GSJ: Volume 10, Issue 1, January 2022, Online: ISSN 2320-9186
www.globalscientificjournal.com

A Framework for Analysing Heuristics Data for Performance Comparison to Solve a Travelling Salesperson Problem

Stanley Murairwa
Africa University

Abstract
The research conducted the designing and execution of experiments and analysis of the heuristic experimental data for performance comparison. The research also evaluated the significance of the normality assumption in heuristic data analysis for performance comparison. The research chose a static way of defining the information in order to collect accurate and reliable data during the experiments. The heuristic algorithms terminated naturally after satisfying certain conditions. A simulated twenty-four Zimbabwe major cities problem (s24z instance) was used to determine the performance of the Simulated Annealing (SA) and its modified variant (MSA), Genetic Algorithm (GA), Ant Colony (AC), Tabu Search (TS) and its modified variant (MTS), Lin Kernighan Heuristic (LKH) and Neural Network Algorithm (NNA). The research established that the normality assumption is irrelevant in statistical heuristic data analysis for performance comparison. A number of statistical tools and procedures were identified and applied to analyse the heuristic data and these include Ryan-Joiner Test, Sample Kolmogorov–Sminorv Test, Box Plot, mean, median, mode, standard deviation, kurtosis, skewness, percentiles (worst and best fits), standard error of the mean and percentage deviations. All the statistical tools and procedures successfully detected the performance of the heuristics for comparison. The modified Tabu Search (MTS) algorithm was found to be the best heuristic among all the compared heuristic algorithms. The research proposed a framework for analysing heuristic data for performance comparison.

Keywords: Statistical Tools and Procedures, Heuristics Performance Comparison, Normality Assumption, Travelling Salesperson Problem, Metaheuristics
JEL Classification: C6, C61

1.0 Introduction

This article deals with the design and execution of heuristic experiments and application of statistical tools and procedures to analyse different heuristic experimental results for performance comparison. Murairwa (2020) stated that the use of the real-life heuristic TSP problems should be preferred instead of the simulated TSP instances so as to tell the full story about the real-world performance of the heuristics. The use of more real life TSP problems creates opportunities for addressing the real life business challenges that most leaders are currently facing in the existing Disruptive Volatile, Uncertain, Complex and Ambiguous (DVUCA) world. The testing of normality assumption in heuristic statistical data analysis particularly for performance comparison is unpopular with all generation researchers. According to Murairwa and Nazri (2010), the standard normal distribution is hard to find in real life and the matrix instances are not population as evidenced by the number of implementation in literature. On the other hand, there is limited use of statistical tools and procedures to analyse heuristic data for performance comparison as supported by Barr *et al*'s (2001) assertion that the area was neglected. The objectives of the research were to evaluate the relevance of the normality assumption and explore the significance of the statistical tools and procedures in analysing heuristics data for comparison to solve a simulated twenty-four Zimbabwe's major cities problem (s24z instance).

1.1 Simulated Twenty-Four Zimbabwe Cities Problem (s24z Instance)

Let the twenty-four Zimbabwe cities be located at A(738; 667), B(860; 222), C(942; 466), D(617; 482), E(880; 566), F(810; 475), G(732; 314), H(725; 418), I(980; 306), J(640; 145), K(714; 174), L(875; 298), M(989; 452), N(879; 171), O(937; 231), P(552; 512), Q(723; 370), R(926; 334), S(833; 268), T(743; 496), U(758; 224), V(971; 379), W(372; 280) and X(644; 104) coordinates (*x; y*). Given a set (c_1, c_2, ………., c_{24}) of the Zimbabwe cities and for each pair (c_i, c_j) of the distinct cities, there is a distance $d(c_i, c_j)$. The matrix distances satisfy the condition that $d(c_i, c_j) = d(c_j, c_i)$ for $1 \leq i, j \leq 24$ as proved by Alba (2005) and Aarts and Lenstra (1997). The goal is to compare the performance of heuristics on determining an ordinary π of the cities that minimises the distance quantity:

$$\sum_{i=1}^{23} d\left(c_{\pi(i)}, c_{\pi(i+1)}\right) + d\left(c_{\pi(24)}, c_{\pi(1)}\right), \quad (1)$$

2.0 Literature Review

The research assumes that $d\left(c_i, c_j\right)$ is the length of the shortest path between city *i* and city *j*, not the length of the shortest physical route that avoids all other cities, triangle inequality and normality assumption holds. Alba (2005) stated that most studies assume normality for data sets of more than 30 or 50 values (an assumption that is formally grounded). Chiarandini, Paquete, Preuss and Ridge (2007) argued that if the distribution of the data is unknown, that does not rule out the use of parametric statistics. Chiarandini, Paquete, Preuss and Ridge (2007) went on further to recommend the use of data transformation techniques (such as logarithm, inverse and square root) to make the data to meet the normality assumption.

The Binary Cat Swarm Optimization Algorithm (Villa & Castillo, 2020) had the lowest error in all the comparison experiments conducted. Chiarandini, Paquete, Preuss and Ridge (2007) stated that more advanced topics of statistics have been used to analyse and compare metaheuristics such as regression trees, Design of Experiments (DOE) and sequential testing through fine-tuning algorithms. According to Sze and Tiong (2007), the researchers can study the performance of the heuristics by increasing the number of the cities while all other conditions are unchanged. This approach measures the capability of each of the heuristic to handle large combinatorial optimisation problem. According to Murairwa (2020), the statistical comparison of the performance of the heuristics can be done at four levels, namely, performance differences, quality of feasible solutions, time utilisation statistics and performance reliability. Murairwa (2020) successfully implemented statistical experimental design to understand and assess heuristic performances and found the metaheuristics to be dominant among all the heuristics compared. Under the multivariate model, Chiarandini, Paquete, Preuss and Ridge (2007) and Johnson (2002) preferred to compare the solution distance and CPU-time of the heuristics.

