The Effect of Genetic Algorithm Parameters Tuning for Route Optimization in Travelling Salesman Problem through General Full Factorial Design Analysis

Nisrina, Nora
Industrial Engineering Department, Faculty of Engineering, Universitas Indonesia

Muhammad Irfan Kemal
Industrial Engineering Department, Faculty of Engineering, Universitas Indonesia

Ilham Ali Akbar
Industrial Engineering Department, Faculty of Engineering, Universitas Indonesia

Widianti, Tri
Industrial Engineering Department, Faculty of Engineering, Universitas Indonesia

https://doi.org/10.5109/4774233

出版情報：Evergreen. 9 (1), pp.163-203, 2022-03. Transdisciplinary Research and Education Center for Green Technologies, Kyushu University
バージョン：

The Effect of Genetic Algorithm Parameters Tuning for Route Optimization in Travelling Salesman Problem through General Full Factorial Design Analysis

Nora Nisrina[1,*], Muhammad Irfan Kemal[1], Ilham Ali Akbar[1], Tri Widianti[1,2]
[1] Industrial Engineering Department, Faculty of Engineering, Universitas Indonesia, Indonesia
[2] National Research and Innovation Agency of the Republic of Indonesia, Indonesia

*Author to whom correspondence should be addressed:
E-mail: noranisrina@gmail.com

(Received February 11, 2022; Revised March 22, 2022; accepted March 22, 2022).

Abstract: This study aims to analyze the effect of genetic algorithm parameters on the distribution mileage of the largest logistics service provider in Central Jakarta using a general full factorial design and ANOVA test. This study revealed that all factors and three types of interactions were statistically significant in influencing the distribution mileage. Moreover, the combination of population size, crossover probability, mutation probability, and the number of iterations = (90, 1.00, 0.010, 800) generated the lowest mean value and was significantly different from other combinations. This study provides a different approach to analyzing those factors and interactions that influence finding the shortest route.

Keywords: Genetic algorithms; population size; crossover probability; mutation probability; iteration number; shortest route; general full factorial design; ANOVA; TSP

1. Introduction

Reliability is one of the dimensions of service quality that shows its ability to provide reliable and accurate services to customers[1)]. In logistics companies, the reliability dimension can be interpreted as the accuracy of the delivery of goods as promised. Wenyong et al.[2)] proved this dimension was the most important for service quality in logistics companies. According to [3)], service quality positively impacts customer satisfaction, consequently affects performance[4)]. One of the most critical problems in logistics companies is accuracy and delivery speed related to the reliability dimension. The problem is the selection of distribution routes is one of the causes. The wrong choice of distribution route might lead to increased cost, late delivery, and customer dissatisfaction[5)]. This condition might also be the source of company loss, especially for the company whose core business is the logistics service provider. Hence, optimizing the distribution route that leads to a decrease in time and cost of transportation is essential[6)]. It means, by determining the best distribution route, the logistics companies might increase their performance and quality of service to meet customers' specific needs and requirements, especially in the reliability dimension.

Routing problems can be approached with the traveling salesman problem (TSP). TSP is a concept of finding the shortest route where each city is only visited once, and after all cities are visited, the salesman will return to the first city[7)]. TSP is a non-deterministic polynomial-time (NP-Hard) problem. Solving the issues needs metaheuristic or heuristic approaches such as genetic algorithm (GA). The application of GA was quite broad, including Shaedi et al.[8)] for entropy minimization of two-phase flow and Muniandy et al.[9)] for TSP. GA is empirically proven to solve TSP[9)] better than other algorithms[10)]. GA, introduced by John Holland in the 1970s, was inspired by the "survival of the fittest" theory[11)]. This algorithm imitates the evolution process in finding the optimal solution[12)].

Some factors that affect the performance of GA are population size, crossover probability, and mutation probability[13,14)], and the number of iterations[15)]. Roeva et al.[16)] also argued that the number of populations in GA affects the accuracy and the speed of finding the optimal solution. Therefore, the right choice of the number of populations, crossover probability, mutation probability, and the number of iterations in GA is essential.

Previous research used Central Composite Design (CCD)[17)] and fractional experimental design[18)] to investigate the factors. But the interactions between factors were not fully investigated, and none of them identified the number of iterations as a separate factor in their studies[17,18)]. We have not found any general full factorial design studies in TSP cases using GA based on the literature study. Therefore, this study aims to analyze the influence of the number of populations, crossover probability, mutation probability, and the number of iterations, and their interactions toward the optimal solution — in this case, is the shortest route. In this study, to assess the influence of those factors and interactions, the general full factorial was designed to conduct the experiment and analyzed using the analysis of variance (ANOVA) method using Minitab software.

In this study, we used NP-Hard symmetric TSP of distribution case of the largest logistics service provider in Indonesia. The number of nodes is 43 based on the average number of branch offices in the Central Jakarta area[19]. This study focused on finding the shortest route to deliver packages to all branch offices in the Central Jakarta area. We ran a GA simulation using MATLAB software to find the shortest route. There are some limitations in this study, as follows:

1. This study only considered the branch offices of the logistics service provider in the Central Jakarta area.
2. Routing was limited to 43 branch offices of the logistics service provider in the Central Jakarta area.
3. The Euclidean distance between branch offices was determined using the branch's actual coordinate point on maps (symmetric TSP).
4. The simulation process was conducted using MATLAB.

The remainder of this paper is structured as follows: Section 2 presents the theoretical background of TSP, GA, factorial design, and ANOVA as the methods used in this study. Section 3 presents the research design, while Section 4 briefly presents the research methodology. Section 5 provides the result and detailed analysis of the findings. Finally, the conclusion, as well as suggestions for future studies, are presented in Section 6.

2. Theoretical Background

In this section, we will briefly explain the brief theories related to the traveling salesman problem (TSP), genetic algorithm (GA), factorial design, and analysis of variance (ANOVA).

2.1 Travelling Salesman Problem (TSP)

The traveling salesman problem (TSP) could be illustrated as a problem of a salesperson traveling to many destinations with one stop at each destination and returning to the starting point with the expectation of minimum total costs[20]. TSP is a classical non-polynomial problem applicable for routing and scheduling problems[20]. GA is one of the methods that can solve TSP[21] and has been implemented previously[22–25].

2.2 Genetic Algorithm (GA)

A genetic algorithm (GA) is a random search method that imitates the evolution process of living organisms[26]. Liu & Zeng[23] has explained that the development of the natural population follows the principle of natural selection, i.e., those who are strong will survive. Natural selection makes individuals compete to survive and reproduce[12]. GA has both exploration and exploitation mechanisms[11]; flexible — can be combined with other methods[27,28], and it has a mechanism to escape from optimal local solution[29]. However, there are remaining issues related to GA: the determination of population size is complicated[30], needed to achieve an optimal solution is relatively long[31,32], and convergence is hard to achieve[33].

In general, solution search in GA begins with population coding and determination of fitness function[12], then followed by six steps of GA's cycle: initial population generation, fitness evaluation, mechanism reproduction (crossover and mutation), parents' selection, elitism, and new population generation[12,34]. The cycle continues until the best population, optimal solution which meets the fitness value is obtained[21,34].

Solution search in GA depends on crossover and mutation probability. Generally, the crossover probability is in the range of 0.1-1[12], while the range of mutation probability is 0.001-0.2[12].

2.3 Factorial Design

Factorial design is a well-known technique based on statistical considerations that can produce meaningful information about the influence of a factor in the problem, including the effect of interactions between factors[35,36]. The main effect analysis and the interaction plot are the analytical steps to obtain the optimal combination of interactions[37]. However, these two techniques should not be relied on solely because visual subjectivity is often inaccurate[38]. Then the statistical test analysis of variance (ANOVA) was carried out to check whether there was a difference from the mean between two or more sample groups. ANOVA had been widely used in various studies[39-49].

There are four assumptions used in testing the hypothesis on ANOVA: all populations are normally distributed; the population variance is the same (homogeneity); observations are independent (the occurrence of one individual value does not affect the probability of the occurrence of another event), and the data are in the form of interval or rate ratio[50]. On the other hand, we should test the data adequacy with a power data test. The power test is a method to assess the sufficiency of the observed sample size. The power of a hypothesis test is affected by the significance level test, sample size, mean differences, and data variability. An experiment might fail to detect the effects and mistakenly conclude that none exists if the test has low power[51].

The parameters used in the ANOVA test are the sum of square, degree of freedom, mean-square, and F-ratio or p-value, which represents the probability of error. A common practice in statistical testing is to assume a 95% confidence level in the results for the effect to be categorized as statistically significant, which means that the p-value must be less than 0.05[31]. Meanwhile, if the test is rejected, further tests can be conducted to determine which treatment average has the most role with the least significant difference (LSD) test[52)].

3. Research Design

Using the GA simulation program, this study focused on finding the shortest route in packages distribution of Indonesia's largest logistics service provider. This study considered four factors that might influence the simulation output: population size, crossover probability, mutation probability, and the number of iterations. This study was designed using a general full factorial design and analyzed using the ANOVA method. Nine population factor level numbers (20, 30, 40, 50, 60, 70, 80, 90, and 100)[16)]. The population size of 20 to 100 can increase the optimal solution and provide accurate results[16)]. The crossover probability levels were 0.75[16)] and 1. One is the highest value of the crossover range that is usually used in genetic algorithms[12)]. The highest value was chosen based on previous research findings[13)], which stated that the higher the crossover value, the higher the success of obtaining the shortest route[13)]. Mutation probability factor levels were 0.001 and 0.01. The value of 0.001 is the smallest mutation probability value commonly used in genetic algorithms[12)], and the value of 0.01 refers to Roeva et al.[16)]. Then, iteration numbers factor levels were 800 and 1000. These values were chosen because between population sizes 20 to 100; the values can provide a stable fitness value to avoid premature convergence in genetic algorithms[13)]. This amount meets the requirements to obtain a power data value of 0.8[53)].

This study used nine replications. This number met the requirements to obtain a power data value of 0.8[53)]. Then, the null hypothesis (H_0) and alternative hypothesis (H_1) in this study, as follows:

1. $H_{0(a\text{-}d)}$: The treatment of (a) population size, (b) crossover probability, (c) mutation probability, and (d) number of iterations have no effect on the shortest route.
2. $H_{1(a\text{-}d)}$: The treatment of (a) population size, (b) crossover probability, (c) mutation probability, and (d) number of iterations affect the shortest route.

4. Research Methodology

The methodology employed in this study follows an eight steps procedure illustrated in Fig. 1. In the beginning, the literature review was conducted to identify the research subject and unit analysis. Next, this study determined the factors, level of each factor, number of replications, and other relevant parameters. After the research design was set, distances between branch offices in Central Jakarta were measured. Euclidean distances were calculated based on their actual coordinate point on maps. Next, this study collected the shortest route data. Data were collected randomly by performing simulations of GA for TSP in MATLAB software. The running order was determined by random sequences generated from general factorial design in Minitab software. Then, we collected 648 data since we considered nine levels of population, two levels of crossover probability, two levels of mutation probability, and two levels of the number of iterations with nine replications. The power data test was conducted to determine the research data sufficiency. Afterward, this study performed normality, homogeneity, and independence tests in Minitab software as they are ANOVA assumptions that must be fulfilled. A residual normality test was conducted. The homogeneity test was performed using multiple comparisons and Levene's method. After all, assumptions were fulfilled, an ANOVA test was performed in Minitab. The LSD test would be performed if the null hypothesis (H_0) was rejected. An analysis was carried out after the experiment's outcome. Finally, this study was concluded, and we suggested some future research directions.

