THE INFLUENCE OF THREE PARENT CROSSBREEDING ON THE DUAL POPULATION GENETIC ALGORTIHM

Esra'a Alkafaween[1,2,*], Obada Alhabashneh[2], Maram M. Al-Mjali[2]

[1]Department of Computer Science, Alzaytoonah University of Jordan, Amman, Jordan
[2]Data Science and Artificial Intelligence Department, Mutah University, Karak, Jordan

*E-mail of corresponding author: esrakafaween86@gmail.com

Esra'a Alkafaween 0000-0001-9525-1459

Resume

A genetic algorithm (GA) is an optimization technique based on natural genetics, using selection, crossover, and mutation. Crossover combines genetic material from two parents to create offspring, maintaining diversity and preventing premature convergence. While the two parents are typically used, multi-parent crossover, involving more than two parents, has shown superior results. in this paper is explored the multi-parent crossover in dual genetic algorithms, which facilitate information exchange between populations through interpolation crossbreeding. Offspring inherit traits from both parent populations, improving adaptability. The Cave-Surface GA (CSGA) with three-parent crossover is tested on 15 Travelling Salesman Problem (TSP) benchmarks. Results show that the CSGA outperforms both traditional GAs and two-parent CSGA. This method demonstrates great potential for complex optimization challenges.

Available online: https://doi.org/10.26552/com.C.2025.036

Article info

Received 23 September 2024
Accepted 17 March 2025
Online 16 April 2025

Keywords:

genetic algorithm
TSP
multipopulation
multi-parent order crossover
cave surface GA

ISSN 1335-4205 (print version)
ISSN 2585-7878 (online version)

1 Introduction

Genetic algorithms (GAs) are efficient heuristic random search strategies that are inspired by evolutionary and natural selection theories [1-2]. In the 1970s, Holland [3] investigated the fundamentals of GA. The GA examines a population of chromosomes, each representing a unique candidate solution to a certain problem. The GA includes various operators, including selection, crossover, and mutation. These operators are employed for candidate solutions to achieve improved population generation [4-6].

Among the well-known operators in genetic algorithms is crossover. The crossover process is essential for creating new chromosomes by combining two or more parent chromosomes in the hopes that the result will be an efficient new chromosome. Crossover happens after the selection of parent chromosome pairs and aids in the exchange of information between parents in order to produce offspring. During the crossover, parent chromosomes are taken in pairs and their genes are transferred in a certain order to produce offspring. These offsprings become the parent chromosomes for the following generation [7-8].

Creating the new crossover operators that fit into one of the many chromosomal representations is an important priority for many researchers. Regretfully, the majority of crossover operators' offspring do not acquire sufficient information from their parents. Currently, a crossover operator's multi-parent extension is employed to raise the quality of solutions for optimisation problems. Typically, crossover occurs between only two parents at a time, resulting in one or two offspring. Naturally, the multi-parent reproduction is not used by any species in the natural world. However, limiting the number of parents for crossover to two is not necessary in computer simulations [9].

Since the population maintains diversity, the GA are more resilient than other local search algorithms. Even now, it is still commonly seen that populations lose diversity too soon and that individuals are stuck in local optima, particularly in complex problems with many peaks in the fitness landscape, this issue is in the literature referred to as premature convergence [10-12]. Various methods have been used in numerous previous works to prevent the risk of premature convergence. These methods include: improving the genetic operators (mutation, crossover, and selection) [12-13], dynamic parameter control [14], Multipopulation GA (MPGAs) [15], a multi-objective evolutionary algorithm [16] and more.

The multi-population idea divides the population into subpopulations, each of which is more likely to follow a different search path. The migration process exchanges good individuals between subpopulations, while the crossover operator creates new individual. The migration rate is the number of individuals that must be replaced between subpopulations, and it allows for control over the degree of diversity within the subpopulation. Migration interval, which affects the number of times migration occurs, is another element that encourages subpopulation variety [17] and [18].

Dual-population GA (DPGA) is a form of MPGA in which an additional population acts as a reservoir of diversity. The main population is comparable to that of a traditional GA and evolves to find effective solutions. The reserve population evolves to support and diversify the main population. In contrast to MPGAs, which use migration to communicate information between populations, the DPGAs rely on crossbreeding due to their distinct fitness functions [19] and [20].

Crossbreeding is used by dual populations to share information amongst populations. Crossbreeding refers to a recombination of an individual from the main population and an individual from the reserve population. Since the offspring of this crossbreeding contain genetic material from both populations, their fitness values are generally high in either population, and they can serve as a means of information sharing.

The TSP is an NP-hard problem in combinatorial optimization, as well as an old and challenging combinatorial mathematics topic. Enumeration makes it simple to find the shortest path between some cities [21-22]. If n is actually large, there are (n-1)! potential combinations, and the searching space of the routes will show a pattern of explosive growth. Under these conditions, one cannot locate the ideal path using the conventional searching strategy. As a result, several new types of optimization computation methods emerge to get the TSP's optimal solution; among which, the GA receives the most favor of the public and becomes one of the most effective techniques to solve the TSP, because of its wide applicability and the character that does not require to get additional insight into the problems and depends less on the specific fields of the problem [23].

The GA is utilized not just to solve conventional TSPs, but in a variety of other applications, as well, including transportation planning [24], location-based services [25] and urban design [26], etc.

In this paper, the contribution is to the topic of enhancing dual population GA performance employing three parents during population crossbreeding, in an attempt to promote gene variety and introduce significant features, as well as to reduce the premature convergence and so improve the performance of genetic algorithms.

2 Related work

The GAs primarily use two parents for crossover operations, which corresponds to the natural behaviour of evolution, in which individuals adopt only the two parents to generate an offspring. However, studies have shown that multi-parent crossover is more effective and more successful than the two-parent crossover [27-28].

One possible strategy to increase the GA performance is to add new features into the GA, i.e., features that do not fit within the existing GA paradigm, some recent efforts employing the multi-parent recombination operators. An attempt to maintain the basic GA paradigm while improving the GA performance by permitting multi-parent reproduction [29]. Multi-parent crossover can be seen of as a generalisation of conventional two parent crossover in terms of the number of parents. There have been many suggestions for the multi-parent crossover operators. In general, having more parents results in a more thorough survey to identify the genes of the offspring and increases the likelihood of either exploitation, exploration, or both [30].