3.0 Methodology

The research conducted experiments for selected heuristics to solve the simulated twenty-four Zimbabwe cities problem (s24z instance) that was obtained from Zimbabwe National Statistical Agency (ZimStat). There are several possible ways of defining heuristic information or parameters and this was a key component since the research was comparing the performance of different heuristics using a single TSP instance. Therefore, in order not to compromise the performance of the heuristics, the research selected a static way of defining the information. To collect adequate data for the comparative analysis, the research sets up experiments and performed 100 independent runs of each of the selected heuristics. The experimental data recording table (EDRT) (Murairwa, 2020; 2010) was used to collect the heuristics' longitudinal repeated performance data. According to Alba (2005), 30 independent runs are usually regarded as a minimum number of runs in heuristic experiments. The best fit and its CPU time at termination of each run was recorded. Therefore, the multivariate methodology was selected as the appropriate technique to analyse the heuristic data. The performance of the heuristics was evaluated at three levels; the worst case, probabilistic (average case) and empirical analysis. The research applied Chiarandini, Paquete, Preuss and Ridge's (2007) two specific scenarios under the multivariate model, namely, the study of solution distance and CPU-time when a certain termination criterion is reached, that is, the heuristics terminated naturally when certain termination conditions were met.

3.1 Heuristics parameter tuning

The study selected the best combination of parameters of each heuristic for the experiment. Dorigo, *et al*. (2004) acknowledged after carrying out an experiment that the combination of the parameters of the heuristic influences the performance of the heuristic. However, the availability of better heuristics parameter combinations than those used in this research could not be ruled out since the field is broad with many scholars and/or researchers continuing to discover new heuristic parameter combinations. The three major combinatorial optimisation categories used are constructive and improving heuristics, meta-heuristics/composite and hybrid meta-heuristics. The heuristics parameter combinations used during the experiments are presented henceforth.

- The *Simulated Annealing (SA)* used the 2-Opt method (Croes, 1958), 0.999 temperature reduction factor; starting temperature of 20 and 10 good swaps (inter loop break) at each temperature. The modified Simulated Annealing (MSA) was also applied to solve the s24z instance. Each run terminated after 1000 iterations without change to the current best fit.
- The *Genetic Algorithm (GA)* population size was fixed at 24 with 0.30 and 0.70 as the probability of mutation and recombination respectively and the methods used were inversion for the mutation and partially matched for recombination. The GA terminated after 1000 generations without change to the current best fit.
- The *Ant Colony (AC)* was configured to run for a maximum of 5000 iterations with 1 and 0.5 pheromone trail (additive constant) and persistence respectively. Pheromone visibility sensitivity (number of ants) was configured at 5. These parameters were discovered to be the best (Dorigo, Maniezzo, & Colorni, 1996). Each run of AC terminated after 1000 iterations without change to the current best fit.

- The *Tabu Search (TS)* uses dynamic neighbourhood search strategy to move from a solution i to a solution j in the neighbourhood of i $\{N^*(i)\}$ until the stopping criterion is realised (Glover, 1990a; 1989). A Modified Tabu Search (MTS) was also applied to solve the s24z instance. Each run of the TS terminated after 1000 iterations without change to the current best fit.
- The *Nearest Neighbour Algorithm (NNA)* constructs an ordering $c\pi(1)$, $c\pi(2)$, . ., $c\pi(N)$ of the cities, with the initial city $c\pi(1)$ selected arbitrarily and in general $c\pi(i+1)$ selected to be the city ck that minimises $\{d(c\pi(i), ck)$: $k \neq \pi(j), 1 \leq j \leq i\}$. The corresponding tour traverses the cities in the constructed order, returning to $c\pi(1)$ after visiting city $c\pi(N)$ searching when all nodes are on the tour (Hahsler & Hornik, 2009; 2007) or after returning to city ($c\pi(1)$) after visiting city ($c\pi(N)$) for TSP (Dorigo & Stutzle, 2004; Johnson & McGeoch, 2002; Johnson, et al., 2002).
- The *Local Search (LS)* applies the edge exchange technique and that is generally known as the λ-Opt procedures, where λ is the number of edges to be exchanged in each of the iterations by another λ edges (Hwang, Alidaee, & Johnson, 1999). The research applied the 2-Opt that was configured to terminate after considering all the $\frac{n(n-1)}{2}$ pairs of edges.
- The *Lin Kernighan Heuristic (LKH)* starts from an arbitrary bisection and swaps pairs of nodes in order to improve the cost of the partition (Vahid & Le, 1997; Lin & Kernighan, 1973). The LKH was configured to terminate when all the nodes have been considered without change to the current solution.

3.2 Configuration

The heuristic algorithms were implemented in java programming language and MATLAB R2004a and the experiments were performed on the HP ProBook 450 G7 – Intel(R) Core(TM) i5-10210U CPU @ 1.60GHz 2.11 GHz processor running on a 64 bit Windows 10 Pro with 8 GB of RAM.

3.3 Statistical Data Analysis Tools

The research tested the normality assumption through plotting the network diagram of the distance matrix, performing Ryan-Joiner normality test, Kolmogorv-Sminorv test and box plots and fitting skewness to the data. The percentage deviation (PD) of the best fit (BF) of each heuristic against the global optimum solution (GOS) was calculated with

$$PD = \left(\frac{BF - GOS}{GOS}\right) \times 100, \tag{2}$$

3.4 The Benchmarks

The shortest round route starting from any city obtained in this study is 4478 km. This is used as the current known result of the case study to determine the hit rate of each heuristic. According to Arts and Lenstra (1997), the best construction heuristic, 3-Opt heuristic and “variable-opt” algorithm of Lin and Kernighan (1973) typically get within roughly 10-15%, 3-4% and 1-2% respectively of the global optimum solution in relatively little time. The concept was used to analyse the heuristic data collected for this research. This research used the classes 0-1%, 5-9% and >15% to compare the performance of the heuristics.

3.5 Hypothesis

The research applied Hotelling T-test (Hotelling, 1931) as advocated for by Chiarandini, Paquete, Preuss and Ridge (2007) for bivariate data analysis. The hypothesis used is $H_0: \mu_1 - \mu_2 = 0 \; vs \; H_1: \mu_1 - \mu_2 \neq 0$. The test statistic was computed with

$$F = \frac{n_1 + n_2 - p - 1}{p(n_1 + n_2 - 2)} T^2 \sim F_{p, n_1+n_2-p-1}, \tag{3}$$

where p is the number of parameters, n_1 and n_2 are sample sizes and $n_1 + n_2 - 2$ is the degrees of freedom. The Hotelling T-test was applied to determine a better performing heuristic. The null hypothesis is rejected if the test statistic is greater than the tabulated critical value. If the p-value is greater than the level of significance (α), the difference between the two means is not statistically significant. The difference of two independent means test was used to determine whether two heuristics performed statistically the same or not. The hypothesis used is $H_0: \mu_1 - \mu_2 = 0 \; vs \; H_1: \mu_1 - \mu_2 \neq 0$ at 5% level of significance. The test statistic was computed with

$$Z_{calc} = \frac{(\bar{x}_1 - \bar{x}_2) - (\mu_1 - \mu_2)}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}, \tag{4}$$

The null hypothesis (H_0) is rejected if the absolute test statistic (Z_{calc}) is greater than the absolute tabulated value (Z_{tab}): $|Z_{calc}| > \left|Z_{tab\left(\frac{\alpha}{2},\infty\right)}\right| = Z_{tab\,(0.025,\infty)} = 1.96$.

