Modelling, Control and Automation and International Conference on Intelligent Agents, Web Technologies and Internet Commerce*, 2005, pp. 55-60.