

# Intelligent Heuristic Search Algorithm for N Queens Problem of constraint satisfaction

<sup>1</sup>JAGDISH PRASAD, <sup>2</sup>A. K. BHARDWAJ, <sup>3</sup>SURENDRA KUMAR YADAV

<sup>1</sup>University of Rajasthan, Jaipur-302055

<sup>2</sup>University of Rajasthan, Jaipur-302055

<sup>3</sup>Department of Computer science and Engineering, JNIT, Jaipur

## Abstract

In this paper we have discussed variant of systematic and repair strategies for N queen's problem for different positions and size of board of problem space. We introduce the intelligent Heuristic search algorithm for solving the N queen's problem with different size of board positions. The intelligent Heuristic search algorithm, that we propose here is based on major part of local search methods and backtrack systematic search. The algorithm is more interactive behavior in the strategy of changing the task during the search. Algorithm separates the hard and soft constraints and all the hard constraints have to be completely satisfied while the soft constraints do not required being satisfied. we compare the produced result with another systematic search algorithm and analysis their results and performance.

**Keywords-** Constrains, Algorithms, Repair, Backtracking, L Heuristic, State.

## 1. Introduction

In this paper we indicate the problem of constraint satisfaction with N queens' problem. A constraint satisfaction problem is a problem where one has to find a value for a finite set of variables satisfying a finite set of constraints (Freuder and Mackworth 1994) (Mackworth 1997) (Tsang 1993). Research in this field involves finding methods to solve such problems efficiently. Constraints can be found in many places in daily life like, rules and restrictions, requirements, machine capacity and preferences are all constraints. One major application is scheduling. The variable needs to store values from their respective domains and solution of problem which are define in term of CSP as the assignment of a value to all variables in such a way that no constraint would be violated. In CSP every variable have the define domain of the possible values and variable hold the value only from the defined domain.

### 1.1 What is N-Queens Problem

A commonly used example of CSP is the N-queens problem. The N-queens problem based on domain of a chess board of

$N \times N$  squares using N queens as pieces and as the chess game, queens threaten other pieces horizontally, vertically and diagonally. The goal of a game is to place all queens on the board so that they do not threaten each other.

The constraints defined by the N-queens problem are:

- No two queens may be placed in the same row.
- No two queens may be placed in the same column.
- No two queens may be placed diagonally from each other.
- No two queens occupy the same square on the game board.

First of all we must identify a set of variables to formalize N-queens problem of CSP. For that purpose the 8-queens, problem as a CSP is to make each of the 8 rows and the 8 queens, problem a variable. The formal set of variable is,

$$N = \{ \theta_1, \theta_2, \theta_3, \dots, \theta_8 \}$$

Each variable of set can take one of the eight columns as its value.

$$D\theta_1 = \{1, 2, 3, 4, 5, 6, 7, 8\}$$

$$D\theta_2 = \{1, 2, 3, 4, 5, 6, 7, 8\}$$

⋮

$$D\theta_8 = \{1, 2, 3, 4, 5, 6, 7, 8\}$$

According to the rule of problem each row as a variable has ensured that no two queens can be on the same row. To show that condition as a formal rule.

$$\forall -i, j; \theta_i \neq \theta_j$$

Now let us look for second rule of N-queens problem that no two queens are on the same diagonal.

$\forall i, j$  if  $\theta_i = x$  and  $\theta_j = y$

Then  $i - j \neq y - x$

There are two basic class of strategy for N-queens CSP problem.

### 1.1 (a) Systematic Search Strategies

In systematic search strategies we put one queen onto the chess board at a time and make sure that no constraint is violated, until 8 queens are placed on chess board. Use the backtracking if at any point are cannot find a safe place for a queen. If the squares are tried systematically, all possible board situations will be tried if necessary.

### 1.1 (b) Repair Strategies

In the repair strategies put all 8 queens onto the board on the basis of random choice and if any queen threatens by another queen then try to move it to a new position on the board. There is a possibility that solution can eventually be achieved.

In our testing we are using various systematic and repair search for solving the 8-queens problem. The following are used to solve the 8-queens problems as systematic search strategies:

1. Backtracking Search
2. Look ahead search
3. Evolutionary Algorithm
4. Iterative broadening algorithm (Book-A/121)
5. PCSP Branch and bound algorithm.

## 2. Intelligent Heuristic Search Algorithm

Systematic search strategies produce any solution or all solution of N-queens problem and these algorithms also divided in subgroups. These systemic search methods explore systematically the complete search space. But incomplete search methods or repair strategies do not explore the complete search space. Their non-systematic nature produces the voids results of completeness but their computational time is reasonably reduced. That's why; these algorithms sufficient when just same solution is needed.

The systematic search have the demerit of the free search and if any wrong decision produce by algorithm then backtracking is necessary but in local search methods, when a problem is tightly constrained, solution not produced by these. The above mentioned discussion produce facts that the different features of systematic and non-systematic search

methods are complementary on bases of characteristics of both systematic search algorithm and repair search algorithm we developed a new search algorithm that is names as "*Intelligent Heuristic Search Algorithm*." The intelligent Heuristic search algorithm, that we propose here is based on major part of local search methods and backtrack systematic search. The algorithm is more interactive behaviors in the strategy of changing the task during the search. We propose an interactive algorithm that works in efficient iterations and used a set of various and partial complete solution. Every iteration of intelligent heuristic search algorithm is tried to improve the partial complete solution of previous iteration. We can also test the solution after the assigned variables during the iterations of algorithms. Algorithms used a function that selects an unassigned variable to be assigned in the current iteration step. The process of selecting variable could be expensive in some cases due to complexity of computing and used algorithm. That's why, we can select a subset of unassigned variables randomly and select worst variable from this subset.