4.0 Application of statistical tools to analyse heuristic data for comparison

The network diagram of the s24z instance is presented in Figure 1.

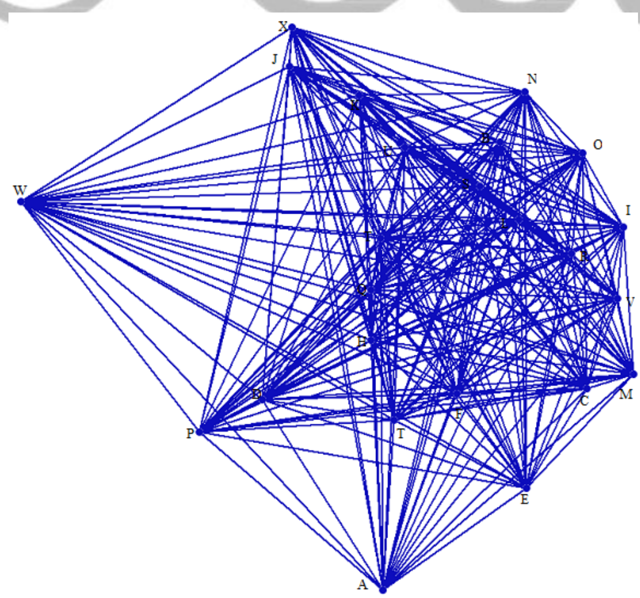

Figure 1: Network diagram of all routes among the cities

The alphabetic letters in Figure 1 represent the cities selected for this research. The graph shows that the distance matrix is symmetric and there are connecting routes from each city to the other twenty-three cities. The research tested for normality of the performance by the heuristics to solve the s24z instance and presented the results in Table 1.

Table 1: Test for normality (Ryan-Joiner Test similar to Shapiro – Wilk Test)

Heuristic	R	p-value	Significant level and Decision		
			0.01	0.05	0.10
LKH	0.9455	<0.0100	*	**	**
LS	0.8379	<0.0100	*	**	**
NNA	0.9875	0.0506	*	*	*
AC	0.9193	<0.0100	*	**	**
GA	0.9907	>0.1000	*	*	*
SA	0.9726	<0.0100	*	**	**
TS	0.9297	<0.0100	*	**	**
MSA	0.9465	<0.0100	*	**	**
MTS	0.9678	<0.0100	*	**	**

* = Accept normality and ** Reject normality. Minitab was used to produce the results

The results in Table 1 shows that the decision on normality depends on the selected level of significance. For one tail test, at 1% level of significance, the normality assumption is not rejected for all the heuristic data. However, the GA and NNA data are normal at all the three levels of significance. The other heuristics performance data are not from a normal distribution at both 5% and 10% levels of significance. According to Alba (2005), at 1% level of significance, the researchers could use parametric test while at both 5% and 10% levels of significance, the researchers could apply non-parametric tests. Alba (2005) suggested that Kolmogorov-Sminorv test could be used to substantiate the results obtained. This test is able to identify more general differences than location differences (mean or median) (Chiarandini, Paquete, Preuss, & Ridge, 2007). The mean is sensitive to extreme data values and thus, Eftimov and Korošec (2019) recommended the use of the median. The results of the Kolmogorov-Sminorv test are presented in Table 2.

Table 2: One – Sample Kolmogorov–Sminorv Test for normality

Heuristics		LS	NNA	LKH	SA	GA	AC	TS	MSA	MTS
N		100	100	100	100	100	100	100	100	100
Normal Parameters(a,b)	Mean	5494.07	5524.05	5385.52	5304.60	5409.24	5405.06	5361.99	4666.51	4843.18
	Std. Dev	113.921	208.943	141.397	211.060	240.202	123.688	178.388	134.720	288.278
Most Extreme Differences	Absolute	0.412	0.107	0.313	0.218	0.084	0.388	0.259	0.176	0.160
	Positive	0.412	0.076	0.218	0.218	0.058	0.263	0.259	0.157	0.160
	Negative	-0.250	-0.107	-0.313	-0.180	-0.084	-0.388	-0.223	-0.176	-0.103
Kolmogorov-Smirnov Z		4.122	1.065	3.127	2.181	0.835	3.881	2.588	1.758	1.596
Asymp. Sig. (2-tailed)		0.000	0.207	0.000	0.000	0.488	0.000	0.000	0.004	0.012

a Test distribution is Normal; b Calculated from data.

Table 2 shows that the normal distribution is rejected for all the performance data of the heuristics. Therefore, the Ryan-Joiner test (similar to Shapiro – Wilk test) and the Kolmogorov–Sminorv test fail to agree on the concept of the distribution of the heuristic data. The box plots were used to further test the performance distribution of the heuristics. The box plot results are presented in Figure 2.

