International Journal of Informatics and Computation (IJICOM)
Vol.7, No.2, 2025
ISSN: 2685-8711, E-ISSN: 2714-5263 DOI: 10.35842/ijicom

Comparative Study of Genetic Algorithm for Solving Teacher Placement Problem

Haris Sriwindono[1]

Abstract
Teacher placement is a combinatorial optimization problem in educational management that requires simultaneously satisfying teacher qualifications, school requirements, geographical constraints, and personal preferences. As the scale of educational systems grows, manual assignment becomes impractical, and the problem's NP-hard nature necessitates efficient computational approaches. This study applies a Genetic Algorithm (GA) framework to evaluate the effectiveness of four crossover operators, including Single-Point Crossover (SPX), Two-Point Crossover (TPX), Cycle Crossover (CX), and Ordered Crossover (OX) for solving the teacher placement problem. The GA uses permutation encoding, roulette wheel selection, and partial shuffle mutation, and operates on real-world data from the Magelang Regency Education Office, comprising 636 teachers and 106 schools. The objective is to minimize the total commuting distance between teachers' residences and assigned schools under varying mutation-to-crossover probability ratios (1:20 to 1:100). Experimental results show that OX consistently produces the best solutions, achieving the lowest average fitness value (10,301.63) across all configurations, followed closely by CX. In contrast, SPX and TPX demonstrate performance degradation at higher crossover probabilities, likely due to their inability to preserve valid permutations. Statistical analysis, including ANOVA and Kruskal–Wallis tests, confirms significant differences in performance, reinforcing the superiority of permutation-preserving crossovers. These results provide actionable guidance for designing intelligent teacher placement systems and selecting optimal GA operators for complex, real-world allocation problems.

Keywords:
Genetic Algorithm, Crossover, Partial Shuffle Mutation, Roulette Wheel Selection

1. Introduction

Teacher placement represents one of the most strategic yet complex challenges in modern educational resource management. The effectiveness of teacher deployment not only influences the quality of teaching and learning but also directly impacts the operational efficiency of educational institutions [1]. This process requires decision-makers to assign teachers to schools in a way that optimally matches their subject expertise, pedagogical competencies, and personal preferences with the specific needs of institutions, all while satisfying geographical, regulatory, and policy constraints [2]. As educational systems expand, the sheer number of teachers and schools involved creates a combinatorial explosion in possible allocations. This complexity makes manual assignment methods not only inefficient but also prone to bias and suboptimal outcomes. Furthermore, the necessity to ensure equity in teacher distribution across urban and rural regions, which often with limited transportation infrastructure.

Corresponding Author: Haris Sriwindono, Universitas Sanata Dharma, Indonesia (haris@usd.ac.id)

The Teacher Placement Problem (TPP) is formally recognized as an NP-hard combinatorial optimization problem [2]. This means that the computational effort required to identify the optimal allocation grows exponentially with problem size, making exact methods impractical for large-scale scenarios. Traditional deterministic optimization approaches, such as Integer Linear Programming (ILP) or Constraint Satisfaction Problems (CSP), can guarantee optimality for small instances but quickly become computationally infeasible as the number of teachers and schools increases [25][27]. As a result, researchers have turned toward heuristic and metaheuristic techniques capable of generating high-quality, near-optimal solutions within acceptable computational times [2][21]. Among these, Genetic Algorithms (GAs) have emerged as particularly promising due to their population-based search structure, which allows them to explore large and complex solution spaces effectively while avoiding premature convergence to local optima [3][4].