Several multi-parent crossover methods have been developed for genetic algorithms such as scanning crossover [27], diagonal crossover [27], center of mass crossover, multi-parent feature-wise crossover, and seed crossover [31], simplex crossover [32]. Studies show that the multi-parent crossover is more effective than the two parent crossover. The performance of these multi-parent crossovers is generally studied using numerical optimization problems [33].

To solve combinatorial optimization problems, only a limited number of multi-parent crossovers have been used, including adjacency-based crossover (ABC) [27], multi-parent extension of partially mapped crossover (MPPMX) [33], multi-parent sequential constructive crossover (MPSCX) [9], extended precedence preservative crossover [34] and a multi-parent order crossover (MPOX) [35].

The ABC is suitable for order-based representations, such as TSP, where value positioning is crucial. This crossover uses a marker update approach to pick genes from all parents and produce viable offspring, while this crossover resulted in a viable TSP child.

The MPPMX [33] extends partially mapped crossover to address the multi-parent crossover. The proposed crossover changed the mapping list and legalisation method to include more parents in the partially mapped crossover.

The multi-parent sequential constructive crossover was developed to address the travelling salesman and job shop scheduling problem. Experiments were conducted to assess MPSCX performance with varying parent numbers and mutation probability. Experimental findings on TSPLIB instances demonstrate that MPSCX considerably increases solution quality.

The MPOX [35] extends the Order Crossover (OX) [36; this extension's primary goal is to generate an acceptable solution from more than two parents that can handle various combinatorial challenges with efficacy. The berth allocation problem (BAP) and TSP have been used to analyse the original OX and MPOX. The outcomes demonstrate that MPOX outperforms the two-parent OX and generates competitive performance for both benchmarks.

3 Experimental settings and result

In the literature, the multiple parent crossovers have successfully demonstrated the power of using more parents. The effectiveness of these operations has been examined experimentally. The performance of MPOX has also been tested on travelling salesman problems, which have been widely studied in the literature. To examine the effectiveness of the multi parent crossover and its effect on the MPGA, and specifically on dual population, the multi-parent order crossover (MPOX) [35] was chosen as a crossbreeding operator between dual population, Specifically in CSGA algorithm.

In this paper, the CSGA algorithm are chosen [37], which uses two populations and is also a type of MPGA. The CSGA method is inspired by the genetic diversity found in Mexican cavefish and is a variation of the Dual Population GA. Through the inter-population crossbreeding, the CSGA enhances variety via the secondary population (cave population) and enables information transmission between populations, effectively preventing the premature convergence.

The CSGA starts with two randomly generated populations: Cave and Surface. Individuals in each population are evaluated with the same fitness functions. The Cave population evolves by a combination of inbreeding within the population and crossbreeding with individuals from the Surface population. Additionally, the surface population evolves predominantly by inbreeding among parents from the same population.

A conventional GA selects two parent chromosomes for a cross-over procedure, yielding two offspring. The CSGA, on the other hand, has two more factors beyond GAs: the crossbreeding interval (CI) and the crossbreeding rate. The crossbreeding interval (CI) is the number of generations between each crossbreeding, whereas the cross-breeding rate (CR) is the number of individuals picked from each group during the cross-breeding. These factors affect both accuracy and computation time. CSGA generates two offsprings through each cross-over operator between the two parents by randomly selecting a number of parents from the two populations for recombination based on the crossbreeding rate. Following that, one of the children is chosen to join the Cave population's next generation through selection for local survival, while the other is transported to the Surface population. In addition to being repeated for each of the two prospective parents, this process is only carried out through particular generations, depending on the CR rate. The pseudocode for CSGA is displayed in Algorithm 1 [37] (see Figure 1).

The MPOX [35] generalizes the Order Crossover (OX) to a multi-parent crossover. To cut each parent into substrings for an n-parent MPOX, n-1 crossing points generated randomly. The second chosen substring, which is between the first and second crossover points, is then duplicated into the newly created offspring at the same absolute position in the following step.

To create a viable offspring from multiple parents, the MPOX begins at the second crossover point of the second parent. It then selects the elements that are not already present in the offspring to fill in the gaps, starting from this crossover point (the third substring).

Algorithm 1: CSGA Algorithm

Begin
step1: initialize input parameters of problems: crossover rate, mutation rate, crossbreeding rate (cr), crossbreeding interval (ci), max generation.
step2: initialize two subpopulations, cave population (cp) and surface population (sp).
step3: For each subpopulation, repeat the following steps until the termination criterion is met.
step4: Calculate fitness value;
step5: inbreeding:
a. Selection
b. Crossover
c. Mutation
step6: crossbreeding (based on ci and cr do):
a. choose individual from cp and choose individual from sp.
b. Crossover
c. Move offspring one to
cp and the second to sp.
step8: output the final best solution.
End.

***Figure 1 The** CSGA Algorithm [37]*

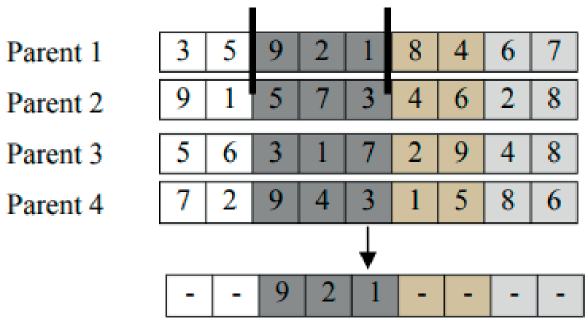

Fig. 4. Substring copy.