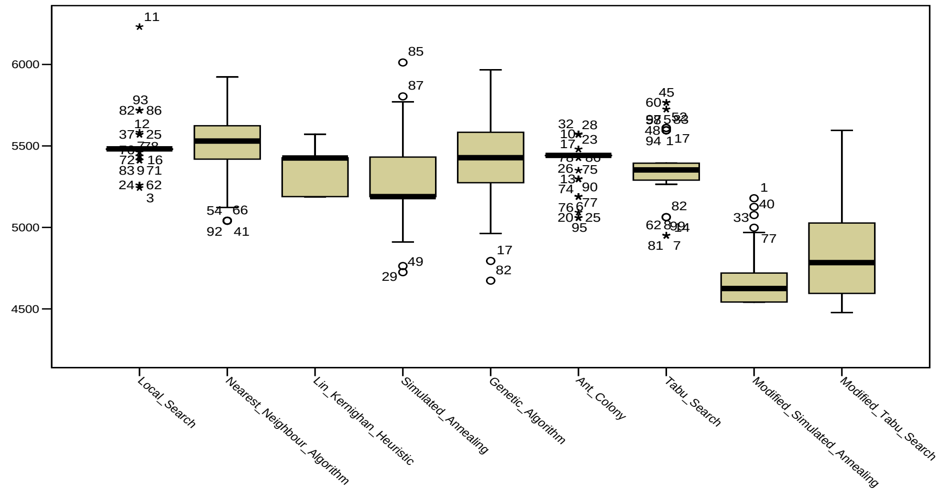

Figure 2: Box plots of the performance of the heuristics

Figure 2 should be interpreted in conjunction with Table 3 below which shows the skewness of the performance of the heuristics. The NNA, LKH, GA and AC data are negatively skewed while the LS, SA, TS, MSA and MTS are positively skewed. Therefore, the results confirm that the normality assumption is not necessary when one is analysing heuristic data for performance comparison. This is a contradiction to Alba (2005) and Chiarandini, Paquete, Preuss and Ridge (2007).

Table 3: Descriptive measures of the performance of the heuristics

Heuristic	Measure							
	Range	Mean		Std. Deviation	Skewness		Kurtosis	
	Statistic	Statistic	Std. Error	Statistic	Statistic	Std. Error	Statistic	Std. Error
LS	988	5494.07	11.39214	113.9214	3.007021	0.24138	17.85678	0.478331
NNA	882	5524.05	20.89435	208.9435	-0.357710	0.24138	0.156913	0.478331
LKH	384	5385.52	14.13967	141.3967	-0.330800	0.24138	-1.188150	0.478331
SA	1287	5304.60	21.10597	211.0597	0.502646	0.24138	1.195955	0.478331
GA	1294	5409.24	24.02022	240.2022	-0.381520	0.24138	0.261987	0.478331
AC	513	5405.06	12.36879	123.6879	-1.574030	0.24138	2.201029	0.478331
TS	816	5361.99	17.83879	178.3879	0.405985	0.24138	1.149054	0.478331
MSA	638	4666.51	13.47201	134.7201	1.564239	0.24138	2.843980	0.478331
MTS	1118	4843.18	28.82775	288.2775	0.882725	0.24138	0.220890	0.478331

Table 3 results show that the NNA, LKH, GA and AC data are negatively skewed while the LS, SA, TS, MSA and MTS are positively skewed. The NNA is commonly known as greedy algorithm and only performs well at the beginning of the construction of the travelling path (Sze & Tiong, 2007). Thus, the solution produced by NNA is not necessarily an optimal solution (Sze & Tiong, 2007). This is the reason for NNA having the largest mean of the best fits. The LS has the smallest standard deviation, followed by the LKH and on the third position is the MSA. However, the best heuristics are the MSA and MTS which performed negatively skewed. This is in support of the idea that the data from a good and reliable heuristic should not be normally distributed. In fact, the normality assumption is not necessary when analysing heuristic data for performance comparison. A comparative analysis of the worst and best scenarios of each heuristic's performance is presented in Table 4.

Table 4: Each heuristic's worst and best of the best optimum fits

	Runs	Worst Best Fit	Best Best Fit	Worst % Deviation	Best % Deviation
LS	100	6231	5243	39.15	17.08
NNA	100	5923	5041	32.27	12.57
LKH	100	5572	5188	24.43	15.86
SA	100	6012	4725	34.26	5.52
GA	100	5967	4673	33.25	4.35
AC	100	5572	5059	24.43	12.97
TS	100	5766	4950	28.76	10.54
MSA	100	5179	4541	15.65	1.41
MTS	100	5596	4478	24.97	0.00

Table 4 shows that the best heuristic is the MTS which managed to find the global optimum solution. The MSA missed the global optimum by 1.41%. The SA and GA missed the target by 5.52% and 4.52% respectively. The LS missed the target by 39.15% and this is the overall worst of the worst scenarios of this research. The least of the worst scenarios of 15.65% was by the MSA. However, the table is not showing the number of attempts which fall in each of the categories in order to further discuss the quality of performance of the heuristics. Table 4 results are graphically presented in Figure 3.

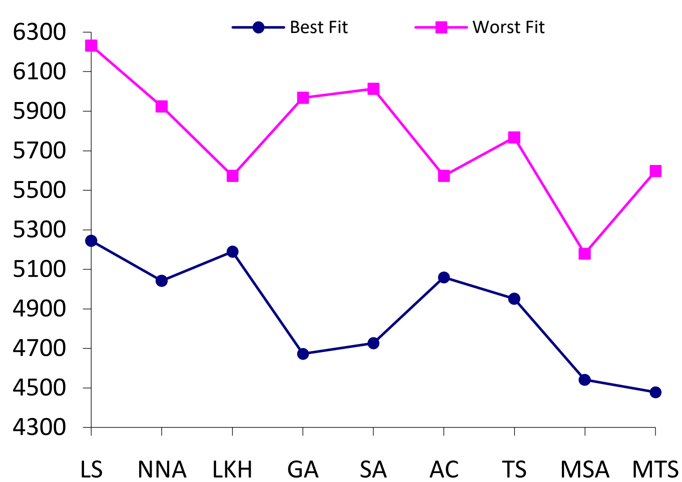

Figure 3: Comparative analysis of the best and worst optimum fits of each heuristic

Figure 3 shows that the MTS managed to find the global optimum solution of the s24z instance. The MSA missed the global optimum solution by a very small margin as shown in Figure 3. Therefore, the MTS was the best heuristic among all the heuristics compared in this research. However, the MSA cannot immediately be ruled out because in Figure 3 it is not clear how many times each of the two heuristics managed to return its best optimum fit for the hundred runs of the experiments. Presented in Figure 4 are the mean standard errors of the performance of the heuristics.