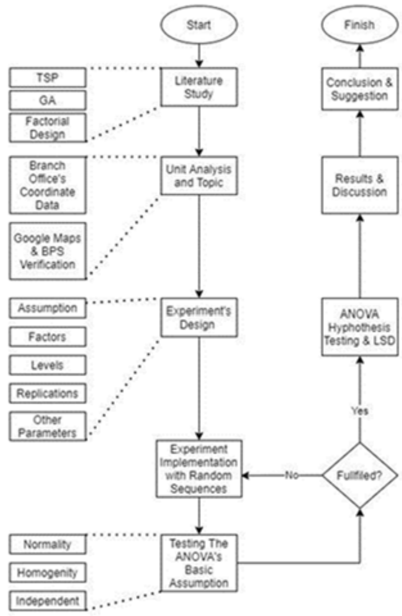

Fig. 1: Research Methodology

5. Result and Discussion

The research data were collected using a genetic algorithm simulation program and can be found in Appendix A1. The results are as follow:

5.1 Power and Sample Size Test

After data collection, we first performed a power test in Minitab to assess the adequacy of the observed sample size[51)]. Based on the power test with the parameters of maximum difference of 9784.8, replication of 9, total runs of 648, and the standard deviation of 4054.2 generated power value of 1, as shown in Fig. 2. This power value was above 0.8. It showed that the amount of research data met the requirements of the expected power value[53)].

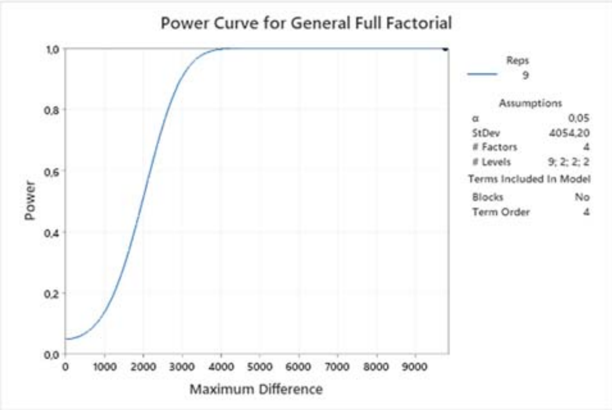

Fig. 2: Power Curve for General Full Factorial

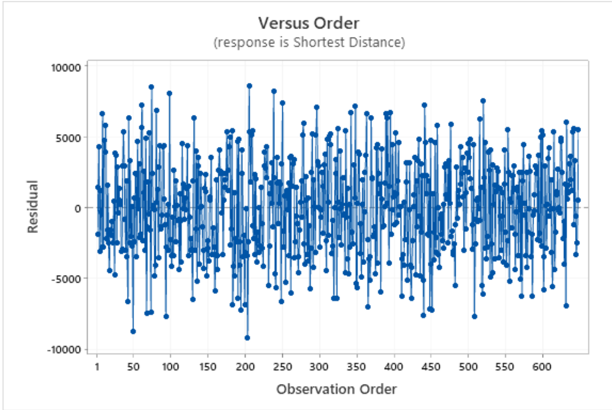

Fig. 4: Test for Equal Variance

5.2 Testing of the Assumptions

Like the *t*-test, ANOVA is also a parametric test and has some assumptions. ANOVA assumes that the residuals are normally distributed. The ANOVA also assumes homogeneity of variance, which means that the variance among the groups should be approximately equal, and the observations are independent of each other. The first step in the ANOVA assumption test is to perform a residual normality test. The probability plot showed the results tend to form a straight line indicating that the residual of the data was normally distributed (see Fig. 3).

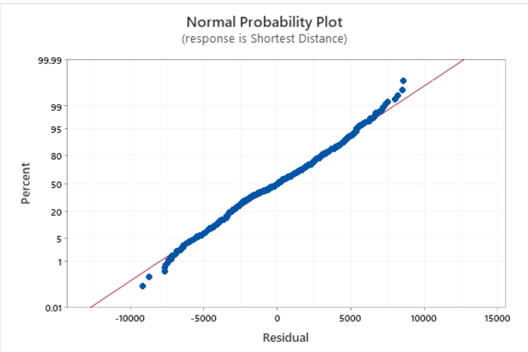

Fig. 3: Normal Probability Plot of Residual

The next step is to perform a homogeneity test to assess the uniformity of variance of each treatment group. The homogeneity test of the data was performed using multiple comparisons and Levene's method. The hypotheses are H_0: there is no difference in the treatment variance and H_1: at least one different treatment variance. Fig. 4 shows the *p*-value of 0.242 in the multiple comparison method and the *p*-value of 0.581 in Levene's test method, more significant than 0.05. It indicates that H_0 was accepted, and the data variance was homogeneous.

Then, the residuals versus order plots are used to verify the assumption that the residuals are independent of one another. Independent residuals show no trends or patterns when displayed in time order. Patterns in the points indicate that residuals near each other may be correlated, thus not independent. Ideally, the residuals on the plot should fall randomly around the centerline[54]. Fig. 5 shows that the residuals were scattered randomly around the center line and did not form any pattern. It showed that the data from the observations were independent of each other.

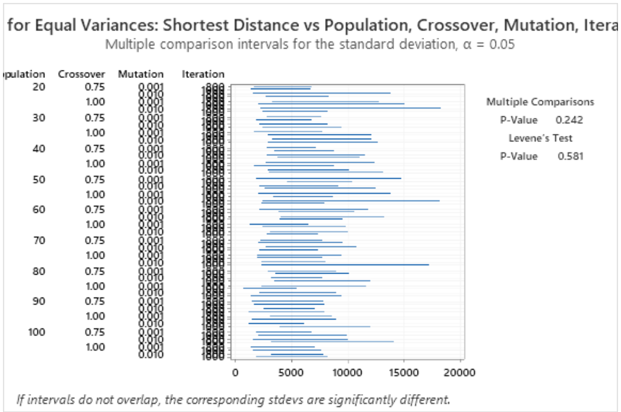

Fig. 5: Residual versus Order Plot

The fulfillment of these three parameters indicates that the data obtained from the observations have met the ANOVA assumption to be processed and analyzed further. Next, an experimental analysis of general full factorial design will be carried out using the ANOVA test, the influence of the main effect factor, the interactions between factors, and the least significant difference test.

5.3 ANOVA Analysis

The results of previous statistical tests ANOVA assumptions were met. Hypothesis testing with factorial ANOVA was carried out to determine whether there was an effect of differences in the value of population size, crossover probability, mutation probability, number of iterations, and the interactions between these factors on the response variable (shortest route). The ANOVA test was carried out with the hypothesis in the research design section. H_0 was accepted if the value of *p*-value > the value of $\alpha = 0.05$. The results of the ANOVA test are in Table 1.

Based on the ANOVA test results, all main effects and three types of interactions were significant. For each main effect (population size, crossover probability, mutation probability, and the number of iterations), the *p*-value was < 0.05. For the interaction between crossover probability and mutation probability, the *p*-value was 0.015. The *p*-value of the interaction between population size, crossover probability, and the number of iterations was 0.001. Then for the interaction of all factors, the *p*-value was 0.019. The results indicated that the four factors and their interactions affected the response variables.

These factors' influences and interactions can be seen from the *p*-value in the ANOVA test and the plot graphs. Fig. 6 shows the main effects graph. All four factors had significant effects on the value of the resulting response variable, where the higher the population size, crossover probability, mutation probability, and the number of iterations used tend to reduce the average shortest route generated.

Although the interaction plot graph could only show interactions between 2-dimensional variables, they could describe the interactions between factors, as shown in Fig. 7. Based on the results of the ANOVA test from the graph, there were striking linear differences in the interaction between crossover and mutation at different combination levels.

Table 1. Analysis of Variance

Source	DF	Adj SS	Adj MS	*F*-Value	*p*-Value
Model	71	3043021881	42859463	3.25	0.000
Linear	11	1707005541	155182322	11.75	0.000
Population	8	1311502048	163937756	12.42	0.000
Crossover	1	150542108	150542108	11.40	0.001
Mutation	1	189034420	189034420	14.32	0.000
Iteration	1	54420786	54420786	4.12	0.043
2-Way Interactions	27	540609535	20022575	1.52	0.047
Population*Crossover	8	96623315	12077914	0.91	0.504
Population*Mutation	8	173143613	21642952	1.64	0.111
Population*Iteration	8	162950468	20368808	1.54	0.139
Crossover*Mutation	1	78855830	78855830	5.97	0.015
Crossover*Iteration	1	739678	739678	0.06	0.813
Mutation* Iteration	1	26548197	26548197	2.01	0.157
3-Way Interactions	25	550407346	22016294	1.67	0.023
Population*Crossover*Mutation	8	31084086	3885511	0.29	0.968
Population*Crossover*Iteration	8	362285548	45285693	3.43	0.001
Population*Mutation*Iteration	8	135642663	16955333	1.28	0.249
Crossover*Mutation*Iteration	1	19845112	19845112	1.50	0.221
4-Way Interactions	8	244748162	30593520	2.32	0.019
Population*Crossover*Mutation*Iteration	8	244748162	30593520	2.32	0.019
Error	575	7591391889	13202421		
Total	646	10634413770			

From the ANOVA, we can generate a regression equation that represents the regression line in the form of an algebraic equation. This regression equation can describe the relationship between the terms in the model and the resulting response51) and predict the value of the response variable. The regression equation from this study can be seen in Appendix A2.

5.4 Least Significant Difference (LSD) Analysis

The LSD test is a follow-up test after knowing the main effects and interactions that affect the response of the shortest route variable. The Least Significant Difference (LSD) test is used in the context of the ANOVA. When the *F*-ratio suggests rejecting the null hypothesis (H_0), the difference between the population means is significant. This test helps to identify the populations whose means are statistically different. The basic idea of the test is to compare the populations taken in pairs[55]. The results of the LSD test are in Table 2.

Fig. 6: Main Effect for Shortest Distance

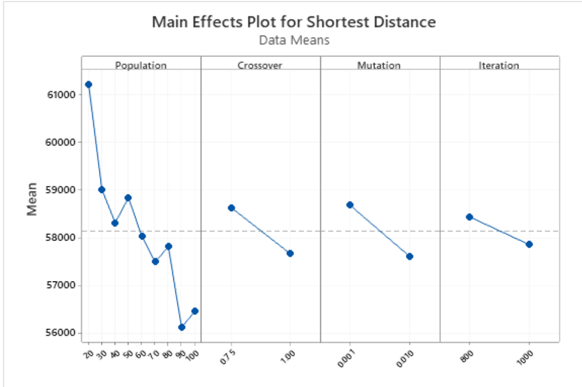

Fig. 7: Interaction Plot for Shortest Distance

Table 2 shows the four combinations of factors and levels that generate the highest mean value and four combinations of factors and levels that generate the lowest mean response value. These results indicated that the combination of population size of 20, crossover probability of 0.75, mutation probability of 0.001, and the number of iterations of 800 generated the highest mean value and is significantly different from other combinations. The combination of population size of 90, crossover probability of 1.00, mutation probability of 0.010, and the number of iterations of 800 generated the lowest mean value and was significantly different from other combinations. Thus, that combination is the best combination of factors and levels in the context of this study.