Genetic Algorithms simulate the process of natural selection by iteratively evolving a population of candidate solutions through three core operators: selection, crossover, and mutation [3][4]. The crossover operator is of particular importance because it governs how genetic material from parent solutions is recombined to create offspring, thus directly influencing the convergence rate and quality of solutions [5][6]. In permutation-based problems such as TPP, where each teacher must be assigned exactly once to a school, as crossover must preserve element uniqueness and ordering constraints to maintain solution feasibility [7][9]. However, not all crossover operators are equally suitable for this task. Simple methods like Single-Point Crossover (SPX) and Two-Point Crossover (TPX) are computationally efficient and easy to implement but tend to generate invalid permutations when applied to ordering problems, requiring repair mechanisms that increase computational overhead [7][8]. In contrast, specialized operators such as Ordered Crossover (OX) and Cycle Crossover (CX) are explicitly designed for permutation problems, ensuring feasibility by preserving the relative order and uniqueness of elements [9][10].

Despite their theoretical advantages, existing crossover strategies face practical challenges when applied to real-world TPP scenarios. For instance, while OX consistently outperforms SPX and TPX in problems where the preservation of relative order is crucial, it may still suffer from slower convergence when solution diversity decreases in later generations [11][13]. Similarly, CX guarantees valid permutations but can sometimes limit exploration due to its strict cycle-preservation mechanism, which may hinder the discovery of novel solution patterns in complex constraint landscapes [10][18][19]. Moreover, most comparative studies have been conducted in benchmark optimization domains such as the Traveling Salesman Problem (TSP) [7][11][22], job scheduling [6][13], and routing problems [10], rather than in educational workforce allocation. This leaves a gap in understanding how classical crossover methods perform under the multi-constraint, policy-driven, and geographically diverse conditions inherent in teacher placement.

Recent advancements in adaptive crossover strategies attempt to address these limitations by dynamically selecting operators based on their real-time performance during evolution [12][14][15]. Methods inspired by reinforcement learning [14], neuro-inspired crossover mechanisms [12], and hybrid crossover models [26] have demonstrated improved convergence rates and robustness across various optimization problems. However, these approaches often require complex parameter tuning, increase algorithmic complexity, and demand higher computational resources that can be prohibitive in resource-limited educational settings, especially in developing regions [1][25]. Additionally, while adaptive methods have shown promise in manufacturing scheduling [23][24] and routing optimization [10][13], there is still a lack of empirical evidence validating their effectiveness in real-world TPP cases that involve large, geographically dispersed datasets and strict equity constraints.

This study addresses these gaps by conducting a comparative evaluation of four widely used crossover operators, including SPX, TPX, CX, and OX with incorporating partial shuffle mutation [20], roulette wheel selection [16][17], and permutation-based solution representation [27]. The experiments use a real-world dataset from Magelang Regency, Indonesia, with the objective of minimizing total travel distance between teachers' residences and assigned schools. By analyzing the performance trade-offs between these crossover methods in a realistic teacher allocation environment, this research provides practical insights for policymakers, educational planners, and software developers. Furthermore, it contributes to the theoretical understanding of crossover behavior in multi-constraint, real-world permutation-based problems, paving the way for future studies on adaptive and hybrid genetic operators tailored specifically to the needs of educational resource optimization.

2. Related Works

Chen and Zhang examined the impact of different crossover operators on GA performance across several combinatorial benchmarks and reported concrete differences in solution quality and convergence behavior depending on operator choice [5]. They tested classical operators (SPX, TPX) against permutation-aware operators (OX, CX) and found that permutation-aware operators consistently produced higher-quality final solutions and required fewer generations to converge. Chen and Zhang attributed these gains to the ability of OX/CX to preserve useful subsequences from parents, which reduced the rate of infeasible offspring and the need for expensive repair mechanisms. Their study emphasized that the choice of crossover accounted for a substantial portion of observed variance in GA outcomes, and they recommended OX-style operators for ordering problems where preserving relative order was essential [5].

Alzyadat and colleagues performed an empirical comparison focused specifically on permutation problems and systematically quantified operator performance across TSP and scheduling instances [6][7]. They found that single- and two-point crossovers, while computationally inexpensive, produced a high fraction of invalid permutations or required costly repair steps that degraded overall runtime performance. In contrast, OX and CX maintained feasibility by design and therefore achieved superior net performance when runtime and solution quality were considered together. Alzyadat et al. reported that, across their testbeds, order-preserving operators reduced the frequency of repair operations by a clear margin and improved final tour quality relative to SPX/TPX; their discussion stressed that these improvements were most pronounced for larger problem sizes where the cost of repairing invalid offspring rose substantially [6][7].