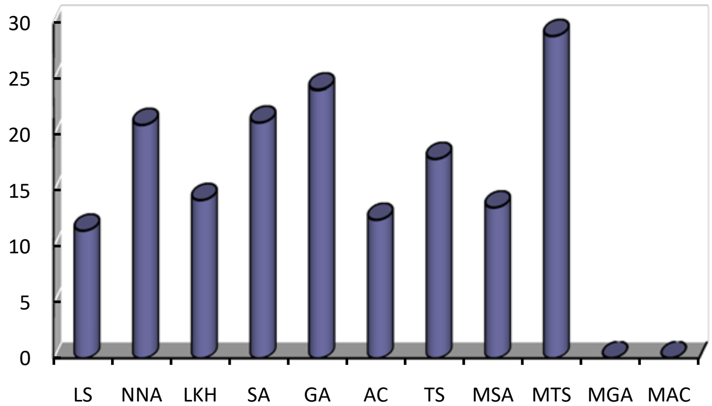

Figure 4: Comparative analysis of the mean standard error

The LS and the MTS have the smallest and largest mean standard errors respectively as shown in Figure 4. The measure is not reliable since the MTS and the MSA were expected to have the smaller mean standard errors. It cannot therefore be concluded that the LS was the best heuristic among those compared. Aarts and Lenstra (1997) concluded that the analysis of the success rate (hit rate) could help to make a meaningful conclusion in such a scenario. The results of the success rate of the heuristics are presented in Table 5.

Table 5: Best fit- hit rate measured against each heuristic's best optimum fit

Heuristic	% hit rate per 100 trials	Average CPU time (in seconds)
LS	1.00	1.0127
NNA	4.00	1.2152
LKH	22.00	1.1796
SA	1.00	11.3927
GA	1.00	379.4716
AC	6.00	2.9157
TS	5.00	1.6772
MSA	23.00	277.0386
MTS	7.00	3.0258

Most of the heuristics (66.67%) successfully obtained their best optimum fit more than once as shown in Table 5. The MSA and LKH succeeded in 23.00% and 22.00% respectively of the total runs to return their best optimum fits. Statistically, the two heuristics performed the same. The LS, SA and GA failed to get their respective best fits for the 100 runs of the experiments. It is therefore tempting to conclude that the MSA and LKH are the best heuristics in terms of persistence search for the global optimum solution since they have the best hit rates. However, Table 4 shows that the best fits by the LKH and MSA are 15.86% and 1.41% respectively away from the global optimum solution. Therefore, the MSA is better than the LKH and it can be concluded that the MSA is the best heuristic. The percentage deviation of the best fit of each heuristic against the global optimum solution computed with Equation 2 is presented in Table 6.

Table 6: Success rate measured against the global optimum solution (4478 km)

Heuristic	% hit	% Deviation of best fit	Number of attempts per 100 trials	Probability of attempts per 100 trials
LS	0	17.08	1	0.01
NNA	0	12.57	4	0.04
LKH	0	15.86	22	0.22
SA	0	5.52	1	0.01
GA	0	4.35	1	0.01
AC	0	12.97	6	0.06
TS	0	10.54	5	0.05
MSA	0	1.41	23	0.23
MTS	7	0.00	7	0.07

Table 6 shows that the MTS outperforms the other heuristics because it successfully obtained the global optimum solution in seven trials. The MSA missed the target by a distance of 1.41% which is not bad. In this case, the MTS can be recommended as the best heuristic among all compared heuristics but the seven success attempts do not truly represent its actual performance over hundred runs. The overall low number of successful attempts indicates that the heuristic does not have the memory of its past run performance. The aspect could assist the heuristic to start searching from its previous run's best fit towards the global optimum solution (improve its past run's best optimum fit). The aspect could improve the hit rate of the heuristic. The comparative analysis of the heuristics over 100 runs is presented in Figure 5.

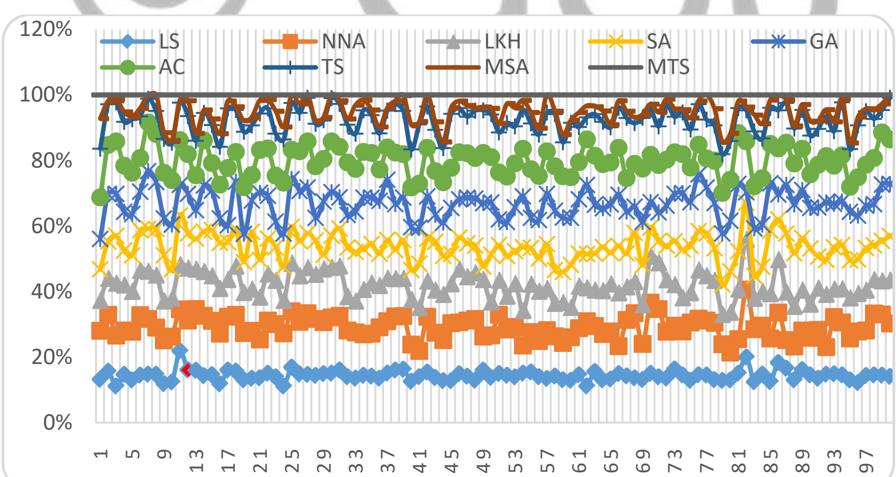

Figure 5: Comparative analysis of the performance of the heuristics over 100 runs

A number of interesting points relating to the performance of the heuristics are portrayed in Figure 5. Firstly, the graph shows a random search of the global optimum solution by the heuristics. This really confirms the idea made earlier that the heuristics are lacking the memory of their past run performance. Secondly, from Figure 5, the MTS outperforms the other compared heuristics. The LS has the worst point (run 12) recorded during the experiments. However, from Figure 5, it is not possible to separate the performance of the MSA and MTS. In order to try and separate the two heuristics' performance, the analysis of the CPU time is presented in Table 7.

Table 7: Comparative analysis of the CPU time of the heuristics

Heuristic	Runs	Mean	Median	TrMean	StDev	SEMean
LS	100	1.0127	0.9895	1.0113	0.2363	0.0236
NNA	100	1.2152	1.2168	1.2137	0.0837	0.0084
LKH	100	1.1796	1.1300	1.1570	0.2115	0.0211
SA	100	11.393	11.730	11.3610	1.9370	0.1940
GA	100	379.50	371.70	373.30	103.70	10.400
AC	100	2.9157	2.7800	2.8626	0.5020	0.0502
TS	100	1.6772	1.7160	1.6857	0.1849	0.0185
MSA	100	277.04	283.24	277.65	37.730	3.7700
MTS	100	3.0258	3.0420	3.0283	0.4259	0.0426

Table 7 shows the time spent by the CPU searching for the global optimum solution. The available information is in favour of the NNA which has the smallest standard deviation and the variance. There is a small difference of time utilization between the AC and MST, with the later taking a small edge over the former. The GA and MSA took an average of 379,50 and 277.04 seconds respectively to terminate with the best optimum fits. The research applied the difference of two independent means test in order to determine whether the two heuristics performed the same or not. The results obtained are Mean difference (102,46), Standard error (11.04), 95% CI (80.70 – 124.22), Z-statistic (9.29) (using Equation 4) and Significance level (P < 0.0001). Since $|Z_{calc}| = 9.29 > |Z_{tab\,(0.025,\infty)}| = 1.96$, the null hypothesis is rejected. The two heuristics performed differently. Therefore, the GA took the longest average time to terminate with the best optimum fit.

