A Study on the Impact of Heuristic Initialization Functions in a Genetic Algorithm Solving the N-Queens Problem

Eneko Osaba
Deusto Institute of Technology
University of Deusto
Bilbao, Spain
e.osaba@deusto.es

Fernando Diaz
Telecommunications Dept.
University of Deusto
Bilbao, Spain
fernando.diaz@deusto.es

Roberto Carballedo
Deusto Institute of Technology
University of Deusto
Bilbao, Spain
roberto.carballedo@deusto.es

Enrique Onieva
Deusto Institute of Technology
University of Deusto
Bilbao, Spain
enrique.onieva@deusto.es

Pedro Lopez
Deusto Institute of Technology
University of Deusto
Bilbao, Spain
p.lopez@deusto.es

ABSTRACT

In this paper the influence of using heuristic functions to initialize the population of a classic genetic algorithm (GA) applied to the N-Queens Problem (NQP) is analyzed. The aim of this work is to evaluate the impact of the heuristic initialization phase on the results of the classic GA. In order to probe this, several experiments using two different initialization functions have been carried out. In this paper, the well-known NQP has been used as benchmark problem, but the objective of the authors is to contrast the findings of this study with other combinatorial optimization problems.

Categories and Subject Descriptors

I.2.8 [**Artificial Intelligence**]: Problem solving, Control Methods and Search—*Heuristic methods*

General Terms

Algorithms

Keywords

Genetic Algorithm, Initialization, N-Queens, Combinatorial Optimization.

1. INTRODUCTION

Thanks to its robustness and its capacity of adaptability to a wide variety of areas, genetic algorithms (GA) [2] are one of the most used meta-heuristics in the literature. Nowadays, a lot of research studies related to this kind of algorithm can be found. Something to keep in mind when developing genetic algorithms, is the way wherein the population is initialized. There are three ways to perform this process: randomly, by using initialization functions, or by a combination of both alternatives. When the GA is applied to solve a very complex problem, the first approach is the most used [6]. It is also the usual alternative when a GA is compared with other meta-heuristics [5]. On the other hand, heuristic initializaton functions are usually used when the goal is to obtain the best possible solution for a given problem [4].

In this paper, the influence of using heuristic functions to initialize the population of a classic GA is analyzed. For this purpose, the N-Queens Problem (NQP), and two heuristic functions has been used for the experimentation.

ACM X-XXXXX-XX-X/XX/XX.

2. N-QUEENS PROBLEM

The NQP consists of placing N queens on a NxN chessboard, in order that they cannot attack each other. This is a classical configuration problem, but it can be also formulated as a combinatorial optimization problem [3]. In the experimentation presented in this paper, a solution is encoded as a N-tuple $(q_1, q_2, ..., q_n)$, which is a permutation of the set $(1, 2, ..., N)$. Each q_i depicts the row occupied by the queen positioned in the ith column. With this representation, horizontal and vertical collisions are avoided. Thereby, the objective function of the NQP is defined as the number of diagonal conflicts along the board. Notice that ith and jth queens collide diagonally if:

$$|i - q_i| = |j - q_j| \qquad \forall i, j : \{1, 2, ..., N\}; i \neq j \tag{1}$$

Therefore, the objective of the NQP is to minimize the number of collisions, being 0 the optimum fitness. This same formulation is frequently used in the literature.

3. EXPERIMENTATION SETUP

For the experimentation two different heuristic initialization function has been used. Both of them are based on the Minimum Conflicts method [4]. The first one, called Order Minimum Conflicts (OMC), begins building a solution placing the first queen in a random row of the first column. Then, iteratively, it inserts the following queen in the next column, in that position which involves the smaller number of conflicts. If two or more position involves the same number of conflicts, the position is chosen randomly between them. The second function, called Random Minimum Conflicts (RMC), inserts the queens following the same procedure as the OMC, but selecting at random the rows in which the queens are inserted.