Theoretically, this study gives new knowledge on the GA and TSP fields. It provides a good combination parameter setting (to gain the shortest route) used in GA to solve the TSP problem. It also provides a new understanding of the design of experiments usage in determining the influence of factors in the genetic algorithm model. As a managerial implication, this study offers a new strategy for the logistics companies to find the shortest route needed to determine the distribution route. The companies can use GA simulation and consider the best combination of factors in the research findings.

Table 2. Grouping Information using Tukey Method with 95% Confidence Level

Population*Crossover*Mutation*Iteration	**N**	**Mean**	**Grouping**
20 0.75 0.001 800	9	62888.1	A
30 0.75 0.001 1000	9	62295.1	A B
20 0.75 0.001 1000	9	62148.9	A B C
30 0.75 0.001 800	9	61926.9	A B C D
90 0.75 0.010 1000	9	54908.8	D E F G
100 0.75 0.001 800	9	53933.8	E F G
90 1.00 0.001 1000	9	53583.8	F G
90 1.00 0.010 800	9	53103.3	G

6. Conclusion

This study proved that the treatment of the population size, crossover probability, mutation probability, and the number of iterations affected the results of the shortest route in the distribution mileage case simulation of the largest logistics service provider in the Central Jakarta area using genetic algorithms. Two treatment conditions generated significant effects. First, the combination of population size of 20, crossover probability of 0.75, mutation probability of 0.001, and the number of iterations of 800 generated the longest route. Second, the combination of population size of 90, crossover probability of 1.00, mutation probability of 0.01, and the number of iterations of 800 significantly generated the shortest route in the observed case. The logistics companies can use these conditions for finding the shortest route solution when using genetic algorithms, which can help to improve their service quality in terms of reliability and delivery time.

The results of this study were in line with previous studies[13,14,16–18]. They stated that some factors that affect the performance of GA were population size, crossover probability, and mutation probability. This finding was also in line with previous research by Younas et al.[15] that said that the number of iterations affects the performance of GA. This study also proved that larger population size, crossover probability, mutation probability, and the number of iterations produce better response variables (shortest route).

The limitation of this study lies in the range of treatment levels for crossover probability, mutation probability, and the number of iterations factors which can be said to be still small (only two levels each). In future research, other treatment levels could be added to provide a broader picture of the effect of population size, crossover probability, mutation probability, and iteration numbers on the genetic algorithm's shortest route search. Aside from that, the power test employed in this study is confined to the data's overall power; in the future, it would be better to examine the power of each main effect and interaction. It is also important to analyze the ability of those four factors and their interactions to explain the model that has been made both linearly and nonlinearly. In addition, the use of actual distance data that considers the highway route in determining the 43 distribution points is recommended. In this study, the fixed method was chosen to determine the levels of each factor. The disadvantage of this method is that inference only applies to those levels. Therefore, in future research, it is recommended to use the random method for level selection so that the inference accommodates the whole population level.

Acknowledgments

All authors are the main contributors to the research. This work could not have been completed without the help of Dr. Zulkarnain, ST., MT. as a supervisor who has guided the authors.

References

1) L.L. Berry, A. Parasuraman, and V.A. Zeithaml, "SERVQUAL: a multiple-item scale for measuring consumer perceptions of service quality," *J. Retail.*, **64** (*1*) 12–40 (1988).
2) Z. Wenyong, Z. Jing, and C. Hongxiang, "Service quality evaluation for international freight forwarder," in: 2010 7th Int. Conf. Serv. Syst. Serv. Manag., IEEE, 2010: pp. 1–5. doi:10.1109/ICSSSM.2010.5530185.
3) S. Subhashini, and S. Preetha, "An empirical analysis of service quality factors pertaining to ocean freight forwarding services," *Marit. Bus. Rev.*, **3** (*3*) 276–289 (2018). doi:10.1108/MABR-01-2018-0004.
4) M.C.H. Yeung, and C.T. Ennew, "From customer satisfaction to profitability," *J. Strateg. Mark.*, **8** (*4*) 313–326 (2000). doi:10.1080/09652540010003663.
5) T. Bosona, "Urban freight last mile logistics—challenges and opportunities to improve sustainability: a literature review," *Sustain.*, **12** (*21*) 1–20 (2020). doi:10.3390/su12218769.
6) L. Pečený, P. Meško, R. Kampf, and J. Gašparík, "Optimisation in transport and logistic processes," *Transp. Res. Procedia*, **44** 15–22 (2020). doi:10.1016/j.trpro.2020.02.003.
7) M. Reza, M. Rahimi, and H. Shah-Hosseini, "Population-Based Optimization Algorithms for Solving the Travelling Salesman Problem," in: Travel. Salesm. Probl., InTech, 2008. doi:10.5772/5586.
8) S.A. Shaedi, N. Mohd-Ghazali, J.T. Oh, R. Ahmad, and Y. Mohd-Yunos, "Entropy generation minimization of two-phase flow in a mini channel with genetic algorithm," *Evergreen*, **6** (*1*) 39–43 (2019). doi:10.5109/2321004.
9) M.A.P. Muniandy, L.K. Mee, and L.K. Ooi, "Efficient route planning for travelling salesman problem," in: 2014 IEEE Conf. Open Syst., IEEE, 2014: pp. 24–29. doi:10.1109/ICOS.2014.7042404.
10) K. Bryant, "Genetic Algorithms: The Travelling Salesman Problem," Harvey Mudd College, 2000. https://medium.com/@becmjo/genetic-algorithms-and-the-travelling-salesman-problem-d10d1daf96a1.
11) J.H. Holland, "Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence," The MIT Press, Cambridge, 1992. https://books.google.com.br/books?id=JE5RAAAAMAAJ.
12) Y. Arkeman, K.B. Seminar, and H. Gundawan, "Algoritma Genetika Teori dan Aplikasinya untuk Bisnis dan industri," IPB Press, 2012.
13) Y. Zhang, Q. Ma, M. Sakamoto, and H. Furutani, "Effects of population size on the performance of genetic algorithms and the role of crossover," *Artif. Life Robot.*, **15** (*2*) 239–243 (2010). doi:10.1007/s10015-010-0836-1.
14) Y. Zhang, M. Sakamoto, and H. Furutani, "Effects of Population Size and Mutation Rate on Results of Genetic Algorithm," in: 2008 Fourth Int. Conf. Nat. Comput., IEEE, 2008: pp. 70–75. doi:10.1109/ICNC.2008.345.
15) I. Younas, F. Kamrani, C. Schulte, and R. Ayani, "Optimization of task assignment to collaborating agents," in: 2011 IEEE Symp. Comput. Intell. Sched., IEEE, 2011: pp. 17–24. doi:10.1109/SCIS.2011.5976547.
16) O. Roeva, S. Fidanova, and M. Paprzycki, "Population size influence on the genetic and ant algorithms performance in case of cultivation process modeling," *Stud. Comput. Intell.*, **580** (*October 2014*) 107–120 (2015). doi:10.1007/978-3-319-12631-9_7.
17) F. Kolahan, and M.H. Doughabadi, "The effects of parameter settings on the performance of genetic

algorithm through experimental design and statistical analysis," *Adv. Mater. Res.*, **433–440** 5994–5999 (2012). doi:10.4028/www.scientific.net/AMR.433-440.5994.
18) P. Pongcharoen, W. Chalnate, and P. Thapatsuwan, "Exploration of genetic parameters and operators through travelling salesman problem," *ScienceAsia*, **33** (*2*) 215–222 (2007). doi:10.2306/scienceasia1513-1874.2007.33.215.
19) BPS, "Jumlah kantor pos pembantu menurut kabupaten/kota (km), 2015 - 2018 2016-2018," *Badan Pus. Stat. Provinsi DKI Jakarta*, (2018). https://jakarta.bps.go.id/indicator/17/320/1/jumlah-kantor-pos-pembantu-menurut-kabupaten-kota-km-2015---2018.html.
20) P. Larranaga, and C. Kuijpers, "Genetic algorithms for the travelling salesman problem: a review of representations and operators," *Artif. Intell. Rev.*, **97** (*1–4*) 131–141 (2000).
21) S. Akter, N. Nahar, M. ShahadatHossain, and K. Andersson, "A New Crossover Technique to Improve Genetic Algorithm and Its Application to TSP," in: 2019 Int. Conf. Electr. Comput. Commun. Eng., IEEE, 2019: pp. 1–6. doi:10.1109/ECACE.2019.8679367.
22) Y. Deng, Y. Liu, and D. Zhou, "An improved genetic algorithm with initial population strategy for symmetric tsp," *Math. Probl. Eng.*, **2015** 1–6 (2015). doi:10.1155/2015/212794.
23) F. Liu, and G. Zeng, "Study of genetic algorithm with reinforcement learning to solve the tsp," *Expert Syst. Appl.*, **36** (*3*) 6995–7001 (2009). doi:10.1016/j.eswa.2008.08.026.
24) F. Yu, X. Fu, H. Li, and G. Dong, "Improved Roulette Wheel Selection-Based Genetic Algorithm for TSP," in: 2016 Int. Conf. Netw. Inf. Syst. Comput., IEEE, 2016: pp. 151–154. doi:10.1109/ICNISC.2016.041.
25) Y. Yu, Y. Chen, and T. Li, "A New Design of Genetic Algorithm for Solving TSP," in: 2011 Fourth Int. Jt. Conf. Comput. Sci. Optim., IEEE, 2011: pp. 309–313. doi:10.1109/CSO.2011.46.
26) C. Fu, L. Zhang, X. Wang, and L. Qiao, "Solving tsp problem with improved genetic algorithm," *AIP Conf. Proc.*, **1967** (*May 2018*) (2018). doi:10.1063/1.5039131.
27) P.A. Vikhar, "Evolutionary algorithms: A critical review and its future prospects," in: 2016 Int. Conf. Glob. Trends Signal Process. Inf. Comput. Commun., IEEE, 2016: pp. 261–265. doi:10.1109/ICGTSPICC.2016.7955308.
28) Z. Zukhri, "Algoritma Genetika: Metode Komputasi Evolusioner untuk Menyelesaikan Masalah Optimasi," Andi, Yogyakarta, 2014.
29) M. Kumar, and H. Banka, "Changing mutation operator of genetic algorithms for optimizing multiple sequence alignment," **3** (*5*) 465–470 (2013).
30) T. El-Mihoub, A.A. Hopgood, L. Nolle, and A. Battersby, "Hybrid genetic algorithms : a review," *Eng. Lett.*, **11** (*August*) 124–137 (2006). file://c/Users/Juan/Documents/Delft/Literature/Bibliography.
31) K. De Jong, "Genetic Algorithms: A 30-Year Perspective," in: Perspect. Adapt. Nat. Artif. Syst., Oxford University Press, 2005. doi:10.1093/oso/9780195162929.003.0006.
32) P. Preux, "Towards hybrid evolutionary algorithms," *Int. Trans. Oper. Res.*, **6** (*6*) 557–570 (1999). doi:10.1016/S0969-6016(99)00019-2.
33) H. Asoh, and H. Mühlenbein, "On the mean convergence time of evolutionary algorithms without selection and mutation," *Int. Conf. Parallel Probl. Solving from Nat.*, (*February*) 88–97 (1994). doi:10.1007/3-540-58484-6_253.
34) Z. Michalewicz, "Genetic Algorithms + Data Structures = Evolution Programs," Springer Berlin Heidelberg, Berlin, Heidelberg, 1992. doi:10.1007/978-3-662-02830-8.
35) K.Y. Chan, M.E. Aydin, and T.C. Fogarty, "An empirical study on the performance of factorial design based crossover on parametrical problems," in: Proc. 2004 Congr. Evol. Comput. (IEEE Cat. No.04TH8753), IEEE, 2004: pp. 620–627. doi:10.1109/CEC.2004.1330915.
36) C.B.B. Costa, M.R.W. Maciel, and R.M. Filho, "Factorial design technique applied to genetic algorithm parameters in a batch cooling crystallization optimisation," *Comput. Chem. Eng.*, **29** (*10*) 2229–2241 (2005). doi:10.1016/j.compchemeng.2005.08.005.
37) C.B.B. Costa, E.A.C. Rivera, M.C.A.F. Rezende, M.R.W. Maciel, and R.M. Filho, "Prior detection of genetic algorithm significant parameters: coupling factorial design technique to genetic algorithm," *Chem. Eng. Sci.*, **62** (*17*) 4780–4801 (2007). doi:10.1016/j.ces.2007.03.042.
38) D.C. Montgomery, "Design and Analysis of Experiments," 8th ed., John Wiley & Sons, Inc., 2012.
39) S. Hartini, Y. Fiantika, Y. Widharto, and M. Hisjam, "Optimal treatment combination for dishwashing liquid soap based on waste cooking oil according to the requirement of indonesian quality standards," *Evergreen*, **8** (*2*) 492–498 (2021). doi:10.5109/4480734.
40) A.K. Srivastava, M. Maurya, A. Saxena, N.K. Maurya, and S.P. Dwivedi, "Statistical optimization by response surface methodology of process parameters during the cnc turning operation of hybrid metal matrix composite," *Evergreen*, **8** (*1*) 51–62 (2021). doi:10.5109/4372260.
41) L.N. Patil, and H.P. Khairnar, "Investigation of human safety based on pedestrian perceptions associated to silent nature of electric vehicle," *Evergreen*, **8** (*2*) 280–289 (2021). doi:10.5109/4480704.
42) S. Choudhary, A. Sharma, S. Gupta, H. Purohit, and