The intelligent heuristic search algorithm used two different functions. First function used for variable selection and the second function used for value selection. These functions are combined characteristics of backtracking search and local search with new developed strategies. We can also used information related to the previous value of variables in selection of non-assigned variables in execution process. In every iteration of loop the first function used that select an unsigned variable to be assigned. After the selecting a variable we are required to complete the process of value selection by the second function. By the function algorithm tries to final most preferred with minimal potential future conflicts values for the variable and also which cause the least problem. Algorithm used intelligent heuristic so that it is possible to apply randomize the value selection strategies or we can say that it is possible to select a set of values. Algorithm separates the hard and soft constraints and all the hard constraints have to be completely satisfied while the soft constraints do not required being satisfied.

The method to select a variable involves heuristic with order in which the variables are instantiated. Instead of doing this randomly the sequence of initiations can be ordered and it can either be done globally before the search starts or locally at every node. In the N-queens problem for instance this would lead to an ordering from the middle rows outward, since a queen in the middle row bounds the search more than one on the top or bottom of the board. The order of the variable dynamically determined at each node of the free and this type of selection called local selection. The second function of algorithm, select-value checks all constraints that refer to the current variable, previous variables and forthcoming variable.

### 3. Result of Algorithm

In this part of chapter we will present the efficiency of the described in intelligent heuristic search algorithm on the N-queens problem and we compare the produced result with another systematic search algorithm.

Size N	No Of Solution	Time (Sec.) Avg. Time		Intelligent Heuristic Search Algorithm
		Systematic Search Algorithm	Repair Search Algorithm	
8	92			
9	352			
10	724	.002	.002	
11	2680	.006	.004	
12	14200	.009	.008	
13	73712	.010	.013	.001
14	365596	.33	.29	0.2
15	2279184	2.10	2.29	1.9
16	14772512	13.89	13.66	11.2
17	95815104	75.01	75.04	77.2
18	666090624	573	571	576

A number of experiments have been done to evaluate the algorithm and compared with other algorithm. We have compared the efficiency of the described intelligent heuristic search algorithm with discussed other algorithms.

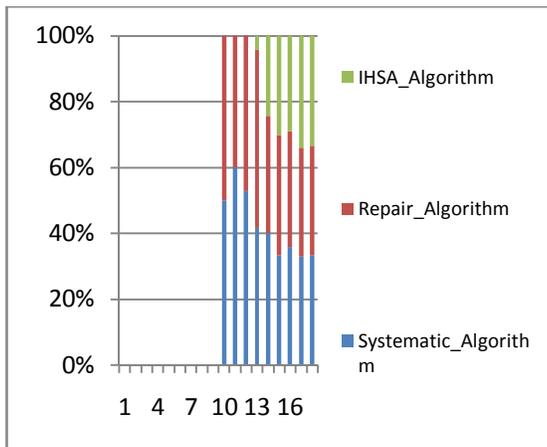


Fig 1: shows the performance of algorithms for constraint satisfaction problem.

### Conclusion:

A wide classification of various algorithm to solve constraint satisfaction problem is proposed and an important section is devoted to the algorithm which try to find a maximal partial solution. The various papers about csp problem based algorithm were studied together with standard text books. All discussed code of algorithm are given in uniform, non-recursive pseudo code and based on

the back tracking search strategies with constraint propagation. As a part of future work will include a positive application to minimal perturbation problem and possible extension of concept include verification on other types of problem. The basic goal was to design an algorithm named intelligent heuristic search algorithm for solving complicated n-queens problem with different size of board which their complete solution could not be found in a reasonable time.

### References:

- 1] Bernhardsson, B. (1991), "Explicit solutions to then-queens problems for all n," ACM SIGART Bulletin, vol. 2, no. 7.
- 2] Robert A. Bosch. Peaceably Coexisting Armies of Queens. Optima (Newsletter of the Mathematical Programming Society),62:6{9, 1999.
- 3] Niklaus Wirth. Algorithms + Data Structures = Programs. Series in Automatic Computation. Prentice-Hall, 1976.
- 4] Barbara M. Smith, Karen E. Petrie, and Ian P. Gent. Models and Symmetry Breaking for Peaceable Armies of Queens.
- 5] Presented at the ECAI 2002 Workshop W9 on Modelling and Solving Problems with Constraints, 2002.
- 6] Foulds, L.R. and D.G. Johnston (1984), "An application of graph theory and integer programming: Chessboard non-attacking puzzles," Mathematics Magazine, Vol. 57, No. 2, pp.95-104.
- 7] Sosi, R. and J. Gu (1994), "Efficient local search with conflict minimization: A case study of the nqueens problem," IEEE Transactions on Knowledge and Data Engineering, Vol. 6, No. 5, pp. 61-68.
- 8] Kale, L.V. (1990), "An almost perfect heuristic for the n non-attacking queens problem," Information Processing Letter, Vol.34, pp.173-78.
- 9] Mandziuk, J. (1995), "Solving the n-queens problem with a binary Hopfield-type network. Synchronous and asynchronous model," Biological Cybernetics, Vol. 72, No. 5, pp. 439-46.
- 10] Purdom, P.W. and C.A. Brown (1983), "An analysis of backtracking with search rearrangement," SIAM Journal of Computing, Vol. 12, No. 4, pp. 717-33.

11] Shagrir, O. (1992), "A neural net with self-inhibiting units for the n-queens problem," International Journal of Neural Systems, Vol. 3, No. 3, pp. 249-52.