OMC	GA_0			GA_{10}			GA_{50}			GA_{100}		
Name	Avg.	S. dev.	Time	Avg.	S. dev.	Time	Avg.	S. dev.	Time	Avg.	S. dev.	Time
20-Queens	1.6	1.2	0.01	1.8	0.8	0.03	**1.4**	0.6	0.02	1.6	0.6	0.01
50-Queens	8.7	2.2	0.02	5.5	1.2	0.13	5.0	1.0	0.03	**4.3**	0.8	0.09
75-Queens	14.9	2.8	0.18	8.4	1.8	0.37	**7.4**	1.3	0.22	8.0	1.8	0.26
100-Queens	22.7	2.9	1.38	10.3	1.2	0.71	**8.8**	2.0	0.63	9.2	1.4	0.65
125-Queens	28.1	3.9	2.78	12.2	2.3	1.83	**10.0**	1.6	1.32	10.3	1.3	1.52
150-Queens	37.9	5.4	5.76	14.9	2.3	2.79	12.6	1.3	2.00	**12.0**	1.4	2.10
175-Queens	42.7	6.1	12.01	16.7	2.9	4.40	**12.4**	2.5	3.97	13.0	1.3	3.44
200-Queens	51.4	6.9	17.50	18.8	2.4	4.94	15.2	1.7	5.02	**14.6**	1.9	5.28
225-Queens	55.0	6.4	26.75	18.8	2.9	7.91	16.4	1.8	7.36	**15.8**	2.6	6.78
250-Queens	64.6	7.0	35.39	20.4	2.9	9.22	**16.0**	2.7	9.79	17.2	2.4	8.85
RMC	GA_0			GA_{10}			GA_{50}			GA_{100}		
20-Queens	1.6	1.2	0.01	1.5	1.2	0.02	**1.3**	0.5	0.02	1.6	0.8	0.02
50-Queens	8.7	2.2	0.02	5.4	1.1	0.15	**4.8**	1.5	0.04	5.1	1.3	0.08
75-Queens	14.9	2.8	0.18	8.9	2.1	0.31	7.9	1.7	0.28	**7.3**	1.5	0.31
100-Queens	22.7	2.9	1.38	10.1	1.4	0.65	**8.3**	2.4	0.75	8.8	1.2	0.71
125-Queens	28.1	3.9	2.78	11.9	2.8	1.72	9.8	2.1	1.41	**9.2**	1.7	1.48
150-Queens	37.9	5.4	5.76	15.2	2.9	3.10	12.9	1.9	2.11	**12.3**	2.1	2.11
175-Queens	42.7	6.1	12.01	16.5	2.7	4.33	**12.7**	2.9	4.04	13.5	1.4	3.28
200-Queens	51.4	6.9	17.50	18.9	2.5	4.78	15.1	1.8	7.66	**14.2**	2.2	5.30
225-Queens	55.0	6.4	26.75	19.2	3.5	8.12	**15.8**	2.1	7.49	16.3	2.6	6.82
250-Queens	64.6	7.0	35.39	20.5	3.0	9.05	**16.1**	3.2	9.80	16.9	2.5	8.71

Table 1: Results of the three techniques applied to the NQP

Thereby, two experiments have been performed, each one with a different initialization function. For each experimentation, four different versions of the GA have been developed. These variants have been called GA_α, where α is the percentage of the population created using heuristic initialization functions. In this case, α takes four values: 0, 10, 50, 100. Every GA has a population composed by 50 individuals. In addition, the crossover probability has been set in 90%, and the mutation probability in 10%. Regarding the selection and survivor functions, a 50% elitist-random function has been used. About the ending criteria, the execution finishes when there are $n + \sum_{k=1}^{n} k$ generations without improvements in the overall fitness of the population, being n the size of the problem. Finally, the well-known order crossover [1] and the swapping function have been used as crossover and mutation operators, respectively.

4. RESULTS

For each experimentation ten different instances has been employed, and every instance has been executed 20 times. In Table 1 the average results, standard deviation and average runtime (in seconds) are shown. It is noteworthy that, in order to avoid redundant tests, the results depicted for GA_0 are the same in both experimentations.

First of all, it can be concluded that the use of initialization functions helps to reach a better solution. Regardless the proportion of the population initialized by heuristics, the results of GA_0 are the worst in all the instances. Anyway, this conclusion is logical, given that the individuals of the other versions of the GA go through an optimization process.

Secondly, it can be seen that GA_{50} obtains better results in the 60% of the cases (12 out of 20). This fact can be reasoned as follows: the overuse of heuristic initialization functions carries a decrease in the exploration capability of the technique and in the diversity of the population. On the other hand, maintaining a balance as the presented in the GA_{50}, the search can benefit from the previous optimization applied to the 50% of the individuals. Meanwhile, the diversity of the population is maintained by the remaining 50% randomly generated individuals.

5. CONCLUSIONS

In this work, the impact of the heuristic initialization phase on the results of the classic GA has been evaluated. For the tests, the well-known NQP has been used as benchmark problem. Anyway, the objetive of the authors is to extend this study to other combinatorial optimization problems. This extension has been planned as future work.

6. REFERENCES

[1] L. Davis. Applying adaptive algorithms to epistatic domains. In *International joint conference on artificial intelligence*, volume 1, pages 161–163, 1985.

[2] J. H. Holland. *Adaptation in natural and artificial systems.* Michigan Press, 1975.

[3] X. Hu, R. C. Eberhart, and Y. Shi. Swarm intelligence for permutation optimization: a case study of n-queens problem. In *Proceedings of the IEEE Swarm Intelligence Symposium*, pages 243–246, 2003.

[4] E. Masehian, H. Akbaripour, and N. Mohabbati-Kalejahi. Landscape analysis and efficient metaheuristics for solving the n-queens problem. *Computational Optimization and Applications*, 56(3):735–764, 2013.

[5] E. Osaba, F. Diaz, and E. Onieva. Golden ball: a novel meta-heuristic to solve combinatorial optimization problems based on soccer concepts. *Applied Intelligence*, pages 1–22, 2014.

[6] J. A. Ventura and S.-H. Yoon. A new genetic algorithm for lot-streaming flow shop scheduling with limited capacity buffers. *Journal of Intelligent Manufacturing*, 24(6):1185–1196, 2013.