S. Sachan, "Use of rsm technology for the optimization of received signal strength for lte signals under the influence of varying atmospheric conditions," *Evergreen*, **7** (*4*) 500–509 (2020). doi:10.5109/4150469.
43) M.A. Berawi, A. Darmawan, Gunawan, P. Miraj, and H.A. Rahman, "Land value capture: defining crucial variables difference-in-differences model for residential properties surrounding mrt jakarta stage i," *Evergreen*, **7** (*2*) 253–261 (2020). doi:10.5109/4055228.
44) M.I. Alhamid, N. Nasruddin, Budihardjo, E. Susanto, T.F. Vickary, and M. Arif Budiyanto, "Refrigeration cycle exergy-based analysis of hydrocarbon (r600a) refrigerant for optimization of household refrigerator," *Evergreen*, **6** (*1*) 71–77 (2019). doi:10.5109/2321015.
45) S.P. Dwivedi, N.K. Maurya, and M. Maurya, "Assessment of hardness on aa2014/eggshell composite produced via electromagnetic stir casting method," *Evergreen*, **6** (*4*) 285–294 (2019). doi:10.5109/2547354.
46) S.K. Deb, N. Deb, and S. Roy, "Investigation of factors influencing the choice of smartphone banking in bangladesh," *Evergreen*, **6** (*3*) 230–239 (2019). doi:10.5109/2349299.
47) F. Aliunir, T.Y.M. Zagloel, and R. Ardi, "Discrete-event simulation and optimization of spare parts inventory and preventive maintenance integration model considering cooling down and machine dismantling time factor," *Evergreen*, **7** (*1*) 79–85 (2020). doi:10.5109/2740949.
48) W.P. Raharjo, R. Soenoko, A. Purnowidodo, and A. Choiron, "Characterization of sodium-bicarbonate-treated zalacca fibers as composite reinforcements," *Evergreen*, **6** (*1*) 29–38 (2019). doi:10.5109/2321001.
49) H. Sosiati, N.D.M. Yuniar, D. Saputra, and S. Hamdan, "The influence of carbon fiber content on the tensile , flexural , and thermal properties of the sisal / pmma composites," **09** (*01*) (2022).
50) D.F. Groebner, P.F. Shannon, P.C. Fry, and K.D. Smith, "Business Statistics: A Decison Making Approach," 7th ed., Pearson/Prentice Hall, 2008.
51) Minitab, "What is power?," (n.d.). https://support.minitab.com/en-us/minitab/19/help-and-how-to/statistics/power-and-sample-size/supporting-topics/what-is-power/ (accessed June 22, 2021).
52) A.J. Hayter, "The maximum familywise error rate of fisher's least significant difference test," *J. Am. Stat. Assoc.*, **81** (*396*) 1000 (1986). doi:10.2307/2289074.
53) M.D. Casler, "Fundamentals of experimental design: guidelines for designing successful experiments," *Agron. J.*, **107** (*2*) 692–705 (2015). doi:10.2134/agronj2013.0114.
54) Minitab, "Residual plots for fitted line plot," (n.d.). https://support.minitab.com/en-us/minitab/18/help-and-how-to/modeling-statistics/regression/how-to/fitted-line-plot/interpret-the-results/all-statistics-and-graphs/residual-plots/#residuals-versus-order (accessed June 22, 2021).
55) Y. Dodge, "The Concise Encyclopedia of Statistics," Springer, Neuchâtel, 2008.

Appendix

A1. Raw Data

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
184	1	60	1	0.01	1000	56034.76
207	2	90	1	0.01	800	54560.7
95	3	40	1	0.01	800	61751.98
314	4	50	0.75	0.001	1000	56043.64
446	5	30	1	0.001	1000	60314.67
499	6	100	0.75	0.01	800	56259.77
401	7	70	0.75	0.001	800	54223.92
183	8	60	1	0.01	800	62.520.44
180	9	60	0.75	0.01	1000	53418.87
49	10	80	0.75	0.001	800	64839.79
366	11	20	1	0.001	1000	65.317.92
344	12	80	1	0.01	1000	62265.57
494	13	90	1	0.001	1000	51385.53
140	14	100	0.75	0.01	1000	55217.52
616	15	60	1	0.01	1000	59493.45
477	16	70	1	0.001	800	55751.74
327	17	60	1	0.01	800	51468.31
539	18	60	0.75	0.01	800	56433.25
106	19	60	0.75	0.001	1000	58139.61
24	20	40	1	0.01	1000	56978.39
12	21	30	0.75	0.01	1000	56443.39
399	22	60	1	0.01	800	53385.26
581	23	20	1	0.001	800	60830.09

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
153	24	30	0.75	0.001	800	57148.39
627	25	80	0.75	0.01	800	60153.34
400	26	60	1	0.01	1000	61616.84
262	27	70	1	0.001	1000	53132.29
345	28	90	0.75	0.001	800	60321.62
357	29	100	1	0.001	800	54553.48
620	30	70	0.75	0.01	1000	57588.37
625	31	80	0.75	0.001	800	60589.58
585	32	30	0.75	0.001	800	62150.2
471	33	60	1	0.01	800	52825.31
331	34	70	0.75	0.01	800	53947.54
182	35	60	1	0.001	1000	60984.39
437	36	20	1	0.001	800	65744.27
427	37	100	0.75	0.01	800	59480.34
632	38	80	1	0.01	1000	55642.81
65	39	100	0.75	0.001	800	50892.75
85	40	30	1	0.001	800	61015.23
143	41	100	1	0.01	800	54519.5
394	42	60	0.75	0.001	1000	53116.36
108	43	60	0.75	0.01	1000	62566.66
269	44	80	1	0.001	800	54964.44
605	45	50	1	0.001	800	64673.94
503	46	100	1	0.01	800	55991.3
615	47	60	1	0.01	800	55993.4
307	48	40	0.75	0.01	800	56134.35
31	49	50	1	0.01	800	50264.73

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
127	50	80	1	0.01	800	59259.7
133	51	90	1	0.001	800	53743.05
167	52	40	1	0.01	800	59824.42
3	53	20	0.75	0.01	800	64073.39
587	54	30	0.75	0.01	800	54223.77
458	55	50	0.75	0.001	1000	63752.77
301	56	30	1	0.001	800	59026.24
546	57	70	0.75	0.001	1000	63795.98
516	58	30	0.75	0.01	1000	56270.38
526	59	40	1	0.001	1000	57704.38
607	60	50	1	0.01	800	64686.38
236	61	40	0.75	0.01	1000	63109.96
16	62	30	1	0.01	1000	53640.2
633	63	90	0.75	0.001	800	58543.02
572	64	100	0.75	0.01	1000	59423.9
330	65	70	0.75	0.001	1000	57682.53
391	66	50	1	0.01	800	63918.95
33	67	60	0.75	0.001	800	53346.99
110	68	60	1	0.001	1000	56731.45
381	69	40	1	0.001	800	59569.79
126	70	80	1	0.001	1000	57292.16
7	71	20	1	0.01	800	65340.91
249	72	60	0.75	0.001	800	59236.59
519	73	30	1	0.01	800	50170.28
284	74	100	0.75	0.01	1000	65326.71

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
122	75	80	0.75	0.001	1000	60459.25
573	76	100	1	0.001	800	55059.41
535	77	50	1	0.01	800	58830.24
408	78	70	1	0.01	1000	50256.08
598	79	40	1	0.001	1000	55572.84
591	80	30	1	0.01	800	64439.1
125	81	80	1	0.001	800	57838.82
636	82	90	0.75	0.01	1000	56200.24
428	83	100	0.75	0.01	1000	53308.13
476	84	70	0.75	0.01	1000	59069.6
468	85	60	0.75	0.01	1000	60583.82
189	86	70	1	0.001	800	56769.11
89	87	40	0.75	0.001	800	60061.03
152	88	20	1	0.01	1000	59229.08
565	89	90	1	0.001	800	60491.74
188	90	70	0.75	0.01	1000	60593.05
139	91	100	0.75	0.01	800	57194.15
363	92	20	0.75	0.01	800	60501.08
374	93	30	1	0.001	1000	51426.71
418	94	90	0.75	0.001	1000	54342.47
263	95	70	1	0.01	800	59590.39
439	96	20	1	0.01	800	60252.1
341	97	80	1	0.001	800	54413.4
312	98	40	1	0.01	1000	65970.54
354	99	100	0.75	0.001	1000	56951.84
523	100	40	0.75	0.01	800	55478.15