The GA works by generating a population of chromosomes, each representing a feasible solution to the problem. New chromosomes are created by crossover and mutation. Chromosomes are then evaluated according to the fitness function with the fittest surviving and less fit being eliminated. The result is a gene pool that evolves over time to produce better and better solutions to a problem. According to Sze and Tiong (2007), the key to finding a good solution using a GA lies in developing a good chromosome representation of solutions to the problem. This explains why the GA has the longest CPU time statistics in Table 7. Table 8 presents the quality performance of the heuristics using the benchmarks of Aarts and Lenstra (1997).

Table 8: Heuristics quality performance: cumulative number of runs in each percentage deviation range

Heuristic	0-1%	0-2%	0-4%	0-9%	0-15%	>15%	
						0-20%	>20%
LS	-	-	-	-	-	4	100
NNA	-	-	-	-	9	21	100
LKH	-	-	-	-	-	30	100
SA	-	-	-	2	10	68	100
GA	-	-	-	2	14	42	100
AC	-	-	-	-	7	20	100
TS	-	-	-	-	7	56	100
MSA	-	30	54	92	99	100	100

MTS	11	15	30	64	84	91	100

The best heuristic according to Aarts and lenstra's (1997) quality measurement classes is the MTS. However, the MSA cannot be ruled out just because it failed to find the global optimum solution. The heuristic has 30% of the 100 runs within 2% of the global optimum solution which is twice the number of success to that of the MTS. The MTS has 16% of the runs that terminated outside the range 10-15% to 1% of the MSA. A clear picture of the performance of the heuristics according to Aarts and Lenstra (1997) is presented in Figure 6.

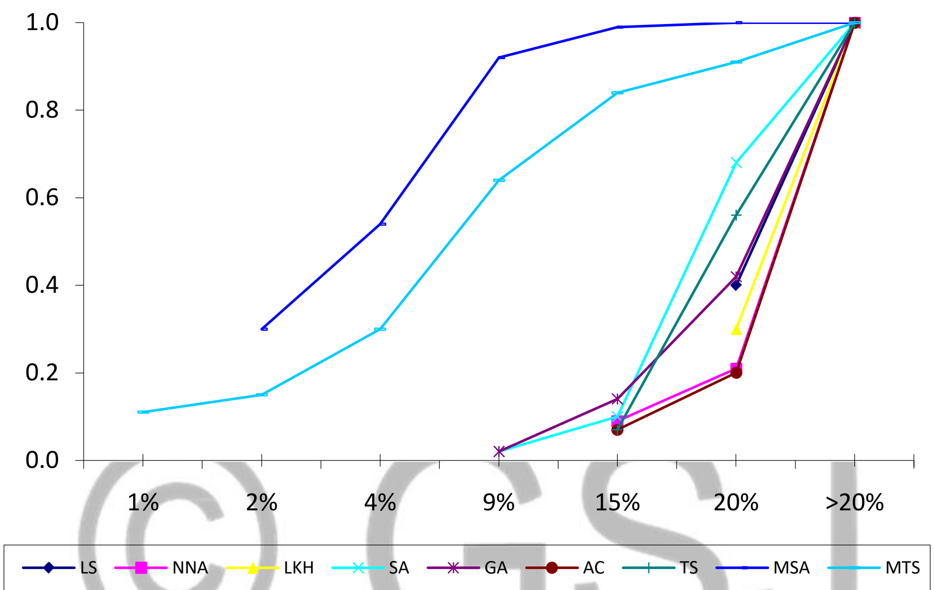

Figure 6: A Less than cumulative probabilities of the performance of the heuristics: the global optimum is in the range 0-1%

The MTS managed to find the global optimum solution with a probability of 0.11. The LS and LKH heuristics failed to find the best optimum fits within the 0-15% range. The performance by the LKH in this research is in contrast to the findings of Lin and Kernighan (1973) that are quoted by Aarts and Lenstra (1997). Richter, Goldengorin, Jäger and Molitor (2007) concluded that the LKH performed better than most of the developed heuristic algorithms. The researchers found that the heuristic's performance lies within the 1-2% region of the global optimum solution. It should be noted that the parameter tuning, coding and programming levels of the algorithms have a bearing on the performance of the heuristics. The comparative analysis of the performance mode of the heuristics is presented in Table 9.

Table 9: Comparative Analysis of the Mode of the Heuristics Data

	LS	NNA	LKH	SA	GA	AC	TS	MSA	MTS
N	100	100	100	100	100	100	100	100	100
Mode	5482	5573	5426	5189	5296(a)	5442	5381	4541	4478(a)
% Dev	22.42	24.45	21.17	15.88	18.27	21.53	20.17	1.41	0.00
Frequency	62	7	31	35	3	63	17	23	7

(a) Multiple modes exist. The smallest value is shown

Table 9 shows that the performance of the LS and AC was almost the same. Their modes' deviations from the global optimum solution are 22.42% and 21.53% respectively. The two heuristics performed badly since more than 60% of their total runs terminated outside Alba's (2005) range (1-15%) of the best heuristic. That leaves a toss for the best heuristic between the MSA and MTS. The MTS has three modes at points 0.00%, 2.61% and 5.14% from the global optimum solution. Hotelling T-test (Hotelling, 1931) and MANOVA are recommended by Chiarandini, Paquete, Preuss and Ridge (2007) as the best for bivariate and multivariate data analysis respectively. In this research, the Hotelling T-test was applied to analyse the MSA and MTS and the results are presented in Table 10.

Table 10: Hotelling's T-Squared Test: bivariate test of the null hypothesis that both heuristics have the same mean

Hotelling's T-Squared	F	df1	df2	Sig
32.190	32.190	1	99	0.000

The covariance matrix is calculated and used in the analysis.

Table 10 shows that $\alpha = 0.5 > p - value =$0.000, therefore, the difference between the means of the two heuristics is statistically significance. Therefore, the means of the MSA and MTS are significantly different. Thus, the two heuristics performed differently. The reliability analysis results are presented in Table 11 to check which heuristic has a better reliability statistic than the other.

