EFFECT OF COMPUTATION TIME, MEMORY USAGE AND NUMBER OF NODES VISITED ON THE PERFORMANCE OF SOME SELECTED HEURISTIC ALGORITHMS

Olabode A. O., Fenwa O. D.

Department of Computer Science and Engineering, Ladoke Akintola University of Technology, Ogbomoso, Nigeria

Corresponding Author: Olabode A. O., anthony2olabode@gmail.com

ABSTRACT: The efficiency with which searching is carried out often has significant impact on the overall efficiency of a program. Some of the factors affecting the performance of informed tree based or heuristic search algorithms include high exponential execution time to search, drastic memory or storage usage and number of nodes visited. However, prioritizing each of these factors based on their influence has been a major challenge. Therefore, this research prioritized computational time, memory usage and number of nodes visited based on their influence using factor analysis by principal component.
KEYWORDS: Heuristic algorithms, Factor analysis, Principal component, computational time, memory usage and number of nodes visited.

1. BACKGROUND OF THE STUDY

Searching for data is a fundamental operation in computing. Often, what distinguishes a fast program and slow one has to do with the use of a good search algorithm for the dataset [O+13]. A searching method is informed or heuristic if its uses additional information about the nodes that have not yet been explored to decide which node to be examined next. Heuristic algorithms make use of problem specific knowledge so as to find efficient solution [PT04]. These techniques make use of evaluation function in determining the next best possible state leading towards the goal state. Some of the selected algorithms include best first, A* search and hill climbing techniques. Heuristic algorithms have exponential time and space complexities as they store complete information of the path including the explored intermediate nodes [BH10].
The heuristic algorithms have been found to be very applicable in areas such as artificial intelligence, robotics, networking [RN10]. It is used in solving routing problem, Travelling Salesperson Problem (TSP); touring problem in which each city must be visited exactly once and the aim is to find the shortest tour, these techniques have been used for tasks such as planning movements of automatic circuit- board drills and stocking machines on shop floors [RN10].
The Best-first search only compares the heuristic value of nodes, ignoring the path cost. When it expands a node that is closest to the goal, it gets a list of possible successors that were not explored before. Best first search derive the heuristics value of each of the successors and pick the best one to expand; the other leaves will be unexplored. It is not optimized, as the cheapest path may involve going through heuristically suboptimal nodes but yet have less path cost, best first search has the same mode of operation as the A* search [KT14].
A* heuristic search combine the value of heuristic function *h(n)* and the cost to reach the node *n*, *g(n)*. A* algorithm uses an evaluation function that accounts for the cost from the initial state to the current state, and the cost from the current state to the goal state [CGR96]. It avoids expanding paths that are already expensive [Yol01]. A* search is similar to uniform cost except that A* uses summation of *h(n) and g(n)* instead of *g(n)* [RN10].
Hill climbing expands only the best node reachable from current node. This method does not involve complex computation and due to this reason cannot ensure the completeness of the solution. Hill climbing method does not give an optimal solution as may terminate without reaching the goal state [C+99]. Hill climbing techniques work in a similar way to greedy search but totally disregard the memory of explored nodes. Computational time, memory usage and number of nodes visited were the three factors used in evaluating the performance of the selected heuristic algorithms.
However, the previous research works focused so much on the performance of the heuristic algorithms in relation to execution time, memory consumed and number of nodes visited but little justification was done in prioritizing execution time, memory used and number of nodes visited based on their influence. Therefore, this research prioritized execution time, memory usage and number of nodes visited based on their influence using factor analysis by principal component.

2. MATERIALS AND METHODS

The decision variables of impact of computational time, memory usage and number of nodes visited are interrelated. The performance of one factor affects its efficiency in another factor. Following Olabiyisi *et al.* [O+13], the general form of mathematical model for evaluating the decision variables is presented as;

$$y_i = \sum_{k=1}^{n} a_{i,k} x_k \qquad i = 1,2,3\ldots,m \qquad (1)$$

where y_i represents the i^{th} heuristic algorithms observation of k^{th} decision variable a_i, k_j represents the assessment of k^{th} decision variable by i^{th} heuristic algorithms. From equation (1) the mathematical model can be expressed by the system of equations;

$$\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ \cdot \\ \cdot \\ y_m \end{pmatrix} = \begin{pmatrix} a_{1,1}x_1 + a_{1,2}x_2 + a_{1,3}x_3 \ldots + a_{1,5}x_5 \\ \cdot \\ \cdot \\ a_{m,1}x_1 + a_{m,2}x_2 + a_{m,3}x_3 \ldots + a_{m,5}x_5 \end{pmatrix} \qquad (2)$$

Where x_j, $j = 1, 2, 3,\ldots,n$ represent j^{th} decision variable, $a_{i,j}$ represent the assessment of the i^{th} informed search techniques for the j^{th} decision variables.