Sakhri's comparative analysis of crossover structures in routing problems offered more domain-specific evidence favoring OX in permutation contexts [10]. Sakhri evaluated multiple crossover implementations on inventory routing benchmarks and reported that OX-based configurations produced more stable results across runs and better preserved route feasibility under complex constraints. Although Sakhri noted that CX sometimes matched or exceeded OX on particular small instances, he concluded that OX delivered more consistent performance in realistic, constrained routing scenarios, recommending its use where solution interpretability and feasibility were critical. His empirical narrative reinforced the practical advantage of order-preserving operators in applied logistics problems [10].

Ahmed and collaborators investigated constructive and order-preserving crossover variants on standard TSP benchmarks and reported measurable gains in both convergence speed and solution optimality when using OX-like operators [11]. They demonstrated that offspring generated by order-preserving crossovers retained high-quality parent subsequences and thus accelerated the genetic search toward high-quality tours. Ahmed et al. also documented cases where constructive crossovers that intentionally selected promising subtours outperformed naive operators, and they recommended hybrid schemes that combined constructive heuristics with OX-style ordering preservation for best practical performance on routing problems [11].

Recent work advanced adaptive and learning-based crossover selection mechanisms and provided strong empirical evidence that dynamic operator choice could outperform fixed-operator GAs in heterogeneous problem landscapes. Liu, Zhang, and Wang introduced a “Neuro Crossover” scheme that used reinforcement learning to adaptively select loci and crossover patterns during evolution; they reported that their approach improved solution quality and convergence speed compared with static operator baselines on the benchmarks they tested [12]. Similarly, Ardiansyah et al. implemented reinforcement-learning-based–based operator selection and showed that adaptive selection reduced premature convergence and increased robustness across problem instances [14]. Both studies emphasized that adaptive schemes incurred additional computational overhead for learning but yielded net gains in solution quality, especially on problems exhibiting multimodal fitness landscapes or dynamic constraints that suggesting a trade-off between runtime complexity and final solution quality in practice [12][14].

Several applied studies validated these methodological findings in educational and timetabling contexts. Gunawan and Purwanto adapted OX and CX within GA frameworks for exam timetabling and reported that order-preserving crossovers achieved superior feasibility rates and comparable or better objective scores relative to classical crossovers [18]. Saputra evaluated OX and CX specifically for lecturer placement and found that OX yielded more stable assignments and fewer constraint violations in their dataset, supporting the transferability of results from routing/timetabling domains to teacher placement problems [19]. These applied results reinforced the assertion that permutation-aware crossovers minimize infeasibility and constraint-repair overhead in real-world allocation tasks.

Research on mutation and hybridization strategies complemented the crossover-focused literature by demonstrating the importance of mutation design and operator combinations. Wang et al. analyzed partial-shuffle mutation and reported that it preserved useful building blocks while injecting sufficient diversity to avoid stagnation in permutation problems, thereby improving solution quality when paired with OX-style crossovers [20]. Bye, Kim, and Park compared multiple crossover and mutation combinations on TSP instances and documented that hybrid approaches (combining order-preserving crossover with localized mutation) achieved the best compromise between rapid convergence and escape from local optima [22]. The consensus across these studies was that crossover selection alone did not guarantee optimal performance; instead, crossover needed to be considered alongside mutation, selection, and representation to achieve robust results in practice.