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
364	101	20	0.75	0.01	1000	57303.17
295	102	20	1	0.01	800	59330.83
192	103	70	1	0.01	1000	51662.86
545	104	70	0.75	0.001	800	59650.68
186	105	70	0.75	0.001	1000	60821.88
624	106	70	1	0.01	1000	52628.85
265	107	80	0.75	0.001	800	61988.58
431	108	100	1	0.01	800	58136.52
29	109	50	1	0.001	800	60468.3
163	110	40	0.75	0.01	800	58056.59
124	111	80	0.75	0.01	1000	53355.43
371	112	30	0.75	0.01	800	58662.8
100	113	50	0.75	0.01	1000	51833.44
547	114	70	0.75	0.01	800	55959.53
245	115	50	1	0.001	800	65868.87
228	116	30	0.75	0.01	1000	58281.13
328	117	60	1	0.01	1000	59563.24
91	118	40	0.75	0.01	800	58026.36
315	119	50	0.75	0.01	800	61505.76
117	120	70	1	0.001	800	58132.46
92	121	40	0.75	0.01	1000	57199.55
51	122	80	0.75	0.01	800	60673.47
466	123	60	0.75	0.001	1000	56366.24
623	124	70	1	0.01	800	61604.08
385	125	50	0.75	0.001	800	59004.83

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
445	126	30	1	0.001	800	56117.1
555	127	80	0.75	0.01	800	54129.4
440	128	20	1	0.01	1000	56834.26
235	129	40	0.75	0.01	800	52365.85
266	130	80	0.75	0.001	1000	59926.03
26	131	50	0.75	0.001	1000	65444.76
634	132	90	0.75	0.001	1000	54095.15
72	133	100	1	0.01	1000	54186.43
209	134	100	0.75	0.001	800	57919.96
481	135	80	0.75	0.001	800	54912.78
303	136	30	1	0.01	800	60531.89
316	137	50	0.75	0.01	1000	59260.25
501	138	100	1	0.001	800	50955.72
270	139	80	1	0.001	1000	57787.45
88	140	30	1	0.01	1000	58412.93
564	141	90	0.75	0.01	1000	54956.12
423	142	90	1	0.01	800	51735.32
226	143	30	0.75	0.001	1000	59259.7
593	144	40	0.75	0.001	800	57059.75
542	145	60	1	0.001	1000	59377.04
17	146	40	0.75	0.001	800	63710.71
147	147	20	0.75	0.01	800	62860.61
339	148	80	0.75	0.01	800	51483.67
613	149	60	1	0.001	800	54950.45
14	150	30	1	0.001	1000	56192.13
82	151	30	0.75	0.001	1000	58523.14

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
288	152	100	1	0.01	1000	56184.53
177	153	60	0.75	0.001	800	61522.17
359	154	100	1	0.01	800	61522.17
574	155	100	1	0.001	1000	53163.21
60	156	90	0.75	0.01	1000	53459.46
350	157	90	1	0.001	1000	52116.06
161	158	40	0.75	0.001	800	58592.63
68	159	100	0.75	0.01	1000	50938.99
646	160	100	1	0.001	1000	53209.88
548	161	70	0.75	0.01	1000	58164.39
404	162	70	0.75	0.01	1000	51907.7
267	163	80	0.75	0.01	800	52948.78
18	164	40	0.75	0.001	1000	60322.97
230	165	30	1	0.001	1000	63165.91
414	166	80	1	0.001	1000	60345.09
336	167	70	1	0.01	1000	57491.87
562	168	90	0.75	0.001	1000	56798.97
219	169	20	0.75	0.01	800	64151.68
138	170	100	0.75	0.001	1000	55829.38
247	171	50	1	0.01	800	59521.71
425	172	100	0.75	0.001	800	55331.14
434	173	20	0.75	0.001	1000	64073.39
479	174	70	1	0.01	800	62523.05
337	175	80	0.75	0.001	800	65473.61
181	176	60	1	0.001	800	58467.7

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
436	177	20	0.75	0.01	1000	64073.39
543	178	60	1	0.01	800	60225.06
473	179	70	0.75	0.001	800	61966.52
305	180	40	0.75	0.001	800	58778.33
508	181	20	0.75	0.01	1000	59966.09
566	182	90	1	0.001	1000	58983.54
384	183	40	1	0.01	1000	51067.01
40	184	60	1	0.01	1000	53764.12
536	185	50	1	0.01	1000	59065.53
97	186	50	0.75	0.001	800	54860.63
639	187	90	1	0.01	800	50955.1
452	188	40	0.75	0.01	1000	53411.79
158	189	30	1	0.001	1000	63750.67
149	190	20	1	0.001	800	53936.48
171	191	50	0.75	0.01	800	65653.16
47	192	70	1	0.01	800	57705.18
324	193	60	0.75	0.01	1000	52133.91
221	194	20	1	0.001	800	53121.79
59	195	90	0.75	0.01	800	62764.92
389	196	50	1	0.001	800	63760.7
448	197	30	1	0.01	1000	52511.37
420	198	90	0.75	0.01	1000	52963.92
453	199	40	1	0.001	800	50898.26
178	200	60	0.75	0.001	1000	56599.39
464	201	50	1	0.01	1000	57977.3
407	202	70	1	0.01	800	56764.83

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
151	203	20	1	0.01	800	50848.77
372	204	30	0.75	0.01	1000	56008.38
480	205	70	1	0.01	1000	63658.6
310	206	40	1	0.001	1000	64991.02
154	207	30	0.75	0.001	1000	64101.42
264	208	70	1	0.01	1000	60167.81
515	209	30	0.75	0.01	800	63118.64
368	210	20	1	0.01	1000	58514.89
449	211	40	0.75	0.001	800	59229.95
529	212	50	0.75	0.001	800	61992.54
444	213	30	0.75	0.01	1000	55094.71
347	214	90	0.75	0.01	800	60732.61
28	215	50	0.75	0.01	1000	51833.44
94	216	40	1	0.001	1000	57881.82
369	217	30	0.75	0.001	800	59574.95
629	218	80	1	0.001	800	54288.57
561	219	90	0.75	0.001	800	57443.84
275	220	90	0.75	0.01	800	56028.78
241	221	50	0.75	0.001	800	61080.76
22	222	40	1	0.001	1000	58463.51
563	223	90	0.75	0.01	800	55862.88
46	224	70	1	0.001	1000	51483.61
352	225	90	1	0.01	1000	60765.55
421	226	90	1	0.001	800	60071.34
544	227	60	1	0.01	1000	60679.67

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
57	228	90	0.75	0.001	800	59545.04
326	229	60	1	0.001	1000	62318.1
164	230	40	0.75	0.01	1000	60054.18
280	231	90	1	0.01	1000	52352.13
292	232	20	0.75	0.01	1000	58712.97
630	233	80	1	0.001	1000	58749.44
239	234	40	1	0.01	800	59185.52
496	235	90	1	0.01	1000	58090.57
78	236	20	1	0.001	1000	64553.71
201	237	90	0.75	0.001	800	56146.49
304	238	30	1	0.01	1000	64187.62
419	239	90	0.75	0.01	800	60819.27
520	240	30	1	0.01	1000	51335.85
380	241	40	0.75	0.01	1000	50198.82
118	242	70	1	0.001	1000	59162.69
50	243	80	0.75	0.001	1000	59761.5
386	244	50	0.75	0.001	1000	57985.2
550	245	70	1	0.001	1000	57442.89
618	246	70	0.75	0.001	1000	61010.1
509	247	20	1	0.001	800	62049.65
614	248	60	1	0.001	1000	51463.32
296	249	20	1	0.01	1000	62890.03
532	250	50	0.75	0.01	1000	63101.97
392	251	50	1	0.01	1000	55248.72
278	252	90	1	0.001	1000	51955.68
406	253	70	1	0.001	1000	56475.27

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
640	254	90	1	0.01	1000	52548.15
36	255	60	0.75	0.01	1000	57295.76
200	256	80	1	0.01	1000	57668.68
169	257	50	0.75	0.001	800	59469.45
353	258	100	0.75	0.001	800	55640.34
227	259	30	0.75	0.01	800	54348.9
647	260	100	1	0.01	800	61519.57
42	261	70	0.75	0.001	1000	54708.84
635	262	90	0.75	0.01	800	62315.88
582	263	20	1	0.001	1000	62890.7
103	264	50	1	0.01	800	62123.79
258	265	70	0.75	0.001	1000	64092.92
109	266	60	1	0.001	800	53765.17
489	267	90	0.75	0.001	800	55047.99
638	268	90	1	0.001	1000	52997.96
559	269	80	1	0.01	800	60026.17
375	270	30	1	0.01	800	56463.92
500	271	100	0.75	0.01	1000	52235.53
215	272	100	1	0.01	800	54535.94
1	273	20	0.75	0.001	800	61311.9
417	274	90	0.75	0.001	800	57823.34
202	275	90	0.75	0.001	1000	59003.79
346	276	90	0.75	0.001	1000	56445.85
212	277	100	0.75	0.01	1000	61970.53
115	278	70	0.75	0.01	800	62826.06

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
512	279	20	1	0.01	1000	57186.86
577	280	20	0.75	0.001	800	58911.81
273	281	90	0.75	0.001	800	59665.17
116	282	70	0.75	0.01	1000	56779.48
497	283	100	0.75	0.001	800	50629.1
492	284	90	0.75	0.01	1000	55558.86
570	285	100	0.75	0.001	1000	55435.48
525	286	40	1	0.001	800	56497.95
205	287	90	1	0.001	800	50415.18
48	288	70	1	0.01	1000	52181.9
309	289	40	1	0.001	800	60229.99
34	290	60	0.75	0.001	1000	64916.19
648	291	100	1	0.01	1000	51423.6
293	292	20	1	0.001	800	62301.2
54	293	80	1	0.001	1000	57828.34
242	294	50	0.75	0.001	1000	55923.15
130	295	90	0.75	0.001	1000	62277.27
451	296	40	0.75	0.01	800	65909.48
317	297	50	1	0.001	800	64591.62
277	298	90	1	0.001	800	56171.2
185	299	70	0.75	0.001	800	59910.78
611	300	60	0.75	0.01	800	55535.4
63	301	90	1	0.01	800	52452.51
198	302	80	1	0.001	1000	57629.52
93	303	40	1	0.001	800	60268.79
268	304	80	0.75	0.01	1000	59212.75

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
413	305	80	1	0.001	800	55610.65
11	306	30	0.75	0.01	800	60752.3
513	307	30	0.75	0.001	800	65677.97
229	308	30	1	0.001	800	56808.89
367	309	20	1	0.01	800	55606.06
132	310	90	0.75	0.01	1000	59705.22
299	311	30	0.75	0.01	800	55769.69
165	312	40	1	0.001	800	58126.14
465	313	60	0.75	0.001	800	65827.7
538	314	60	0.75	0.001	1000	64904.1
32	315	50	1	0.01	1000	62499.24
43	316	70	0.75	0.01	800	53618.85
159	317	30	1	0.01	800	56696.46
208	318	90	1	0.01	1000	55120.71
98	319	50	0.75	0.001	1000	52726.77
285	320	100	1	0.001	800	54935.23
609	321	60	0.75	0.001	800	64154.09
643	322	100	0.75	0.01	800	58984.03
175	323	50	1	0.01	800	52571.99
619	324	70	0.75	0.01	800	62423.18
74	325	20	0.75	0.001	1000	62541.16
2	326	20	0.75	0.001	1000	60054.06
123	327	80	0.75	0.01	800	56200.66
38	328	60	1	0.001	1000	62017.55
254	329	60	1	0.001	1000	55423.46