Table 11: Reliability of the MSA and MTS to maintain the same performance under the same conditions

Heuristic	Cronbach's Alpha	Cronbach's Alpha Based on Standardized Items	N of Items
MSA	0.022026	0.022721	2
MTS	0.174589	0.195422	2

The Cronbach's alpha statistics that are presented for these heuristics are estimates of the true alpha statistics, which in turn are the lower bound statistics for the true reliability. Therefore, the MTS had a better lower bound for the true reliability than the MSA. Therefore, the MTS had a better reliability than the MSA. However, without any upper bound statistics, it is difficult to make a meaningful conclusion basing only on the difference between the lower bound statistics. Thus, under the scenario presented in Table 11, one of the two heuristics can have a better reliability than the other.

5.0 Heuristic Data Analysis Framework for performance comparison

The research proposed and examined a framework that can be implemented to analyse heuristic data for performance comparison. The research also identified appropriate statistical tools and procedures that can be applied to analyse heuristic data for performance comparison. The framework that can be applied to analyse heuristic data for performance comparison is presented in Figure 7.

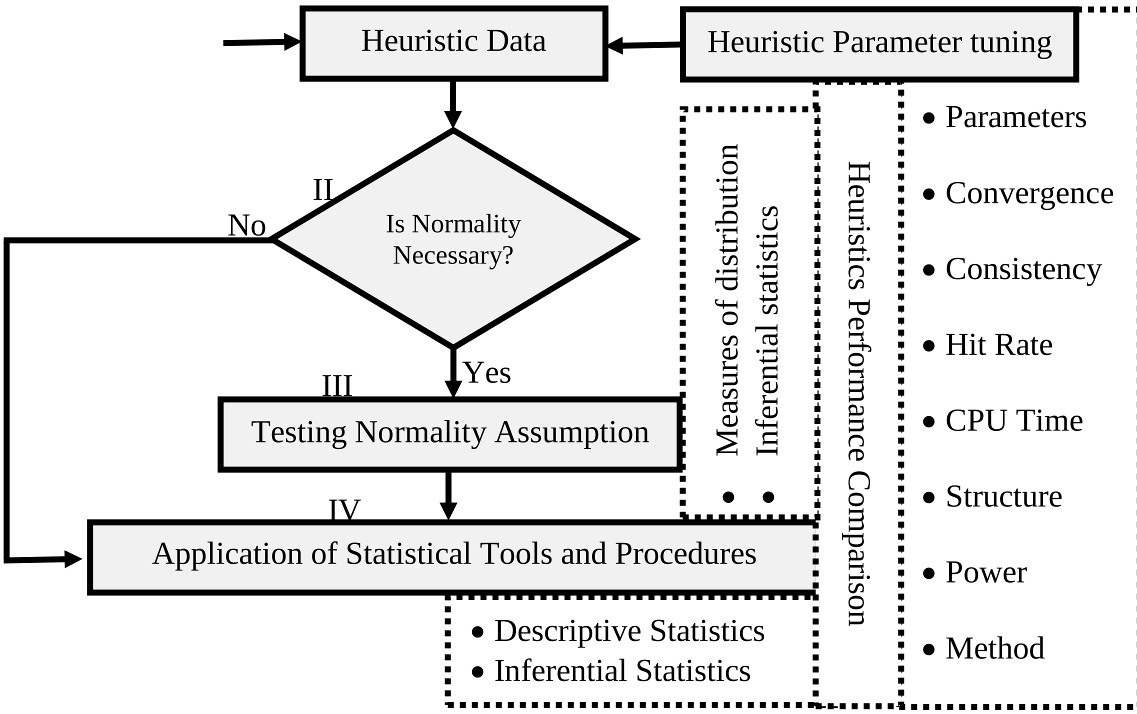

Figure 7: An Appropriate Heuristic Data Analysis Framework

Figure 7 shows that the comparison of the performance of the heuristics should begin at the selection of the algorithms and parameters to use. This is the initial stage of comparing the performance of heuristics. The stage assesses the level of coding and programming, search method and structure of the selected heuristic algorithms. Thus, the initial stage determines whether the comparison of the selected heuristic algorithms should proceed or not. If the initial stage is skipped, it may be a mammoth task to successfully compare the performance of the heuristic algorithms since the stage anchors the whole process. The initial stage (I) determines whether the heuristic algorithms are comparable. In second stage (II), the researcher makes a decision on whether the normality assumption is relevant or not. If not, the researcher implements the fourth stage; otherwise the researcher implements the third stage. In third stage (III), the measures of distribution and inferential statistics are applied to test the normality assumption of the heuristic data. The purpose of this stage is to determine whether parametric or non-parametric tests should be used to analyse the heuristic data. The results of the normality test can also be used in the fourth stage to compare the performance of the heuristics. In fourth stage (IV), the heuristic data for performance comparison are analysed using descriptive and inferential statistics as shown in Figure 7.

The heuristics can be compared on the parameter tuning (restrictions, conditions, instructions), convergence rate (termination speed), consistency (reliability), hit rate, CPU time (search period), structure (use of memory), power (capability, search effort, escape effort) and method (procedure, rule, modus operandi, approach, formula, plan) applied. Eftimov and Korošec (2019) stated that the search distribution information of the heuristic can be used to discover the exploitation and exploration powers of the compared heuristic algorithms.

6.0 Discussion of the findings

The assumption of normality is not necessary when analyzing heuristic data for performance comparison. However, Alba (2005) stated that most researchers assume normality of the data sets of more than 30 or 50 values (an assumption that is formally grounded). The research approved beyond reasonable doubt that the normality assumption is unnecessary when analysing heuristic data for performance comparison because it will be rejected after testing. Of all the nine heuristics analysed, normality assumption was rejected in favour of the skewed distribution. Therefore, there is need to determine the distribution of the heuristic data instead of assuming normality. If one is to make an assumption when minimizing (maximizing) a combinatorial optimisation problem, the positively (negatively) skewed distribution assumption sounds more practical than the normality assumption.

The results obtained clearly indicate that MSA and MTS performed exceptionally well for all the benchmarks considered due to their capability to explore and exploit the solution space effectively. The two heuristics were inseparable throughout the analysis. However, the MTS managed to find the global optimum solution in 7% of all the experimental runs. Another research could assist to determine a better heuristic between the two heuristics.