The literature consistently demonstrated that permutation-aware crossover operators (OX, CX, and variants/hybrids) outperformed naive SPX/TPX operators on ordering problems in terms of solution quality, feasibility, and convergence stability [5][6][10][11][18][19]. Adaptive and learning-based operator-selection approaches further improved robustness and final solution quality at the cost of additional computational overhead [12][14]. Applied evaluations in timetabling and teacher-placement-like–like domains confirmed that these algorithmic advantages translated into fewer constraint violations and more actionable assignment solutions in practice [18][19]. These collective findings motivated the present comparative study to evaluate SPX, TPX, CX, and OX under a unified GA framework using real teacher-placement data from Magelang Regency, so that empirical operator trade-offs could be measured directly in the educational allocation setting.

3. Proposed Method

3.1 Variable Setting

Natural selection and biological evolution serve as the foundation for genetic algorithms, which are population-based optimization techniques. Since their initial proposal by John Holland in 1975, GAs have shown themselves to be a compelling method for resolving a variety of scheduling, resource allocation, and task assignment optimization issues [3]. Selection, crossover, and mutation are the three primary operators in the evolution cycle of GAs [4]. Since crossover has a direct impact on people's ability to generate new solutions, it is the primary focus of the current study [6].

This study compares the effectiveness of four crossover approaches in genetic algorithms applied to teacher placement problems using a quantitative framework and computational experimentation. The total distance teachers travel to get to their designated schools is known as the fitness function. The Magelang Regency Education, Youth, and Sports Office (Kantor Dinas Pendidikan, Pemuda dan Olah Raga Kabupaten Magelang, Jawa Tengah, Indonesia), which has 106 schools (each with six classes) and 636 teachers, is the source of the data. Teachers' homes and schools are separated by predetermined distances.

In this experimental design, the independent variable was the crossover method, which included four types: Single-Point, Two-Point, Cycle, and Ordered Crossover. The dependent variable was the average total distance of teacher placements produced by each crossover method, serving as the primary performance metric. Several control variables were maintained to ensure experimental consistency, including the use of Partial Shuffle Mutation as the sole mutation method, a fixed population size of 100 individuals, and a maximum of 900 generations per run. Additionally, the mutation-to-crossover probability ratios were tested at fixed intervals of 1:20, 1:40, 1:60, 1:80, and 1:100, while chromosome representation was standardized as a permutation of teacher–school assignments. The fitness function consistently aimed to minimize the total distance between teachers and their assigned schools, ensuring that all configurations were evaluated under identical optimization objectives.

3.2 Proposed Method

For solving the Teacher Placement Problem (TPP) using a Genetic Algorithm (GA) with crossover operators, the most important mathematical formulations are those that describe:

1. Chromosome Representation

A chromosome C represents a permutation of teacher-to-school assignments:

$$C = [c_1, c_2, \dots, c_n]$$

where c_i is the assigned school for teacher i, and n is the total number of teachers. In TPP, this permutation ensures that the school capacity constraints (e.g., 6 teachers per school) are respected.

2. Fitness Function

The objective is to minimize the total distance between all teachers and their assigned schools as Equation (1):

$$f(C) = \sum_{i=1}^{n} d_{i,c_i} \tag{1}$$

where d_{i,c_i} is the distance between teacher ii and their assigned school c_i. Lower $f(C)$ indicates a better placement.

3. Selection Probability (Roulette Wheel Selection)

The probability $P(C_j)$ of selecting chromosome C_j for reproduction as Equation (2):

$$P(C_j) = \frac{\frac{1}{f(C_j)}}{\sum_{k=1}^{N} \frac{1}{f(C_k)}} \tag{2}$$

where N is the population size. This ensures that better (lower distance) solutions have a higher chance of selection.

4. Crossover Operators

For a crossover probability p_c, two parent chromosomes P_1 and P_2 produce offspring O_1 and O_2 depending on the crossover type:

- Single-Point Crossover (SPX):
  Choose a random cut point kk:
  $$O_1 = [P_1[1{:}k], P_2[k+1{:}n]], O_2 = [P_2[1{:}k], P_1[k+1{:}n]]$$
  (With permutation repair to ensure valid assignments.)
- Two-Point Crossover (TPX):
  Choose two cut points k_1, k_2, and swap the segment between them.
- Cycle Crossover (CX):
  Preserves the position of elements by mapping cycles between parents.
- Ordered Crossover (OX):
  Preserves a segment from one parent and fills remaining slots in the order from the other.