Figure 2 *MPOX with four parents [36]*

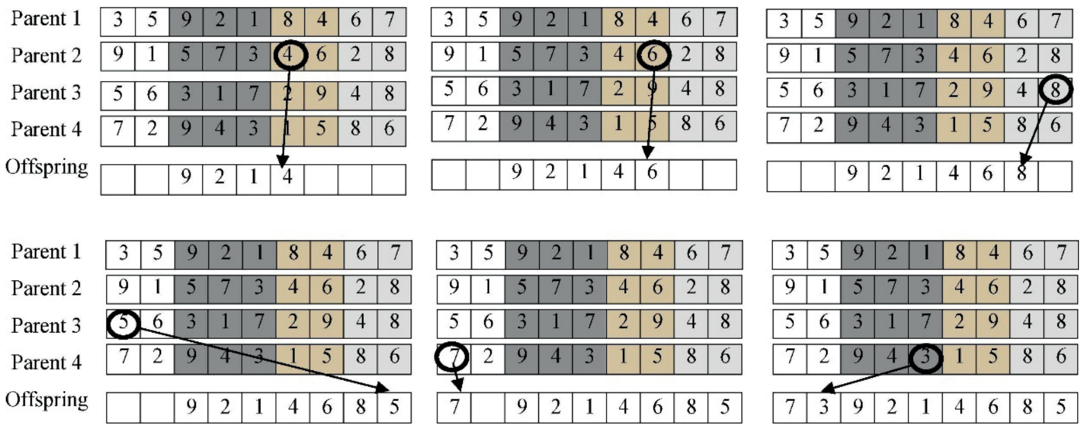

Figure 3 *An example of MPOX-based offspring production [36]*

This process continues until the third substring of the offspring is fully populated.

Next, the MPOX moves to the fourth crossover point of the next parent, again choosing elements not included in the offspring and filling them in from that crossover point. This continues until the next crossover point or the end of the offspring is reached. Once all the parents have been processed, the MPOX starts over with the first parent, selecting any missing elements and copying them into the offspring from the beginning up to the second crossover point. This ensures the offspring is fully completed with all the necessary elements (see Figure 2 and Figure 3).

Going back to step 6 in the CSGA in Algorithm 1, MPOX crossover was used as a crossbreeding operator between the two populations., where the two offspring are selected from the first population and one offspring from the second population. Two of the resulting individuals were sent to the first population, and one was sent to the second population. (see Figure 4). In this way, the population was aimed to be more diverse and thus reduce the premature convergence.

To investigate the effect of utilizing three parents in the crossbreeding of the CSGA algorithm, investigations were conducted in 15 instances from the TSPLIB [38]. The experiment was carried out ten times for each instance.

Table 1 displays the GA parameters that were selected to perform the experiments. Since the primary objective of the study was to validate the effectiveness of the CSGA with 3 parent crossover, the genetic algorithm and CSGA regardless of the parameters used, No complex parameter control procedures were utilized. The GA's simple and standard parameters were used.

The fitness level of each individual was ascertained by a truncated selection process. Truncation selection is the simplest selection technique, and this is a common way to allocate the fitness function to each chromosome in the GA population. This kind of selection involves sorting the population based on fitness and then eliminating the proportion of people who are less fit [39].

Since one-point modified crossover and exchange mutation are two of the most straightforward approaches that have been applied to situations that are classified as permutation problems, they were employed for the reproduction process in our research [22]. One-point modified crossover creates the offspring by using the single point fragmentation of the parents and then combining the parents at the crossover point. One-point crossover chooses two parents for crossover and then chooses any crossover point at random. The parents are then combined at the crossover point to create two offspring. The frequency with which two chromosomes exchange some of their parts during a single generation is known as the crossover rate; crossover rate is in the range of [0,1] [7].