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
475	330	70	0.75	0.01	800	57111.18
30	331	50	1	0.001	1000	59136.62
517	332	30	1	0.001	800	58583.3
488	333	80	1	0.01	1000	56026.1
80	334	20	1	0.01	1000	55197.25
191	335	70	1	0.01	800	65509.9
213	336	100	1	0.001	800	54284.44
248	337	50	1	0.01	1000	59369.66
447	338	30	1	0.01	800	53967.32
39	339	60	1	0.01	800	51210.55
176	340	50	1	0.01	1000	58239
113	341	70	0.75	0.001	800	54223.92
196	342	80	0.75	0.01	1000	64393.59
498	343	100	0.75	0.001	1000	57474.11
58	344	90	0.75	0.001	1000	54587.76
168	345	40	1	0.01	1000	55355.87
13	346	30	1	0.001	800	62150.2
214	347	100	1	0.001	1000	57224.97
424	348	90	1	0.01	1000	64952.59
412	349	80	0.75	0.01	1000	52354.37
462	350	50	1	0.001	1000	51359.09
75	351	20	0.75	0.01	800	61995.33
583	352	20	1	0.01	800	63931.02
90	353	40	0.75	0.001	1000	61473.59
231	354	30	1	0.01	800	54298
4	355	20	0.75	0.01	1000	60345.78

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
259	356	70	0.75	0.01	800	51936.9
387	357	50	0.75	0.01	800	61031.85
155	358	30	0.75	0.01	800	59682.59
450	359	40	0.75	0.001	1000	53832.24
173	360	50	1	0.001	800	61053.48
187	361	70	0.75	0.01	800	59364.3
377	362	40	0.75	0.001	800	65059.14
197	363	80	1	0.001	800	64039.89
272	364	80	1	0.01	1000	54231.18
107	365	60	0.75	0.01	800	51370.04
174	366	50	1	0.001	1000	53390.05
362	367	20	0.75	0.001	1000	58673.58
67	368	100	0.75	0.01	800	51453.62
210	369	100	0.75	0.001	1000	65497.11
506	370	20	0.75	0.001	1000	64195.32
355	371	100	0.75	0.01	800	58677.74
217	372	20	0.75	0.001	800	60821.13
388	373	50	0.75	0.01	1000	53791.81
470	374	60	1	0.001	1000	56708.36
199	375	80	1	0.01	800	53941.28
81	376	30	0.75	0.001	800	66162.5
244	377	50	0.75	0.01	1000	51791.85
575	378	100	1	0.01	800	62444.68
325	379	60	1	0.001	800	59746.88
8	380	20	1	0.01	1000	60774.92

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
442	381	30	0.75	0.001	1000	63972.76
27	382	50	0.75	0.01	800	54881.36
528	383	40	1	0.01	1000	62029.46
37	384	60	1	0.001	800	57885.06
260	385	70	0.75	0.01	1000	54909.39
56	386	80	1	0.01	1000	55220.85
340	387	80	0.75	0.01	1000	57802.69
114	388	70	0.75	0.001	1000	62460.1
568	389	90	1	0.01	1000	64417.58
172	390	50	0.75	0.01	1000	55967.27
250	391	60	0.75	0.001	1000	66067.76
645	392	100	1	0.001	800	57960.43
405	393	70	1	0.001	800	54353.47
502	394	100	1	0.001	1000	58019.87
484	395	80	0.75	0.01	1000	64432.95
157	396	30	1	0.001	800	57234.23
253	397	60	1	0.001	800	57158.79
534	398	50	1	0.001	1000	61478.67
463	399	50	1	0.01	800	59763.3
289	400	20	0.75	0.001	800	63912.27
617	401	70	0.75	0.001	800	55800.76
588	402	30	0.75	0.01	1000	63626.46
552	403	70	1	0.01	1000	56077.68
370	404	30	0.75	0.001	1000	61275.7
592	405	30	1	0.01	1000	57852.12
602	406	50	0.75	0.001	1000	63685.18

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
642	407	100	0.75	0.001	1000	58622.79
541	408	60	1	0.001	800	57387.73
628	409	80	0.75	0.01	1000	57808.85
52	410	80	0.75	0.01	1000	51466.59
120	411	70	1	0.01	1000	51480.58
5	412	20	1	0.001	800	57070.3
461	413	50	1	0.001	800	57290.56
360	414	100	1	0.01	1000	56046.71
21	415	40	1	0.001	800	55493.54
62	416	90	1	0.001	1000	55388.81
156	417	30	0.75	0.01	1000	61576.31
287	418	100	1	0.01	800	52859.39
571	419	100	0.75	0.01	800	5606..61
121	420	80	0.75	0.001	800	59945.69
376	421	30	1	0.01	1000	59161.26
622	422	70	1	0.001	1000	51544.32
410	423	80	0.75	0.001	1000	51678.05
112	424	60	1	0.01	1000	53661.29
551	425	70	1	0.01	800	57799.77
553	426	80	0.75	0.001	800	60205.21
478	427	70	1	0.001	1000	56490.43
256	428	60	1	0.01	1000	61455.33
129	429	90	0.75	0.001	800	52723.42
351	430	90	1	0.01	800	54964.18
194	431	80	0.75	0.001	1000	56739.51

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
396	432	60	0.75	0.01	1000	59540.26
300	433	30	0.75	0.01	1000	63222.85
560	434	80	1	0.01	1000	55920.05
600	435	40	1	0.01	1000	53107.96
576	436	100	1	0.01	1000	58740.49
141	437	100	1	0.001	800	53177.59
25	438	50	0.75	0.001	800	61881.01
527	439	40	1	0.01	800	52340.56
101	440	50	1	0.001	800	53743.2
467	441	60	0.75	0.01	800	65550.52
373	442	30	1	0.001	800	63728.04
378	443	40	0.75	0.001	1000	58677.2
240	444	40	1	0.01	1000	60176.7
393	445	60	0.75	0.001	800	58898.68
20	446	40	0.75	0.01	1000	53886.54
504	447	100	1	0.01	1000	57520.46
438	448	20	1	0.001	1000	54204.15
261	449	70	1	0.001	800	63015.63
193	450	80	0.75	0.001	800	56104.37
313	451	50	0.75	0.001	800	52409.75
15	452	30	1	0.01	800	58518.55
604	453	50	0.75	0.01	1000	53995.04
291	454	20	0.75	0.01	800	58050.6
554	455	80	0.75	0.001	1000	62618.86
162	456	40	0.75	0.001	1000	54528.16
641	457	100	0.75	0.001	800	51846.46

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
35	458	60	0.75	0.01	800	64143.49
66	459	100	0.75	0.001	1000	59166.95
332	460	70	0.75	0.01	1000	52917.22
597	461	40	1	0.001	800	53524.19
311	462	40	1	0.01	800	56695.62
342	463	80	1	0.001	1000	55900.22
533	464	50	1	0.001	800	60900.85
10	465	30	0.75	0.001	1000	59944.85
211	466	100	0.75	0.01	800	56782.47
279	467	90	1	0.01	800	50261.64
579	468	20	0.75	0.01	800	64546.66
73	469	20	0.75	0.001	800	65959.32
556	470	80	0.75	0.01	1000	58620.73
415	471	80	1	0.01	800	59205.27
586	472	30	0.75	0.001	1000	65217.89
61	473	90	1	0.001	800	53651.33
102	474	50	1	0.001	1000	60296.54
160	475	30	1	0.01	1000	56310.52
55	476	80	1	0.01	800	64461.07
294	477	20	1	0.001	1000	59307.76
99	478	50	0.75	0.01	800	59209.27
409	479	80	0.75	0.001	800	56798.21
644	480	100	0.75	0.01	1000	53773.27
631	481	80	1	0.01	800	56075.05
64	482	90	1	0.01	1000	52324.39

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
343	483	80	1	0.01	800	57362.45
166	484	40	1	0.001	1000	62509.89
83	485	30	0.75	0.01	800	56960.81
637	486	90	1	0.001	800	56945.5
510	487	20	1	0.001	1000	62578.43
454	488	40	1	0.001	1000	59885.44
251	489	60	0.75	0.01	800	62633.01
416	490	80	1	0.01	1000	58109.75
356	491	100	0.75	0.01	1000	59238.63
531	492	50	0.75	0.01	800	64343.15
486	493	80	1	0.001	1000	58483.36
608	494	50	1	0.01	1000	55034.97
119	495	70	1	0.01	800	62873.62
590	496	30	1	0.001	1000	60113.13
281	497	100	0.75	0.001	800	52442.59
220	498	20	0.75	0.01	1000	65287.29
190	499	70	1	0.001	1000	58096
530	500	50	0.75	0.001	1000	63625.95
150	501	20	1	0.001	1000	64871.83
518	502	30	1	0.001	1000	63888.17
218	503	20	0.75	0.001	1000	64206.68
216	504	100	1	0.01	1000	57003.54
349	505	90	1	0.001	800	53208.75
469	506	60	1	0.001	800	56092.03
557	507	80	1	0.001	800	61861.56
179	508	60	0.75	0.01	800	50661.8

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
603	509	50	0.75	0.01	800	58300.01
318	510	50	1	0.001	1000	56189.59
41	511	70	0.75	0.001	800	57540.8
422	512	90	1	0.001	1000	52597.71
45	513	70	1	0.001	800	58920.35
286	514	100	1	0.001	1000	56465.56
482	515	80	0.75	0.001	1000	64319.34
86	516	30	1	0.001	1000	58143.88
148	517	20	0.75	0.01	1000	62917.12
232	518	30	1	0.01	1000	50459.28
540	519	60	0.75	0.01	1000	50150.25
237	520	40	1	0.001	800	65302.5
395	521	60	0.75	0.01	800	60463.57
487	522	80	1	0.01	800	55774.57
457	523	50	0.75	0.001	800	64259.83
522	524	40	0.75	0.001	1000	54083.41
134	525	90	1	0.001	1000	50831.57
44	526	70	0.75	0.01	1000	54216.42
233	527	40	0.75	0.001	800	65274.75
53	528	80	1	0.001	800	60115.02
104	529	50	1	0.01	1000	53121.65
460	530	50	0.75	0.01	1000	59756.16
549	531	70	1	0.001	800	55683.06
252	532	60	0.75	0.01	1000	51588.37
474	533	70	0.75	0.001	1000	63739.44

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
111	534	60	1	0.01	800	56123.41
578	535	20	0.75	0.001	1000	63760.27
485	536	80	1	0.001	800	53850.33
334	537	70	1	0.001	1000	56784.84
495	538	90	1	0.01	800	55920.57
459	539	50	0.75	0.01	800	62317.12
514	540	30	0.75	0.001	1000	64853.34
411	541	80	0.75	0.01	800	52757.24
610	542	60	0.75	0.001	1000	61112.72
398	543	60	1	0.001	1000	57478.43
472	544	60	1	0.01	1000	54936.71
243	545	50	0.75	0.01	800	60245.75
490	546	90	0.75	0.001	1000	58162.55
580	547	20	0.75	0.01	1000	57610.14
195	548	80	0.75	0.01	800	57993.76
483	549	80	0.75	0.01	800	60391.29
455	550	40	1	0.01	800	55712.84
626	551	80	0.75	0.001	1000	54014.81
308	552	40	0.75	0.01	1000	53509.77
87	553	30	1	0.01	800	63040.58
612	554	60	0.75	0.01	1000	58691.89
234	555	40	0.75	0.001	1000	52479.11
430	556	100	1	0.001	1000	57108.35
319	557	50	1	0.01	800	59372.73
601	558	50	0.75	0.001	800	61881.4
321	559	60	0.75	0.001	800	59381.35