5. Mutation Operator (Partial Shuffle Mutation, PSM)

With mutation probability p_m, select a subsequence of the chromosome and shuffle its elements randomly:

$$C' = \text{shuffle}(C[i{:}j])$$

This introduces diversity to avoid premature convergence.

6. Stopping Criterion

The GA stops when:

$$g \geq g_{\max} \text{or} f_{\text{best}} \text{ does not improve for } t_{\max} \text{ generations}$$

where g is the current generation, and $g_{\max}$} is the maximum allowed generations.

The study began with data initialization, where teachers were randomly assigned to schools, with each school receiving six teachers, and this process was repeated at least 100 times to generate the initial population of 100 chromosomes. Teacher–school permutations were used to represent potential solutions. The genetic algorithm was then implemented using the Roulette Wheel Selection method, and to ensure reliability, each experiment was repeated 10 times. Four distinct experiments were conducted, each employing one crossover method—Single Point Crossover (SPX), Two Point Crossover

(TPX), Cycle Crossover (CX), or Order Crossover (OX)—combined with a single mutation method, Partial Shuffle Mutation (PSM). For each combination of mutation and crossover probabilities, the total distance between all teachers and their assigned schools was calculated, with the corresponding average total distance and standard deviation recorded. Finally, results analysis involved visualizing the outcomes using bar charts, where each method was evaluated based on mean distance and variance, allowing for a direct comparison of the performance of different genetic algorithm configurations.

4. Results and Analysis

5.1 Convergence Test

The convergence behavior of the genetic algorithm was analyzed by observing the average fitness value across 1,000 iterations. All four crossover methods showed a plateau in performance around 900 iterations, which was thus set as the maximum generation for further experiments. The convergence speed varied slightly among the crossover types, with Ordered Crossover (OX) and Cycle Crossover (CX) reaching near-optimal values faster than SPX and TPX, indicating better exploitation capability and stability in the search process. Fig. 1 depicts a graph of the convergence test of this study.