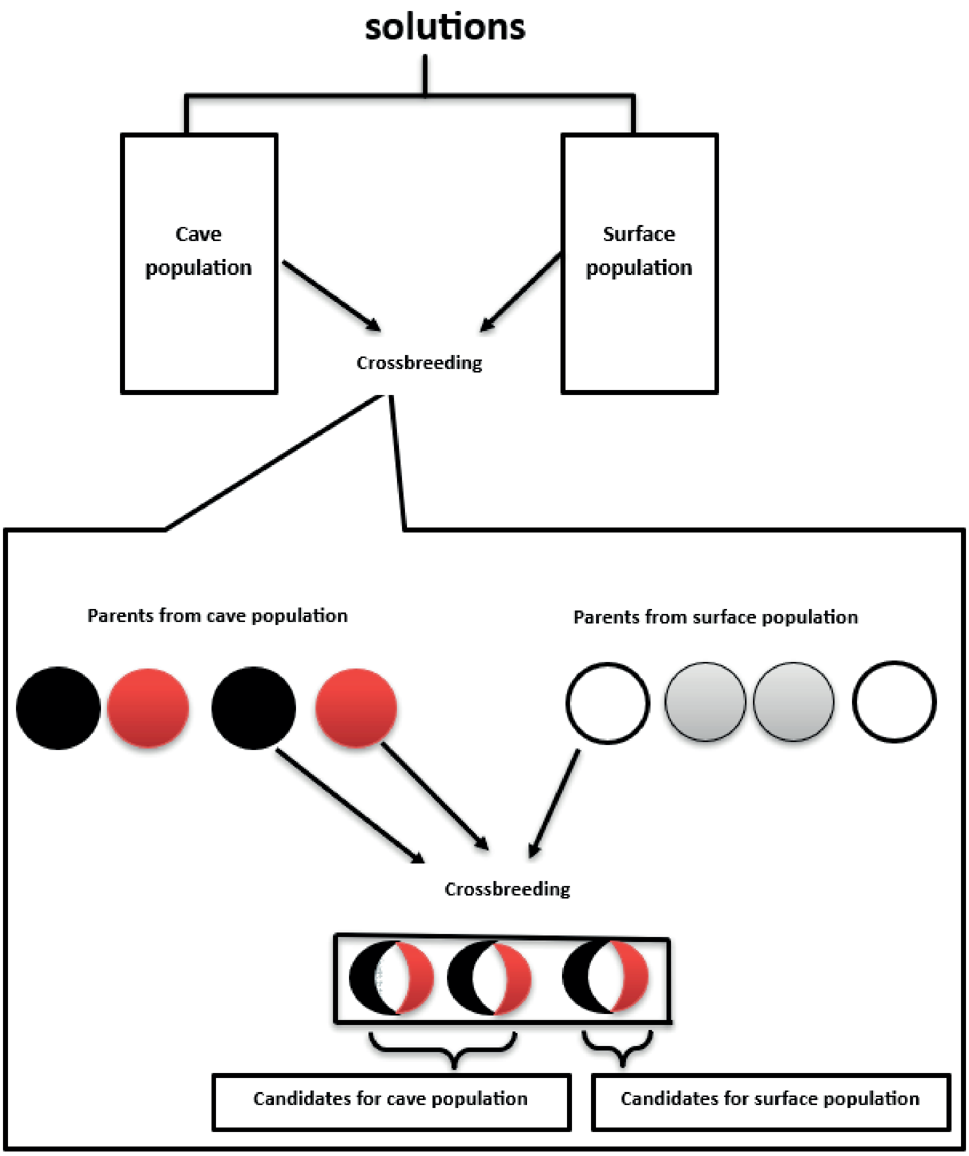

Figure 4 *CSGA method with 3 parents crossbreeding*

Table 1 *GA configuration*

Parameter	Value
Population size	200
Generation limit	3000
Initialization method	Random
Crossover	One-point modified
Crossover rate	0.85
Mutation	Exchange
Mutation rate	0.08
Selection	Truncation Selection
Crossbreeding Rate	5
Crossbreeding Interval	7
Termination criteria	Generation limit
Crossover type used in Crossbreeding	MPOX

The exchange mutation randomly selects two genes and switches their locations. The mutation rate, which ranges from 0 to 1, specifies the number of chromosomes to be modified in a single generation [8].

Our GA employed the reinsertion method, which is an expansion sampling technique, in each experiment [40]. This technique ensures that only the best half of the population - from both the new and old generations - is chosen for the following generation. The old generation competes with the new individuals when a new generation is created.

The genetic algorithm's basic parameters were purposefully and manually selected. Neither parameter tuning, nor intricate control processes, were used. Natural ratios for the crossover and mutation process were the crossover and mutation ratios. In addition, the population size was manual, as well as the number of generations, and this was fixed for all methods for fair comparison. Again, these parameters are standard, and are used in many researches. This strategy is consistent with our paper's primary objective, which is to confirm the efficacy and emphasize the advantages of employing

Table 2 The *TSP instance results obtained through 3000 generations using GA and CSGA algorithms*

Optimal Solution	Instance	GA		CSGA		CSGA with 3 Parent Crossbreeding	
		Min	Average	Min	Average	Min	Average
2579	a280	6952	7587.1	5914‡	6541.2‡	6317	6575.7
10628	att48	35843	41766.4	35704	40468.3	35873	39482‡
7542	berlin52	8253‡	9123.1‡	8497	9572.5	8599	9247
118282	bier127	152453	170944.7	146855‡	161837.4	149325	157590.6‡
6110	ch130	8865	10000.7	8768	9777.2	8442‡	9335.8‡
6528	ch150	10114	10914.667	9965	10906	9455‡	10314.6‡
426	eil51	465	478.3‡	476	502.5	455‡	493
21282	kroA100	27555	32230.6	27175	31655.4	26256‡	28069.9‡
14379	lin105	20153	24129.1	20006	23199.1	17778‡	21983.8‡
108159	pr76	134438	133195.82‡	130101	143111.4	129849‡	144835.2
58537	pr144	112926‡	119005.8‡	117911	134050.4	117026	128547
42029	lin318	125686	139947.1	112936‡	119163.4‡	117471	126146.1
202339	ali535	9484	10249.5	8418‡	8827.6‡	8713	9429.333
8806	rat783	51871	53879.1	46348‡	47547‡	49308	52293.4
29437	kroB200	58704	67404.5	57500	61319.4	51832‡	59243.8‡

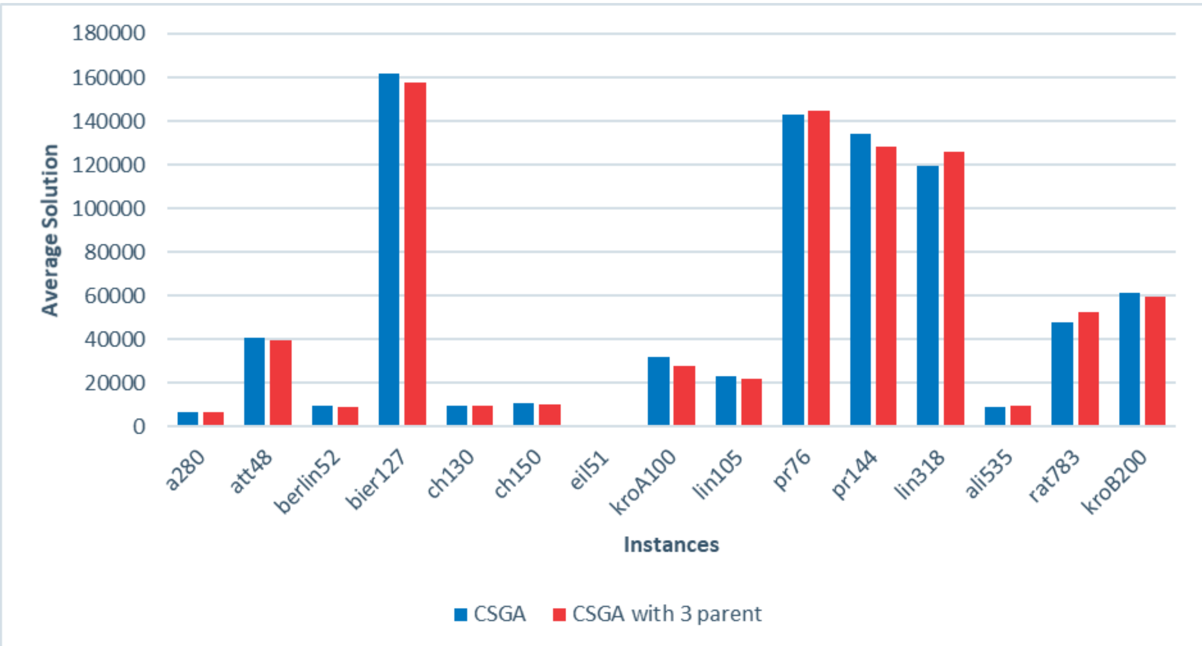

Figure 5 *Average Convergence of each Algorithm in 15 instances*

three parents throughout the crossbreeding process in comparison to the genetic algorithm in terms of diversity.