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
426	560	100	0.75	0.001	1000	61459.43
282	561	100	0.75	0.001	1000	62135.79
558	562	80	1	0.001	1000	55870.91
302	563	30	1	0.001	1000	54667.84
329	564	70	0.75	0.001	800	56727.89
274	565	90	0.75	0.001	1000	58540.55
298	566	30	0.75	0.001	1000	63506.82
246	567	50	1	0.001	1000	53611.63
306	568	40	0.75	0.001	1000	60230.9
537	569	60	0.75	0.001	800	61481.4
338	570	80	0.75	0.001	1000	53023.26
71	571	100	1	0.01	800	60489.46
23	572	40	1	0.01	800	51169.58
77	573	20	1	0.001	800	62280.77
223	574	20	1	0.01	800	64445.28
131	575	90	0.75	0.01	800	59055.36
142	576	100	1	0.001	1000	56023.26
335	577	70	1	0.01	800	56748.47
379	578	40	0.75	0.01	800	61310.42
491	579	90	0.75	0.01	800	54485.51
443	580	30	0.75	0.01	800	55963.09
204	581	90	0.75	0.01	1000	54859.26
69	582	100	1	0.001	800	58578.53
584	583	20	1	0.01	1000	62313.18
225	584	30	0.75	0.001	800	58448.95

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
441	585	30	0.75	0.001	800	60050.42
170	586	50	0.75	0.001	1000	52873.76
70	587	100	1	0.001	1000	56015.16
505	588	20	0.75	0.001	800	62647.91
238	589	40	1	0.001	1000	60538.34
397	590	60	1	0.001	800	60245.13
206	591	90	1	0.001	1000	55997.46
435	592	20	0.75	0.01	800	64478.37
403	593	70	0.75	0.01	800	54575.41
9	594	30	0.75	0.001	800	65610.01
203	595	90	0.75	0.01	800	56401.37
390	596	50	1	0.001	1000	55187.36
333	597	70	1	0.001	800	63258.03
146	598	20	0.75	0.001	1000	59442.11
76	599	20	0.75	0.01	1000	66916.82
596	600	40	0.75	0.01	1000	60990.97
222	601	20	1	0.001	1000	55592.95
84	602	30	0.75	0.01	1000	54910.18
144	603	100	1	0.01	1000	58608.83
621	604	70	1	0.001	800	58408.35
283	605	100	0.75	0.01	800	53745.45
290	606	20	0.75	0.001	1000	62393.78
255	607	60	1	0.01	800	59361.89
19	608	40	0.75	0.01	800	63101.89
511	609	20	1	0.01	800	59406.34
383	610	40	1	0.01	800	57240.59

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
569	611	100	0.75	0.001	800	55891.53
128	612	80	1	0.01	1000	53310.64
524	613	40	0.75	0.01	1000	50275.81
429	614	100	1	0.001	800	54863.31
606	615	50	1	0.001	1000	62354.52
432	616	100	1	0.01	1000	51272.01
589	617	30	1	0.001	800	57425.75
79	618	20	1	0.01	800	61183.12
137	619	100	0.75	0.001	800	54810.62
271	620	80	1	0.01	800	61296.54
567	621	90	1	0.01	800	52893.39
323	622	60	0.75	0.01	800	58312.98
456	623	40	1	0.01	1000	57524.03
358	624	100	1	0.001	1000	61393.18
493	625	90	1	0.001	800	60838.89
105	626	60	0.75	0.001	800	63263.92
348	627	90	0.75	0.01	1000	52739.52
145	628	20	0.75	0.001	800	61918.54
257	629	70	0.75	0.001	800	53009.52
433	630	20	0.75	0.001	800	64553.9
507	631	20	0.75	0.01	800	54795.63
594	632	40	0.75	0.001	1000	63745.46
297	633	30	0.75	0.001	800	62518.72
135	634	90	1	0.01	800	54186.41
6	635	20	1	0.001	1000	62963.63

Std Order	Run Order	Factors				Shortest Route (Response) in Meter
		Population	Crossover	Mutation	Iteration	
361	636	20	0.75	0.001	800	65956.35
320	637	50	1	0.01	1000	61655.3
136	638	90	1	0.01	1000	59773.67
96	639	40	1	0.01	1000	59103.22
224	640	20	1	0.01	1000	65099.3
365	641	20	1	0.001	800	65931.82
276	642	90	0.75	0.01	1000	53736.84
521	643	40	0.75	0.001	800	64680.95
382	644	40	1	0.001	1000	59056.9
322	645	60	0.75	0.001	1000	56436.07
402	646	70	0.75	0.001	1000	58271.73
595	647	40	0.75	0.01	800	59349.17
599	648	40	1	0.01	800	62899.81

A2. Regression Equation

Shortest Route = 58142 + 3078 Population_20 + 867 Population_30 + 175 Population_40
+ 696 Population_50 - 110 Population_60 - 654 Population_70 - 333 Population_80
- 2028 Population_90 - 1690 Population_100 + 482 Crossover_0.75 - 482 Crossover_1.00
+ 541 Mutation_0.001 - 541 Mutation_0.010 + 290 Iteration_800 - 290 Iteration_1000
+ 351 Population*Crossover_20 0.75 - 351 Population*Crossover_20 1.00
+ 589 Population*Crossover_30 0.75 - 589 Population*Crossover_30 1.00
- 350 Population*Crossover_40 0.75 + 350 Population*Crossover_40 1.00
- 495 Population*Crossover_50 0.75 + 495 Population*Crossover_50 1.00
+ 258 Population*Crossover_60 0.75 - 258 Population*Crossover_60 1.00
- 261 Population*Crossover_70 0.75 + 261 Population*Crossover_70 1.00
- 248 Population*Crossover_80 0.75 + 248 Population*Crossover_80 1.00
+ 463 Population*Crossover_90 0.75 - 463 Population*Crossover_90 1.00
- 308 Population*Crossover_100 0.75 + 308 Population*Crossover_100 1.00
- 70 Population*Mutation_20 0.001 + 70 Population*Mutation_20 0.010

+ 1054 Population*Mutation_30 0.001 - 1054 Population*Mutation_30 0.010
+ 263 Population*Mutation_40 0.001 - 263 Population*Mutation_40 0.010
- 94 Population*Mutation_50 0.001 + 94 Population*Mutation_50 0.010
+ 398 Population*Mutation_60 0.001 - 398 Population*Mutation_60 0.010
- 125 Population*Mutation_70 0.001 + 125 Population*Mutation_70 0.010
- 9 Population*Mutation_80 0.001 + 9 Population*Mutation_80 0.010
- 563 Population*Mutation_90 0.001 + 563 Population*Mutation_90 0.010
- 855 Population*Mutation_100 0.001 + 855 Population*Mutation_100 0.010
- 259 Population*Iteration_20 800 + 259 Population*Iteration_20 1000
- 215 Population*Iteration_30 800 + 215 Population*Iteration_30 1000
+ 251 Population*Iteration_40 800 - 251 Population*Iteration_40 1000
+ 1087 Population*Iteration_50 800 - 1087 Population*Iteration_50 1000
- 239 Population*Iteration_60 800 + 239 Population*Iteration_60 1000
+ 283 Population*Iteration_70 800 - 283 Population*Iteration_70 1000
+ 11 Population*Iteration_80 800 - 11 Population*Iteration_80 1000
- 38 Population*Iteration_90 800 + 38 Population*Iteration_90 1000
- 883 Population*Iteration_100 800 + 883 Population*Iteration_100 1000
+ 349 Crossover*Mutation_0.75 0.001 - 349 Crossover*Mutation_0.75 0.010
- 349 Crossover*Mutation_1.00 0.001 + 349 Crossover*Mutation_1.00 0.010
+ 34 Crossover*Iteration_0.75 800 - 34 Crossover*Iteration_0.75 1000
- 34 Crossover*Iteration_1.00 800 + 34 Crossover*Iteration_1.00 1000
- 203 Mutation*Iteration_0.001 800 + 203 Mutation*Iteration_0.001 1000
+ 203 Mutation*Iteration_0.010 800 - 203 Mutation*Iteration_0.010 1000
- 355 Population*Crossover*Mutation_20 0.75 0.001
+ 355 Population*Crossover*Mutation_20 0.75 0.010
+ 355 Population*Crossover*Mutation_20 1.00 0.001
- 355 Population*Crossover*Mutation_20 1.00 0.010
+ 86 Population*Crossover*Mutation_30 0.75 0.001
- 86 Population*Crossover*Mutation_30 0.75 0.010
- 86 Population*Crossover*Mutation_30 1.00 0.001
+ 86 Population*Crossover*Mutation_30 1.00 0.010
- 57 Population*Crossover*Mutation_40 0.75 0.001
+ 57 Population*Crossover*Mutation_40 0.75 0.010
+ 57 Population*Crossover*Mutation_40 1.00 0.001
- 57 Population*Crossover*Mutation_40 1.00 0.010
- 238 Population*Crossover*Mutation_50 0.75 0.001
+ 238 Population*Crossover*Mutation_50 0.75 0.010
+ 238 Population*Crossover*Mutation_50 1.00 0.001