The research discovered that the heuristics considered could perform better if they could use information of their past run performance such as the last optimum fit. This component could assist the heuristics to improve their hit rates and previous optimum fits. The idea needs further investigation to determine whether it is feasible to include such a function without compromising the current performance of the heuristic algorithms. It is therefore recommended that the MSA be further modified in the next phase to improve its 23% hit rate level.

Finally, the results are indicative of what the selected parameters are capable of achieving and not what the best parameters combination available, if any, can perform. The research does not regard this research as having regarded the best parameter combinations of the heuristics investigated despite the effort applied to try and achieve the goal but as an initial point to further investigate the best performing parameters of the outstanding heuristics.

References

Aarts, E., & Lenstra, J. K. (1997). *Local Search in Combinatorial Optimization.* Chichester: John Willey & Sons.

Alba, E. (2005). *Parallel Metaheuristics: A New Class of Algorithms.* New York: John Wiley & Sons.

Barr, R. S., Golden, B. L., Kelly, J. P., Resende, M. G., & Stewart, W. R. (2001). *Guidelines for Designing and Reporting on Computational Experiments with Heuristic Methods.* South Methodist University, Department of Computer Science and Engineering, Dallas.

Chiarandini, M., Paquete, L., Preuss, M., & Ridge, E. (2007). *Experiments on Metaheuristics: Methodological Overview and Open Issues.*

Croes, G. A. (1958). A Method for Solving TSPs. *Operations Research, 6*, 791 - 812.

Dorigo, M., & Stutzle, T. (2004). *Ant Colony Optimisation.* Bradford: MIT Press.

Dorigo, M., Birattari, M., Blum, C., Gambardella, L. M., Mondada, F., & Stutzle, T. (2004). Ant Colony Optimisation and Swarm Intelligence. *4th International Workshop, ANTS 2004.* Brussels, Belgium.

Dorigo, M., Maniezzo, V., & Colorni, A. (1996). Ant System: Optimisation by a colony of cooperating agents. *IEEE Transactions on Systems, Man, and Cybernetics-Part B, 26*(1), 29 - 41. doi:10.1109/3477.484436

Eftimov, T., & Korošec, P. (2019). A novel statistical approach for comparing meta-heuristic stochastic optimization algorithms according to the distribution of solutions in the search space. *Information Sciences, 489*, 255 – 273. doi:10.1016/j.ins.2019.03.049

Glover, F. (1989). Tabu Search-Part I. *ORSA: Journal on computing, 1*(3), 190 - 206.

Glover, F. (1990a). Tabu Search-Part II. *ORSA: Journal on computing, 2*(1), 4 - 32.

Golden, B., Bodin, L., Doyle, T., & Stewart, W. J. (1980). Approximate Travelling Salesman Algorithms. *Operations Research, 28*(3), 694 - 711.

Hahsler, M., & Hornik, K. (2007). TSP - Infrastructure for the TSP. *Journal of Statistical Software, 23*(2), 1 - 21.

Hahsler, M., & Hornik, K. (2009). *Introduction to TSP – Infrastructure for the TSP.* Working paper.

Hotelling, H. (1931). The generalization of Student’s ratio . *Annals of Mathematical Statistics, 2*(3), 360 – 378.

Hwang, C.–P., Alidaee, B., & Johnson, J. D. (1999). A Tour Construction Heuristic for the TSP. *Journal of the Operational Research Society, 50*(8), 797 - 809. doi:10.1057/palgrave.jors.2600761

Johnson, D. S. (2002). A Theoretician's Guide to the Experimental Analysis of Algorithms. In M. H. Goldwasser, D. S. Johnson, & C. C. McGeoch (Eds.), *Data Structures, Near Neighbour Searches, and Methodology: Fifth and Sixth DIMACS Implementation Challenges* (pp. 215 - 250). American Mathematical Society.

Johnson, D. S., & McGeoch, L. A. (2002). Experimental Analysis of Heuristics for the STSP. In Gutin, & Punnen (Eds.), *The TSP and its Variations.* Boston: Kluwer Academic Publishers.

Johnson, D. S., Gutin, G., McGeoch, L. A., Yeo, A., Zhang, W., & Zverovitch, A. (2002). Experimental Analysis of Heuristics for the ASTSP. In Gutin, & Punnen (Eds.), *The TSP and its Variations.* Boston: Kluwer Academic Publishers.

Lin, S., & Kernighan, B. W. (1973). An Effective Heuristic Algorithm for the TSP. *Operations Research, 21*(2), 498 - 516.

Murairwa, S. (2010). *A Hybrid Heuristic Research Framework: The Travelling Salesperson Problem.* Kedah: Univerisiti Utara Malaysia.

Murairwa, S. (2020). *Third Generation Heuristic: Hybrid Erosion And Deposition (HEAD) Metaheuristic.* Mauritius: LAP Lambert Academic Publishing.

Murairwa, S., & Nazri, E. M. (2010). A Multivariate Data Analysis Procedure for Measuring and Comparing the Quality of Performance of Heuristics. *Research India Publications: International Journal of Statistics and Systems , 5*(4), 555 – 567.

Richter, D., Goldengorin, B., Jäger, G., & Molitor, P. (2007). Improving the Efficiency of Helsgaun’s Lin-Kernighan Heuristic for the Symmetric TSP. In J. Janssen, & P. Prałat, *Combinatorial and Algorithmic Aspects of Networking.* Berlin, Heidelberg: Springer. doi:10.1007/978-3-540-77294-1_10

Sze, S. N., & Tiong, W. K. (2007). A Comparison between Heuristic and Meta-Heuristic Methods for Solving the Multiple TSP. *International Journal of Computational and Mathematical Sciences, 1*(3), 200 – 2003.

Vahid, F., & Le, D. T. (1997). Evaluating the Kernighan–Lin Heuristic for Hardware/Software Functional Partitioning. *Design Automation for Embedded Systems, 17*(2), 237 – 261. Retrieved from https://www.cs.ucr.edu/~vahid/pubs/daes97_kl.pdf

Villa, T. C., & Castillo, O. (2020). Adaptation of Parameters with Binary Cat Swarm Optimization Algorithm of Controller for a Mobile Autonomous Robot. In O. Castillo, & P. Melin (Eds.), *Hybrid Intelligent Systems in Control, Pattern Recognition and Medicine* (pp. 34 - 46). Springer. doi:10.1007/978-3-030-34135-0_4