Table 2 summarizes the results of the three-parent crossover on TSP instances.

As demonstrated in Table 2, in 7 out of 15 cases, the CSGA with 3 parent crossover outperformed the CSGA and GAs. The CSCA with three-parent crossover also had the lowest costs in nine out of the fifteen cities, as can be seen in the table's Min column.

The Average convergence of each algorithm to a minimum value is shown in Figure 5. Once more, the CSGA with three parent crossover performs better in terms of convergence to a minimum value on KroA100 and ch130 than both CSGA and GA. The better convergence becomes possible by the population diversity that the CSGA with three parent crossover provides as shown in Figure 6 and 7.

The contribution of using a crossover with three parent is observed in the attempt to find diversity in the dual population., despite not using any complex parameters, and the stability of the parameters for the two methods, the importance of using the crossover with 3 parent is shown, and what also contributed to the diversity is sending the resulting individuals to the first population, which led to the diversity in the population by allowing the extraction of characteristics from the parents, and thus the resulting offspring possesses good characteristics from the three parents.

As can be seen from Table 3 and Figure 8, GA is the fastest algorithm in all the cases, showing that it requires the least computation. The CSGA takes more time than GA and CSGA with 3 Parent Crossbreeding takes the longest execution time. For small instances, GA might be sufficient. For large instances, CSGA or CSGA with 3 Parent Crossbreeding is preferable.

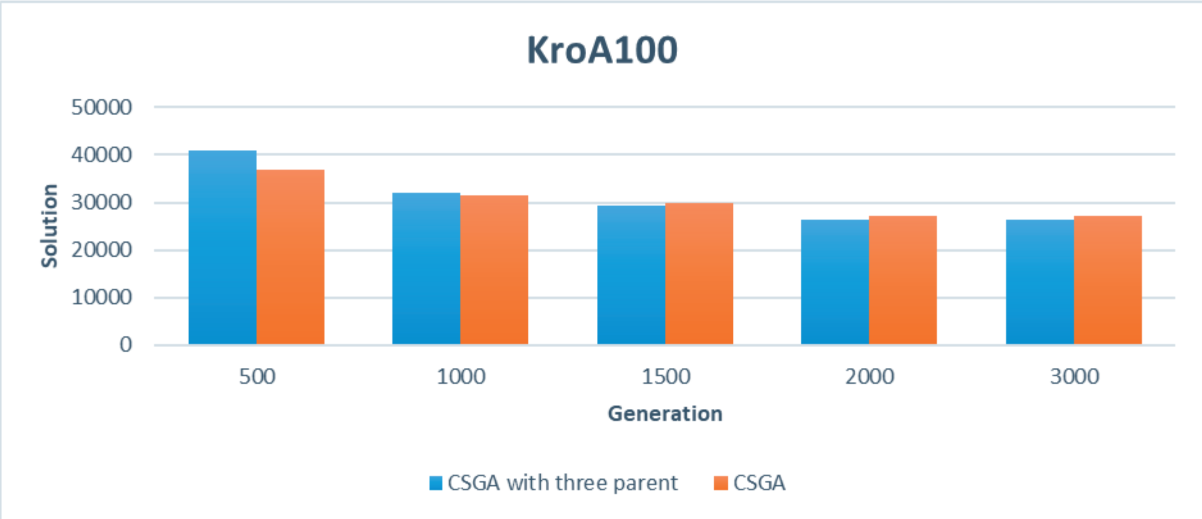

Figure 6 *The KroA100 Convergence of Min value*

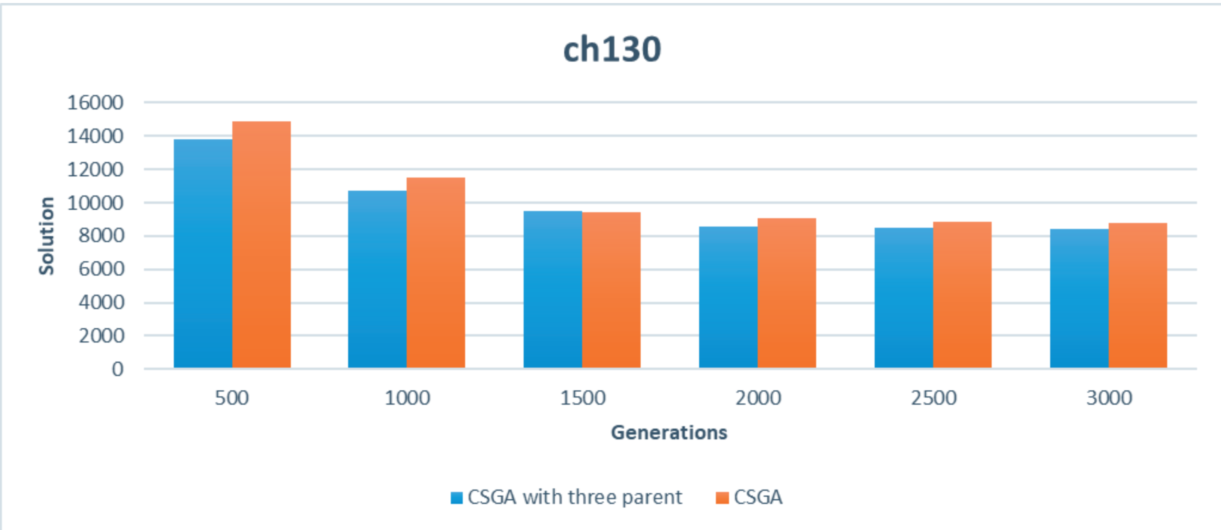

Figure 7 *The ch130 Convergence of min value*

As Table 4 illustrates, the GA is the fastest but least effective in finding optimal solutions. The CSGA improves both the minimum and average solution values, while maintaining a reasonable execution time. The CSGA with 3 Parent Crossbreeding achieves the best solutions, but takes the longest time to execute.