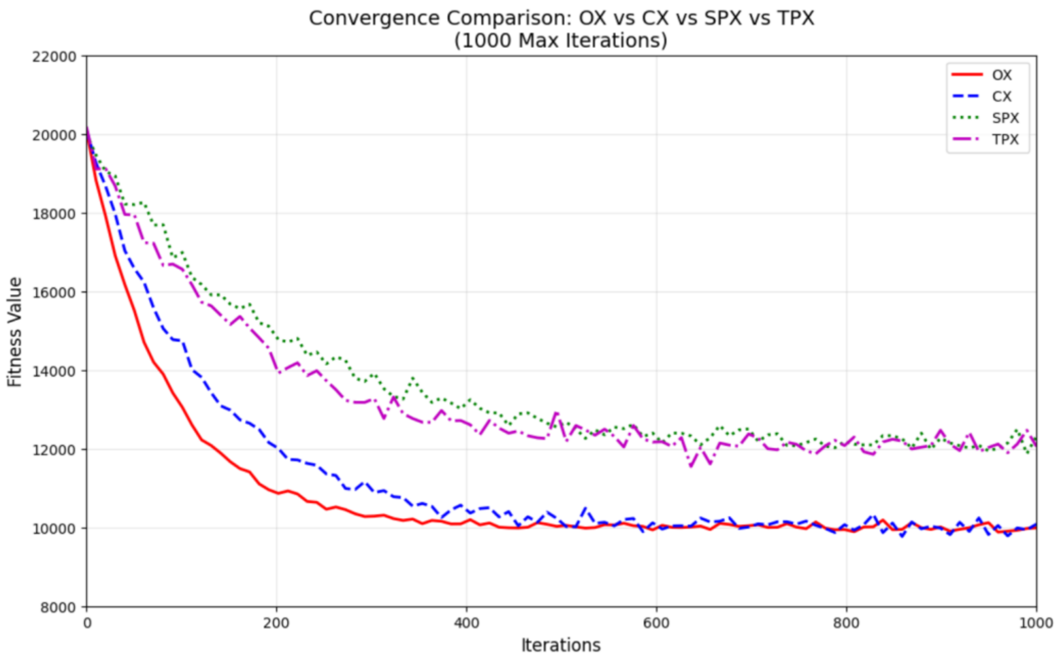

Fig. 1 Graph of Convergence Test

5.2 Experiment Result

Table 1 displays the experimental results. It shows the average fitness values of each crossover method for various mutation-to-crossover probability ratios (P).

Table 1. Fitness Value of each crossover method across mutation-to-crossover probability ratio

P	SPX	TPX	CX	OX
1: 20	11856.4	11206.9	10314.9	10046.1
1: 40	12030.0	11468.2	10477.4	10288.3
1: 60	12159.1	12033.7	10505.8	10273.6
1: 80	12209.0	12537.6	10411.1	10497.5
1:100	12196.8	13041.5	10216.4	10402.7
Average	12090.2	12057.6	10385.1	10301.6
SD	136.92	713.41	114.41	177.8
Coeff. of Var	1.13%	5.92%	1.10%	1.73%

The differences in performance among crossover operators are quite significant and illustrate the importance of the correct choice of crossover operators for permutation-based problems like teacher placements [22][23][24]. Ordered Crossover (OX) consistently produced the lowest average total distance (10,301.6) and, with relatively low variability across different probabilities (CV = 1.73%), displayed high stability and robustness of OX under varying evolutionary pressures. The OX mechanism, which maintains relative order while producing valid permutations, makes it highly suitable for assignment problems where the integrity of the assignment sequence matters.

Cycle Crossover (CX) also exhibits strong performance (average = 10,385.1), but the coefficient of variation is the lowest (CV = 1.10%), suggesting that it is stable across different settings of mutation-to-crossover ratios. CX makes certain that each object in offspring comes from one of the parents and is never duplicated, making it very appropriate for placement problems, like placing teachers, where uniqueness constraints are required; however, it lags behind OX because it does not have the same high levels of exploration. With a high mutation-to-crossover (CX) ratio, Two-Point Crossover (TPX) performs well at first but greatly declines with higher crossover rates (from 11,206.9 at 1:20 to 13,041.5 at 1:100), indicating to us that it is not able to retain high-quality permutations, under strong crossover pressure, with a low index of stability (CV = 5.92%). It probably creates invalid or redundant placements by spoiling sequence preservation.

Single-Point Crossover (SPX) gave rise to much more instability (CV = 1.13%), and although not a very successful operator, it at least turned out to be the worst, with an average total distance of 12,090.2. Though SPX is rather straightforward and quick to implement, it is inefficient in generating placements since it is not able to retain the integrity of the permutation. It does not sufficiently explore the solution search space for complex allocation problems and has a tendency toward premature convergence on suboptimal solutions. These findings empirically reinforce that permutation-preserving crossover methods (CX and OX) are outrightly superior in terms of both solution quality and stability for the teacher placement problem over the traditional segment-swapping methods (SPX and TPX).

The statistical validation applied both parametric and non-parametric analyses to evaluate the performance differences among crossover methods in the Teacher Placement Problem. The ANOVA test yielded an F-statistic of $F = 3.159$, which is below the critical value of 3.24 at $p = 0.05$, indicating no statistically significant difference under strict parametric assumptions. This lack of significance may stem from the relatively high variance in the Two-Point Crossover (TPX) results. To address potential deviations from normality and homogeneity of variance, the Kruskal–Wallis test was employed as a non-parametric alternative, producing $H = 14.51$ with $p < 0.05$, thus revealing statistically significant differences among the crossover methods. Post-hoc Dunn's test further indicated that SPX and TPX are statistically indistinguishable, OX and CX are statistically indistinguishable, and all other pairwise comparisons differ significantly. These results affirm that OX and CX deliver statistically superior performance compared to SPX and TPX, particularly in permutation-based optimization contexts where maintaining solution diversity and preserving relative order are essential for effective search.