Factor analysis by principal components of the obtained experimental data was carried out using SPSS 23.0 for the purpose of estimating the contribution of each factor to the success of the heuristic algorithms and validation of the most critical factor. The following statistics were generated and used for the stated objectives; descriptive statistics, communalities, Bartlett's test of sphericity, kayer-olkin mayer (KMO), total variance explained and eigenvalues.

The descriptive statistics presents the mean and standard deviation of the raw score of each performance indices given by the sample assessors. Correlation matrix shows the strength or magnitude of relationship between two variables and must be greater than 0.3(>0.3). Communalities show the proportion of variance of a variable explained by the common factors.

KMO and Bartlett's sampling adequacy tested if the dataset was suitable for factor analysis. The KMO value must be greater or equal to 0.5 (KMO>=0.5) and Bartlett's test value must be less than or equal to 0.005(p<0.005). The total variance explained determined the number of components to be extracted.

2.1. Data Collection, Analysis and Interpretation of Results

The informed tree based search or heuristic algorithms were implemented in Java programming language and run on Windows 7 64-bits operating system, intel®Pentium CPU 2030M@ 2.5GHZ Central Processing Unit (CPU), 4GB Random Access Memory (RAM) and 750 GB hard disk drive. The performance of the algorithms was tested for each of the experiment by varying the input routes of Romanian road map distance to produce different results for the execution time, memory usage and number of nodes visited.

2.2. Data Generated

The descriptive statistics show the mean and standard deviation of the rating of the impact of the execution time, memory usage and number of nodes visited on the efficiency of best first, A* heuristic and hill climbing algorithms. For instance, mean and standard deviation for best first on time taken are (359388.86, 194888.72) respectively. For A*, the mean and standard deviation for time taken are (707811.71, 221205.18) respectively. The mean and standard deviation of time taken for hill climbing are (1120938.14, 704756.30) respectively.

The extraction method used was factor analysis by principal component and rotation method was Promax with Kaiser Normalization. The analyzed results show that each factors show high correlation in terms of their loading on the heuristic algorithms. For best first search, the correlation between time taken and number of nodes is 0.986. For A*, the correlation between time taken and number of nodes is 0.989. For hill climbing, time taken and number of nodes is 0.752. KMO and Bartlett's sampling adequacy tested if the dataset was suitable for factor analysis. The KMO value must be greater or equal to 0.5 (KMO>=0.5) and Bartlett's test value must be less than or equal to 0.005(p<0.005). Barlett's test of sphericity for best first algorithm produce X^2 of 31.131, degree of freedom of 3 and significance level of 0.000 with KMO of 0.673, which indicated adequacy of sample data.

The communalities of the performance indices generated for the heuristic algorithms with principal component as the extraction method are presented in Tables 1-3, with initial values for the three factors (computational time (nanoseconds), memory usage (bits) and number of nodes visited as 1.000 for best first, A* heuristic and hill climbing algorithms.

Table 1: Communalities for Best First Search

	Initial	Extraction
TIMETAKEN (nanoseconds)	1.000	.982
NUMBER OF NODES	1.000	.996
MEMORY USED (bits)	1.000	.983

Table 2: Communalities for A* Search

	Initial	Extraction
TIMETAKEN (nanoseconds)	1.000	.993
NUMBER OF NODES	1.000	.994
MEMORY USED (bits)	1.000	.996

Table 3: Communalities for Hill climbing

	Initial	Extraction
TIMETAKEN (nanoseconds)	1.000	.862
NUMBER OF NODES	1.000	.872
MEMORY USED (bits)	1.000	.970

The generated component score coefficient matrices are used to estimate the assessment of each assessor of the impact of computational time, memory usage and number of nodes visited on the performance of heuristic algorithms.
This can be achieved by formulating the linear equation of the form;

$$C_{i,j} - \sum_{k=1}^{3} b_{k,j} S_{i,k} \qquad i = 1,2 \dots n, \quad j - 1 \qquad (3)$$

Where $C_{i,j}$ represents the contribution of *ith* assessor to j^{th} factor; $b_{k,j}$ represents the component score coefficient of k^{th} decision variables for j^{th} factor; $S_{i,k}$ represents the standard score of *ith* assessor for k^{th} decision variables and *n* represents the number of sampled assessor
$S_{i,k}$ is estimated by;

$$S_{i,k} = A + \frac{x_I + y_I}{d_i} \qquad (4)$$

Where A represents the allowance minimum raw score for decision variable, x_i represents the raw score of *ith* decision variables; y_i represents the mean of the raw scores of *ith* decision variables; d_i represents the standard deviation of the raw scores of *ith* decision variables. For each sampled assessor, the system of linear equation for the single extracted factor can be represented as follows;

$$\boldsymbol{b_{1,1}S_{i,1} + b_{2,1}S_{i,2} + \dots + b_{4,1}S_{i,4} = C_{i,1}} \qquad (5)$$