CSGA with 3 Parent Crossbreeding is only suitable if solution quality is more important than speed.

Table 3 *Time consumed by each method after 3000 generations*

Instance	Time, ms		
	GA	CSGA	CSGA with 3 Parent Crossbreeding
a280	48083	65754‡	108230
att48	44117	55874	52664‡
berlin52	39001	58008	54515‡
bier127	46659	73485‡	75473
ch130	44808	87922	70094‡
ch150	72956	69967‡	76813
eil51	52743	37872	37174‡
kroA100	52028	45897‡	57707
lin105	51643	53942‡	58133
pr76	30063	38676‡	40638
pr144	50496	63113‡	65438
lin318	71389	86459‡	106096
ali535	136794	294048	272006‡
rat783	96934	618016	427801‡
kroB200	105338	147291	143102‡

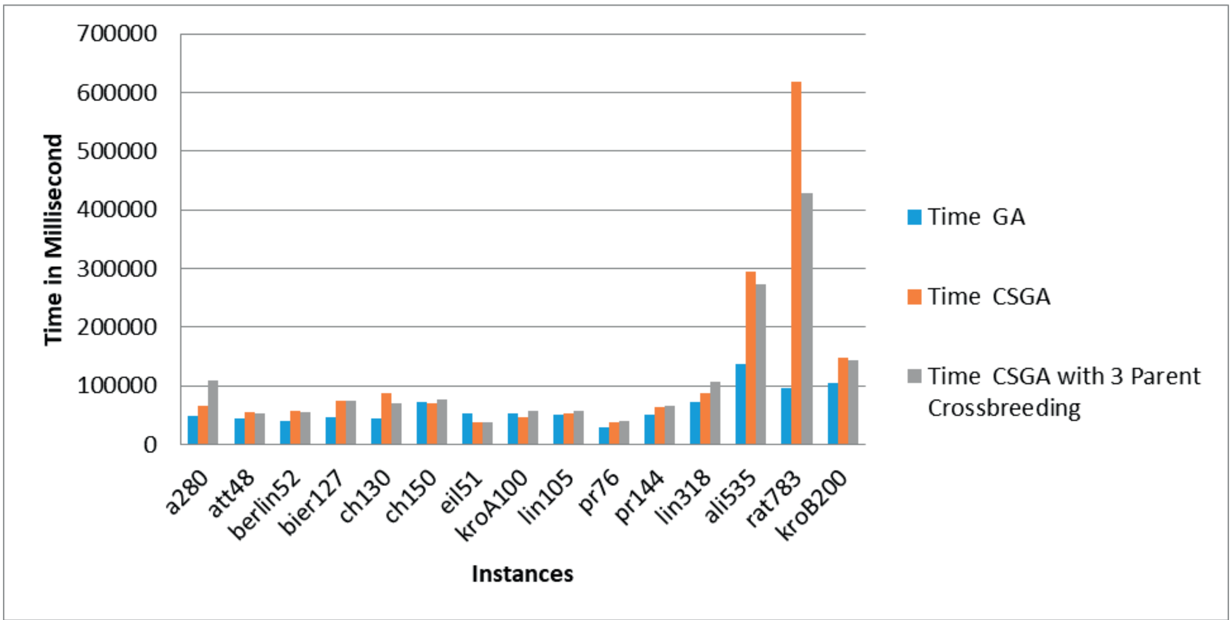

Figure 8 *Time Consumed*

Table 4 *Trade-off between execution time and solution quality*

Algorithm	Solution Quality	Execution Time
GA	Worst	Fastest
CSGA	Good	Moderate
CSGA with 3 Parent Crossbreeding	Best	Slows

4 Conclusion

The multi-parent crossover operator is one of the mechanisms used in evolutionary algorithms to improve the population diversity. Multi-parent crossovers have shown their superiority over the classic two-parent crossovers in several problems as shown in the literature. In this paper, the performance of three parent crossover methods on MPGA was investigated., and more specifically on the crossbreeding process on CSGA. To tackle combinatorial optimisation problems, the MPOX crossover was selected, which is an extension of OX. This extension's primary goal is to generate a workable solution from more than two parents that can be applied to various combinatorial problems effectively. This assessment was conducted on the TSP problem, and a comparison was made between the traditional GA, which uses only two parents in the crossover process, the CSGA algorithm, which uses two parents in the crossbreeding between populations and the CSGA algorithm, which uses three parents in the crossbreeding between populations. According to the results, three parent crossover in crossbreeding typically yields competitive outcomes in terms of solution quality. Finally, the three-parent crossover in crossbreeding with MPGA resolved combinatorial optimisation issues (e.g., CSGA) and yield beneficial results. However, it should be noted that a number of factors influence the genetic algorithm's performance. In future studies, the inbreeding process will be carried out using multi-parents and also the plan is to examine the effect of the number of parents in the multiparent crossover, and how it affects the inbreeding and crossbreeding process. Comprehensive tests and comparisons will be conducted in the next study., with the aim of providing a detailed assessment of the effectiveness of multiparent crossover in the multipopulation approach.

Acknowledgement

The authors received no financial support for the research, authorship and/or publication of this article.

Conflicts of interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] LIU, X., DENG, Y. A new QoS-aware service discovery technique in the internet of things using whale optimization and genetic algorithms. *Journal of Engineering and Applied Science.* 2024, **71**(1), 4. ISSN 1110-1903, eISSN 2536-9512. Available from: https://doi.org/10.1186/s44147-023-00334-1

[2] ALKAFAWEEN, E., HASSANAT, A. B., TARAWNEH, S. Improving initial population for genetic algorithm using the multi linear regression based technique (MLRBT). *Communications - Scientific Letters of the University of Zilina* [online]. 2021, **23**(1), p. E1-E10. ISSN 1335-4205, eISSN 2585-7878. Available from: https://doi.org/10.26552/com.C.2021.1.E1-E10

[3] HOLLAND, J. H. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence* [online]. MIT press, 1992. ISBN 9780262275552. Available from: https://doi.org/10.7551/mitpress/1090.001.0001