- 238 Population*Crossover*Mutation_50 1.00 0.010

+ 204 Population*Crossover*Mutation_60 0.75 0.001

- 204 Population*Crossover*Mutation_60 0.75 0.010

- 204 Population*Crossover*Mutation_60 1.00 0.001

+ 204 Population*Crossover*Mutation_60 1.00 0.010

+ 395 Population*Crossover*Mutation_70 0.75 0.001

- 395 Population*Crossover*Mutation_70 0.75 0.010

- 395 Population*Crossover*Mutation_70 1.00 0.001

+ 395 Population*Crossover*Mutation_70 1.00 0.010

+ 153 Population*Crossover*Mutation_80 0.75 0.001

- 153 Population*Crossover*Mutation_80 0.75 0.010

- 153 Population*Crossover*Mutation_80 1.00 0.001

+ 153 Population*Crossover*Mutation_80 1.00 0.010

- 80 Population*Crossover*Mutation_90 0.75 0.001

+ 80 Population*Crossover*Mutation_90 0.75 0.010

+ 80 Population*Crossover*Mutation_90 1.00 0.001

- 80 Population*Crossover*Mutation_90 1.00 0.010

- 107 Population*Crossover*Mutation_100 0.75 0.001

+ 107 Population*Crossover*Mutation_100 0.75 0.010

+ 107 Population*Crossover*Mutation_100 1.00 0.001

- 107 Population*Crossover*Mutation_100 1.00 0.010

+ 184 Population*Crossover*Iteration_20 0.75 800

- 184 Population*Crossover*Iteration_20 0.75 1000

- 184 Population*Crossover*Iteration_20 1.00 800

+ 184 Population*Crossover*Iteration_20 1.00 1000

- 366 Population*Crossover*Iteration_30 0.75 800

+ 366 Population*Crossover*Iteration_30 0.75 1000

+ 366 Population*Crossover*Iteration_30 1.00 800

- 366 Population*Crossover*Iteration_30 1.00 1000

+ 1096 Population*Crossover*Iteration_40 0.75 800

- 1096 Population*Crossover*Iteration_40 0.75 1000

- 1096 Population*Crossover*Iteration_40 1.00 800

+ 1096 Population*Crossover*Iteration_40 1.00 1000

+ 4 Population*Crossover*Iteration_50 0.75 800

- 4 Population*Crossover*Iteration_50 0.75 1000

- 4 Population*Crossover*Iteration_50 1.00 800

+ 4 Population*Crossover*Iteration_50 1.00 1000

+ 709 Population*Crossover*Iteration_60 0.75 800

- 709 Population*Crossover*Iteration_60 0.75 1000
- 709 Population*Crossover*Iteration_60 1.00 800
+ 709 Population*Crossover*Iteration_60 1.00 1000
- 1382 Population*Crossover*Iteration_70 0.75 800
+ 1382 Population*Crossover*Iteration_70 0.75 1000
+ 1382 Population*Crossover*Iteration_70 1.00 800
- 1382 Population*Crossover*Iteration_70 1.00 1000
- 180 Population*Crossover*Iteration_80 0.75 800
+ 180 Population*Crossover*Iteration_80 0.75 1000
+ 180 Population*Crossover*Iteration_80 1.00 800
- 180 Population*Crossover*Iteration_80 1.00 1000
+ 750 Population*Crossover*Iteration_90 0.75 800
- 750 Population*Crossover*Iteration_90 0.75 1000
- 750 Population*Crossover*Iteration_90 1.00 800
+ 750 Population*Crossover*Iteration_90 1.00 1000
- 815 Population*Crossover*Iteration_100 0.75 800
+ 815 Population*Crossover*Iteration_100 0.75 1000
+ 815 Population*Crossover*Iteration_100 1.00 800
- 815 Population*Crossover*Iteration_100 1.00 1000
+ 106 Population*Mutation*Iteration_20 0.001 800 - 106 Population*Mutation*Iteration_20 0.001 1000 - 106 Population*Mutation*Iteration_20 0.010 800
+ 106 Population*Mutation*Iteration_20 0.010 1000 + 47 Population*Mutation*Iteration_30 0.001 800 - 47 Population*Mutation*Iteration_30 0.001 1000
- 47 Population*Mutation*Iteration_30 0.010 800 + 47 Population*Mutation*Iteration_30 0.010 1000 + 116 Population*Mutation*Iteration_40 0.001 800
- 116 Population*Mutation*Iteration_40 0.001 1000
- 116 Population*Mutation*Iteration_40 0.010 800 + 116 Population*Mutation*Iteration_40 0.010 1000 + 52 Population*Mutation*Iteration_50 0.001 800
- 52 Population*Mutation*Iteration_50 0.001 1000 - 52 Population*Mutation*Iteration_50 0.010 800 + 52 Population*Mutation*Iteration_50 0.010 1000
+ 225 Population*Mutation*Iteration_60 0.001 800 - 225 Population*Mutation*Iteration_60 0.001 1000 - 225 Population*Mutation*Iteration_60 0.010 800
+ 225 Population*Mutation*Iteration_60 0.010 1000
- 644 Population*Mutation*Iteration_70 0.001 800 + 644 Population*Mutation*Iteration_70 0.001 1000 + 644 Population*Mutation*Iteration_70 0.010 800
- 644 Population*Mutation*Iteration_70 0.010 1000
+ 329 Population*Mutation*Iteration_80 0.001 800 - 329 Population*Mutation*Iteration_80 0.001 1000 - 329 Population*Mutation*Iteration_80 0.010 800

+ 329 Population*Mutation*Iteration_80 0.010 1000

+ 681 Population*Mutation*Iteration_90 0.001 800 - 681 Population*Mutation*Iteration_90 0.001 1000 - 681 Population*Mutation*Iteration_90 0.010 800

+ 681 Population*Mutation*Iteration_90 0.010 1000

- 911 Population*Mutation*Iteration_100 0.001 800

+ 911 Population*Mutation*Iteration_100 0.001 1000

+ 911 Population*Mutation*Iteration_100 0.010 800

- 911 Population*Mutation*Iteration_100 0.010 1000

- 175 Crossover*Mutation*Iteration_0.75 0.001 800

+ 175 Crossover*Mutation*Iteration_0.75 0.001 1000

+ 175 Crossover*Mutation*Iteration_0.75 0.010 800

- 175 Crossover*Mutation*Iteration_0.75 0.010 1000

+ 175 Crossover*Mutation*Iteration_1.00 0.001 800

- 175 Crossover*Mutation*Iteration_1.00 0.001 1000

- 175 Crossover*Mutation*Iteration_1.00 0.010 800

+ 175 Crossover*Mutation*Iteration_1.00 0.010 1000

+ 393 Population*Crossover*Mutation*Iteration_20 0.75 0.001 800

- 393 Population*Crossover*Mutation*Iteration_20 0.75 0.001 1000

- 393 Population*Crossover*Mutation*Iteration_20 0.75 0.010 800

+ 393 Population*Crossover*Mutation*Iteration_20 0.75 0.010 1000

- 393 Population*Crossover*Mutation*Iteration_20 1.00 0.001 800

+ 393 Population*Crossover*Mutation*Iteration_20 1.00 0.001 1000

+ 393 Population*Crossover*Mutation*Iteration_20 1.00 0.010 800

- 393 Population*Crossover*Mutation*Iteration_20 1.00 0.010 1000

+ 404 Population*Crossover*Mutation*Iteration_30 0.75 0.001 800

- 404 Population*Crossover*Mutation*Iteration_30 0.75 0.001 1000

- 404 Population*Crossover*Mutation*Iteration_30 0.75 0.010 800

+ 404 Population*Crossover*Mutation*Iteration_30 0.75 0.010 1000

- 404 Population*Crossover*Mutation*Iteration_30 1.00 0.001 800

+ 404 Population*Crossover*Mutation*Iteration_30 1.00 0.001 1000

+ 404 Population*Crossover*Mutation*Iteration_30 1.00 0.010 800

- 404 Population*Crossover*Mutation*Iteration_30 1.00 0.010 1000

+ 428 Population*Crossover*Mutation*Iteration_40 0.75 0.001 800

- 428 Population*Crossover*Mutation*Iteration_40 0.75 0.001 1000

- 428 Population*Crossover*Mutation*Iteration_40 0.75 0.010 800

+ 428 Population*Crossover*Mutation*Iteration_40 0.75 0.010 1000

- 428 Population*Crossover*Mutation*Iteration_40 1.00 0.001 800

+ 428 Population*Crossover*Mutation*Iteration_40 1.00 0.001 1000

+ 428 Population*Crossover*Mutation*Iteration_40 1.00 0.010 800
- 428 Population*Crossover*Mutation*Iteration_40 1.00 0.010 1000
- 823 Population*Crossover*Mutation*Iteration_50 0.75 0.001 800
+ 823 Population*Crossover*Mutation*Iteration_50 0.75 0.001 1000
+ 823 Population*Crossover*Mutation*Iteration_50 0.75 0.010 800
- 823 Population*Crossover*Mutation*Iteration_50 0.75 0.010 1000
+ 823 Population*Crossover*Mutation*Iteration_50 1.00 0.001 800
- 823 Population*Crossover*Mutation*Iteration_50 1.00 0.001 1000
- 823 Population*Crossover*Mutation*Iteration_50 1.00 0.010 800
+ 823 Population*Crossover*Mutation*Iteration_50 1.00 0.010 1000
- 116 Population*Crossover*Mutation*Iteration_60 0.75 0.001 800
+ 116 Population*Crossover*Mutation*Iteration_60 0.75 0.001 1000
+ 116 Population*Crossover*Mutation*Iteration_60 0.75 0.010 800
- 116 Population*Crossover*Mutation*Iteration_60 0.75 0.010 1000
+ 116 Population*Crossover*Mutation*Iteration_60 1.00 0.001 800
- 116 Population*Crossover*Mutation*Iteration_60 1.00 0.001 1000
- 116 Population*Crossover*Mutation*Iteration_60 1.00 0.010 800
+ 116 Population*Crossover*Mutation*Iteration_60 1.00 0.010 1000
- 65 Population*Crossover*Mutation*Iteration_70 0.75 0.001 800
+ 65 Population*Crossover*Mutation*Iteration_70 0.75 0.001 1000
+ 65 Population*Crossover*Mutation*Iteration_70 0.75 0.010 800
- 65 Population*Crossover*Mutation*Iteration_70 0.75 0.010 1000
+ 65 Population*Crossover*Mutation*Iteration_70 1.00 0.001 800
- 65 Population*Crossover*Mutation*Iteration_70 1.00 0.001 1000
- 65 Population*Crossover*Mutation*Iteration_70 1.00 0.010 800
+ 65 Population*Crossover*Mutation*Iteration_70 1.00 0.010 1000
+ 910 Population*Crossover*Mutation*Iteration_80 0.75 0.001 800
- 910 Population*Crossover*Mutation*Iteration_80 0.75 0.001 1000
- 910 Population*Crossover*Mutation*Iteration_80 0.75 0.010 800
+ 910 Population*Crossover*Mutation*Iteration_80 0.75 0.010 1000
- 910 Population*Crossover*Mutation*Iteration_80 1.00 0.001 800
+ 910 Population*Crossover*Mutation*Iteration_80 1.00 0.001 1000
+ 910 Population*Crossover*Mutation*Iteration_80 1.00 0.010 800
- 910 Population*Crossover*Mutation*Iteration_80 1.00 0.010 1000
- 1172 Population*Crossover*Mutation*Iteration_90 0.75 0.001 800
+ 1172 Population*Crossover*Mutation*Iteration_90 0.75 0.001 1000
+ 1172 Population*Crossover*Mutation*Iteration_90 0.75 0.010 800
- 1172 Population*Crossover*Mutation*Iteration_90 0.75 0.010 1000

+ 1172 Population*Crossover*Mutation*Iteration_90 1.00 0.001 800

- 1172 Population*Crossover*Mutation*Iteration_90 1.00 0.001 1000

- 1172 Population*Crossover*Mutation*Iteration_90 1.00 0.010 800

+ 1172 Population*Crossover*Mutation*Iteration_90 1.00 0.010 1000

+ 42 Population*Crossover*Mutation*Iteration_100 0.75 0.001 800

- 42 Population*Crossover*Mutation*Iteration_100 0.75 0.001 1000

- 42 Population*Crossover*Mutation*Iteration_100 0.75 0.010 800

+ 42 Population*Crossover*Mutation*Iteration_100 0.75 0.010 1000

- 42 Population*Crossover*Mutation*Iteration_100 1.00 0.001 800

+ 42 Population*Crossover*Mutation*Iteration_100 1.00 0.001 1000

+ 42 Population*Crossover*Mutation*Iteration_100 1.00 0.010 800

- 42 Population*Crossover*Mutation*Iteration_100 1.00 0.010 1000