In an attempt to evaluate the percentage contribution of each factor to the efficiency of the heuristic algorithms, the eigenvalue of each factor is generated. The eigenvalue of j^{th} factor denoted by $\boldsymbol{E_j}$ is calculated by;

$$E_j = \sum_{k=1}^{3} X_{i,j}^2 \qquad i = 1,2,3; \ j = 1 \qquad (6)$$

Where $X_{i,j}$ represents the loading of j^{th} factor on the of *ith* decision variables.
The eigenvalue is used to indicate how well each of the factors fit the experimental data. The percentage of variance

$$P = 100\left(\frac{E_J}{n}\right) \qquad (7)$$

Where *n* represents the number of decision variables considered in the research. Tables 4-6 presented the eigenvalues, percentage of contribution and cumulative percentage of contribution of the three considered factors for each of the three heuristic algorithms.

Table 4: Total Variance Explained for Best first Search

Component	Total	% of Variance	Cummulative %
Initial Eigenvalues			
1	2.961	98.701	98.701
2	.033	1.106	99.807
3	.006	.193	100.000
Extraction Sums of Squared Loading			
1	2.961	98.701	98.701

Table 5: Total Variance Explained for A* Search

Component	Total	% of Variance	Cummulative %
Initial Eigenvalues			
1	2.984	99.459	99.459
2	.011	.365	99.824
3	.005	.176	100.000
Extraction Sums of Squared Loading			
1	2.984	99.459	99.459

Table 6: Total Variance Explained for Hill climbing

Component	Total	% of Variance	Cummulative %
Initial Eigenvalues			
1	2.704	90.142	90.142
2	.249	8.289	98.430
3	.047	1.570	100.000
Extraction Sums of Squared Loading			
1	2.704	90.142	90.142

The three factors contributed total of 100% to the efficiency of the three heuristic algorithms considered. From the results,' time taken' contributed 98.701%,' memory usage 'contributed 1.106% and number of nodes visited contributed 0.193% impact on the efficiency of best first algorithm.

3. CONCLUSION AND RECOMMENDATIONS

The results obtained from the research showed that computational time was the main factor affecting the efficiency of the informed tree based search algorithm based on the influence. Computation time contributed 98.701, 99.459, 90.142 and 94.102% percentage of variance for best first, A*heuristic, hill climbing and greedy search respectively. The number of nodes visited contributed 1.106, 0.365, 8.289 and 4.717% percentage of variance for best first, A*heuristic, hill climbing and greedy search respectively. Also, memory usage also contributed 0.193, 0.176, 1.570 and 1.182% percentage of variance for best first, A* heuristic, hill climbing and greedy search respectively.
Percentage of variance forms the basis for prioritizing each factor based on their influence on the efficiency of the heuristic algorithm. Therefore, A* heuristic performed better than best first, hill climbing and greedy search algorithm in terms of computational time. The research prioritized computational time as the main factor affecting the efficiency of informed tree based or heuristic algorithms. It is highly recommended that other researchers should work with other factor analysis techniques to see whether there will be deviation from the results obtained in this work.

REFERENCES

[BH10] **B. Bonet, E. A. Hansen** - *Heuristic Search for Planning under Uncertainty*, Chapter in Heuristics, Probability and Casuality, A Tribune to Judea Pearl College Publications, pp.3-22, 2010.

[C+99] **P. P. Chakrabarti, S. Ghose, A. Acharya, S. C. De Sarkar** - *Heuristic Search in Restricted Memory*, Artificial intelligence, Vol.41(2), pp. 197- 221, 1999.

[CGR96] **B. V. Charkassy, A. V. Goldberg, T. Radzik** - *Shortest Path Algorithms, Theory and Experimental Evaluation*, Mathematical Programming, Vol. 73(2), pp. 127-74, 1996.

[KT14] **M. Kuruvilla, M. Tabassum** - *Experimental Comparison of Uninformed and Heuristic Artificial Intelligence for N-puzzle and * Queen Puzzle Solution*, International Journal of Digital Information and Wireless Communication (IJDIWC), The Society of Digital Information and Wireless Communications, Vol. 4(1), pp. 143-154, 2014.

[O+13] **S. O. Olabiyisi, T. S. Aladesae, F. I. Oyeyinka, O. Oladipo** - *Evaluation of Critical Factors Affecting the Efficiency of Searching Techniques*, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 3(7), pp. 34-40, 2013.

[PT04] **G. P. Potdar, R. C. Thool** - *Comparison of various Heuristic Search Techniques for Finding Shortest Path*, International Journal of Artificial Intelligence Applications (IJAIA), Department of Information Technology, Vol. 5(4), pp. 63-74, 2004.

[RN10] **S. Rusell, P. Norvig** - *Artificial Intelligence*, Pearson Education, International Upper Saddle, Vol. 63, River, NJ, 2010.

[Yol01] **P. Yolum** - *Principles of Artificial Intelligence*, Department of Mathematical and Computer Science, University of Southern Denmark, 2001.