[4] BALA, A., SHARMA, A. K. A comparative study of modified crossover operators. In: 2015 3rd International Conference on Image Information Processing ICIIP: proceedings [online]. IEEE. 2015. ISBN 978-1-5090-0148-4. Available from: https://doi.org/10.1109/ICIIP.2015.7414781

[5] ALKAFAWEEN, E. O. Novel methods for enhancing the performance of genetic algorithms [online]. Master's thesis. Jordan: Mutah University, 2015. Available from: https://doi.org/10.48550/arXiv.1801.02827

[6] ALKAFAWEEN, E., HASSANAT, A. B. Improving TSP solutions using GA with a new hybrid mutation based on knowledge and randomness. *Communications - Scientific Letters of the University of Zilina* [online]. 2020, **22**(3), p. 128-139. ISSN 1335-4205, eISSN 2585-7878. Available from: https://doi.org/10.26552/com.C.2020.3.128-139

[7] HASSANAT, A. B., ALKAFAWEEN, E. On enhancing genetic algorithms using new crossovers. *International Journal of Computer Applications in Technology* [online]. 2017, **55**(3), p. 202-212. ISSN 0952-8091, eISSN 1741-5047. Available from: https://doi.org/10.1504/IJCAT.2017.10005868

[8] HASSANAT, A. B., ALKAFAWEEN, E., AL-NAWAISEH, N. A., ABBADI, M. A., ALKASASSBEH, M., ALHASANAT, M. B. Enhancing genetic algorithms using multi mutations: experimental results on the travelling salesman problem. *International Journal of Computer Science and Information Security*. 2016, **14**(7), 785. eISSN 1947-5500.

[9] AHMED, Z. H. Multi-parent extension of sequential constructive crossover for the travelling salesman problem. *International Journal of Operational Research* [online]. 2011, **11**(3), p. 331-342. ISSN 1745-7645, eISSN 1745-7653. Available from: https://doi.org/10.1504/IJOR.2011.041347

[10] LEUNG, Y., GAO Y., XU, Z.-B. Degree of population diversity - a perspective on premature convergence in genetic algorithms and its Markov chain analysis. *IEEE Transactions on Neural Networks* [online]. 1997, **8**(5), p. 1165-1176. ISSN 1045-9227. Available from: https://doi.org/10.1109/72.62321

[11] LIU, L., FEI, T., ZHU, Z., WU, K., ZHANG, Y. A survey of evolutionary algorithms. In: 2023 4th International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering ICBAIE: proceedings [online]. IEEE. 2023. eISBN 979-8-3503-4361-8, p. 22-27. Available from: https://doi.org/10.1109/ICBAIE59714.2023.10281260

[12] HASSANAT, A., ALMOHAMMADI, K., ALKAFAWEEN, E., ABUNAWAS, E., HAMMOURI, A., PRASATH, V. S. Choosing mutation and crossover ratios for genetic algorithms - a review with a new dynamic approach. *Information* [online]. 2019, **10**(12), 390. eISSN 2078-2489. Available from: https://doi.org/10.3390/info10120390

[13] XUE, Y., ZHU, H., LIANG, J., SLOWIK, A. Adaptive crossover operator based multi-objective binary genetic algorithm for feature selection in classification. *Knowledge - Based Systems* [online]. 2021, **227**, 107218. ISSN 0950-7051, eISSN 1872-7409. Available from: https://doi.org/10.1016/j.knosys.2021.107218

[14] JEBARI, K., BOUROUMI, A., ETTOUHAMI, A. Parameters control in GAs for dynamic optimization. *International Journal of Computational Intelligence Systems* [online]. 2013, **6**(1), p. 47-63. eISSN 1875-6883. Available from: https://doi.org/10.1080/18756891.2013.754172

[15] WHITLEY, D., RANA, S., HECKENDORN, R. B. The Island model genetic algorithm: on separability, population size and convergence. *Journal of Computing and Information Technology*. 1999, **7**, p. 33-48. ISSN 1330-1136, eISSN 1846-3908.

[16] DEB, K. Multi-objective evolutionary algorithms. In: *Springer handbook of computational intelligence* [online]. KACPRZYK, J., PEDRYCZ, W. (Eds.). Berlin Heidelberg: Springer-Verlag, 2015. p. 995-1015. ISBN 978-3-662-43504-5, eISBN 978-3-662-43505-2. Available from: https://doi.org/10.1007/978-3-662-43505-2

[17] DENG, D.-S., LONG, W., LI, Y.-Y., SHI, X.-Q. Multipopulation genetic algorithms with different interaction structures to solve flexible job-shop scheduling problems: a network science perspective. *Mathematical Problems in Engineering* [online]. 2020, **2020**(1), 8503454. ISSN 1024-123X, eISSN 1563-5147. Available from: https://doi.org/10.1155/2020/8503454

[18] SHI, X., LONG, W., LI, Y., DENG, D. Multi-population genetic algorithm with ER network for solving flexible job shop scheduling problems. *PloS One* [online]. 2020, **15**(5), e0233759. eISSN 1932-6203. Available from: https://doi.org/10.1371/journal.pone.0233759

[19] PARK, T., RYU, K. R. A dual population genetic algorithm with evolving diversity. In: 2007 IEEE Congress on Evolutionary Computation: proceedings [online]. IEEE. 2007. ISSN 1089-778X, eISSN 1941-0026, ISBN 978-1-4244-1339-3. Available from: https://doi.org/10.1109/CEC.2007.4424928

[20] YANG, S. PDGA: the primal - dual genetic algorithm. In: *Design and application of hybrid intelligent systems* [online]. ABRAHAM, A., KOPPEN, M., FRANKE, K. (Eds.). Amsterdam: IOS Press, 2003. ISBN 978-1586033941, p. 214-223. Available from: https://doi.org/10.13140/RG.2.1.3134.3445

[21] ALKAFAWEEN, E., ELMOUGY, S., ESSA, E., MNASRI, S., TARAWNEH, A. S., HASSANAT, A. IAM-TSP: iterative approximate methods for solving the travelling salesman problem. *International Journal of Advanced Computer Science and Applications* [online]. 2023, **14**(11), p. 420-428. ISSN 2158-107X, eISSN 2156-5570. Available from: https://doi.org/10.14569/IJACSA.2023.0141143

[22] ALKAFAWEEN, E., HASSANAT, A., ESSA, E., ELMOUGY, S. An efficiency boost for genetic algorithms: initializing the GA with the iterative approximate method for optimizing the traveling salesman problem - experimental insights. *Applied Sciences* [online]. 2024, **14**(8), 3151. eISSN 2076-3417. Available from: https://doi.org/10.3390/app14083151

[23] YU, Y., CHEN, Y., LI, T. A new design of genetic algorithm for solving TSP. In: 2011 4th International Joint Conference on Computational Sciences and Optimization: proceedings [online]. IEEE. 2011. ISBN 978-1-4244-9712-6. Available from: https://doi.org/10.1109/CSO.2011.46

[24] KALEYBAR, H. J., DAVOODI, M., BRENNA M, ZANINELLI, D. Applications of genetic algorithm and its variants in rail vehicle systems: a bibliometric analysis and comprehensive review. *IEEE Access* [online]. 2023, **11**, p. 68972-68993. eISSN 2169-3536. Available from: https://doi.org/10.1109/ACCESS.2023.3292790

[25] AMIRI, F. Optimization of facility location-allocation model for base transceiver station antenna establishment based on genetic algorithm considering network effectiveness criteria (case study north of Kermanshah). *Scientia Iranica* [online]. 2023, **30**(5), p. 1841-1854. ISSN 1026-3098, eISSN 2345-3605. Available from: https://doi.org/10.24200/sci.2021.55207.4116

[26] FUENTES-MARILES, O. A., GRACIA-SANCHEZ, J., CHOMPA-ABARCA, J. A., DE, F. Predesign of an urban rainfall drainage network with genetic algorithms. *Journal of Building Technology* [online]. 2024, **6**(2), p. 2717-5103. ISSN 2705-1390, eISSN 2717-5103. Available from: https://doi.org/10.32629/jbt.v6i2.2475

[27] EIBEN, A. E., RAUE, P.-E., RUTTKAY, Z. Genetic algorithms with multi-parent recombination. In: International Conference on Parallel Problem Solving from Nature: proceedings [online]. Springer. 1994, p. 78-87. Available from: https://doi.org/10.1007/3-540-58484-6_252

[28] ROYCHOWDHURY, S., ALLEN, T. T., ALLEN, N. B. A genetic algorithm with an earliest due date encoding for scheduling automotive stamping operations. *Computers and Industrial Engineering* [online]. 2017, **105**, p. 201-209. ISSN 0360-8352, eISSN 1879-0550. Available from: https://doi.org/10.1016/j.cie.2017.01.007

[29] EIBEN, A. E., VAN KEMENADE, C. H. Performance of multi-parent crossover operators on numerical function optimization problems. 1995.

[30] TING, C.-K., CHEN, C.-C. The effects of supermajority on multi-parent crossover. 2007.

[31] TSUTSUI, S., GHOSH, A. A study on the effect of multi-parent recombination in real coded genetic algorithms. In: 1998 IEEE International Conference on Evolutionary Computation and IEEE World Congress on Computational Intelligence (Cat. No. 98TH8360): proceedings [online]. IEEE. 1998. ISBN 0-7803-4869-9. Available from: https://doi.org/10.1109/ICEC.1998.700159

[32] TSUTSUI, S., JAIN, L. C. On the effect of multi-parents recombination in binary coded genetic algorithms. In: 1998 2nd International Conference. Knowledge-Based Intelligent Electronic Systems. Proceedings KES'98 (Cat. No. 98EX111): proceedings [online]. IEEE. 1998. ISBN 0-7803-4316-6. Available from: https://doi.org/ 10.1109/KES.1998.725966

[33] TING, C.-K., SU, C.-H., LEE, C.-N. Multi-parent extension of partially mapped crossover for combinatorial optimization problems. *Expert Systems with Applications* [online]. 2010, **37**(3), p. 1879-1886. ISSN 0957-4174, eISSN 1873-6793. Available from: https://doi.org/10.1016/j.eswa.2009.07.082

[34] SIN, O. C., MOIN, N. H., OMAR, M. Multi parents extended precedence preservative crossover for job shop scheduling problems. *Malaysian Journal of Computer Science* [online]. 2013, **26**(3), p. 170-181. ISSN 0127-9084. Available from: https://ejournal.um.edu.my/index.php/MJCS/article/view/6769

[35] ARRAM, A., AYOB, M. A novel multi-parent order crossover in genetic algorithm for combinatorial optimization problems. *Computers and Industrial Engineering* [online]. 2019, **133**, p. 267-274. ISSN 0360-8352, eISSN 1879-0550. Available from: https://doi.org/10.1016/j.cie.2019.05.012

[36] DAVIS, L. Applying adaptive algorithms to epistatic domains. In: Joint International Conference on Artificial Intelligence IJCAI: proceedings. 1985. p. 162-164.

[37] ALKAFAWEEN, E., HASSANAT, A., ESSA, E., ELMOUGY, S. CSGA: a dual population genetic algorithm based on Mexican cavefish genetic diversity. *Jordanian Journal of Computers and Information Technology* [online]. 2024, **10**(3), p. 1-16. ISSN 2413-9351, eISSN 2415-1076. Available from: https://doi.org/10.5455/jjcit.71-1707664207

[38] REINELT, G. TSBLIB [online] [accessed 2024-09-01]. University of Heidelberg, 1996. Available from: http://comopt.ifi.uni-heidelberg.de/

[39] JEBARI, K., MADIAFI, M. Selection methods for genetic algorithms. *International Journal of Emerging Sciences*. 2013, **3**(4), p. 333-344. ISSN 2222-4254.

[41] DONG, M., WU, Y. Dynamic crossover and mutation genetic algorithm based on expansion sampling. In: *Artificial intelligence and computational intelligence. AICI 2009. Lecture notes in computer science. Vol. 5855* [online]. DENG, H., WANG, L., WANG, F. L., LEI, J. (Eds.). Berlin, Heidelber: Springer, 2009. ISBN 978-3-642-05252-1, eISBN 978-3-642-05253-8. Available from: https://doi.org/10.1007/978-3-642-05253-8_16