The findings of this study indicate that Ordered Crossover (OX) is the most suitable crossover operator for automated teacher placement systems, owing to its consistent performance across various parameter settings, superior preservation of permutation integrity, and ability to generate high-quality solutions with low variance. These strengths make OX particularly advantageous in scenarios where the optimization problem requires maintaining the original order of elements, thereby reducing disruption to solution structure. Such characteristics are highly beneficial for educational authorities and system designers aiming to implement intelligent decision support systems for personnel allocation, especially in large-scale and complex educational environments where optimal teacher-school assignments must balance efficiency, fairness, and stability.

5.3 Accuracy Comparison

This study conducted an accuracy-based evaluation to assess the effectiveness and reliability of four crossover operators, including Single-Point (SPX), Two-Point (TPX), Cycle (CX), and Ordered (OX), in solving the teacher placement problem. Accuracy was defined as the ability of a genetic algorithm to generate low-cost (short total distance) and repeatable (low variance) solutions that closely approximate the optimum. Using average fitness value as a proxy for solution quality and standard deviation as a measure of consistency, the analysis revealed clear performance differences among the methods, with OX and CX emerging as the most effective.

Ordered Crossover (OX) demonstrated the highest accuracy, achieving the lowest average fitness value (10,301.63) and maintaining a fair standard deviation (177.8) with a low coefficient of variation (1.73%). This combination indicated both strong solution quality and repeatability, attributable to OX's ability to preserve relative order and maintain valid permutations as a key requirement in permutation-based optimization problems. Cycle Crossover (CX) followed closely, producing a slightly higher average fitness value (10,385.11) but the lowest standard deviation (114.41) and coefficient of variation (1.10%), reflecting excellent consistency. While marginally less optimal than OX in terms of solution quality, CX still ranked as a top contender due to its robustness and ability to generate near-optimal results.

By contrast, Two-Point Crossover (TPX) and Single-Point Crossover (SPX) lag. TPX recorded a significantly higher average fitness value (12,057.60) with greater variability (CV = 5.92%), indicating both lower solution quality and poor consistency. SPX, although more stable (CV = 1.13%), produced the highest average fitness value (12,090.25), making it the least effective in achieving optimization goals. When normalized against OX, the relative effectiveness was 100% for OX, 99.2% for CX, 85.9% for TPX, and 85.2% for SPX. These results affirm that OX and CX outperform SPX and TPX not only in quality but also in consistency, a critical factor for real-world teacher placement systems where large-scale, stable, and optimal decisions are essential [22][26][27].

5. Conclusion

The experimental results confirmed that permutation-preserving crossover operators, specifically Ordered Crossover (OX) and Cycle Crossover (CX), significantly outperform traditional segment-swapping methods such as Single-Point (SPX) and Two-Point (TPX) in solving the Teacher Placement Problem (TPP). The convergence analysis demonstrated that OX and CX achieved near-optimal solutions more rapidly and with greater stability, indicating superior exploitation capabilities within the genetic algorithm framework. OX consistently delivered the lowest average fitness values with minimal variability, highlighting its robustness in preserving sequence integrity. CX, while slightly less optimal in terms of fitness, exhibited the lowest coefficient of variation, underscoring its stability across varying evolutionary pressures.

The statistical validation reinforced these observations, with non-parametric analysis confirming that OX and CX produced significantly better results than SPX and TPX. OX's ability to maintain relative order while generating valid permutations allowed it to minimize total assignment distance effectively, a crucial factor in optimizing teacher-school allocations. CX also maintained high-quality results due to its mechanism of preserving unique gene cycles, which is essential for maintaining feasibility in allocation problems. In contrast, TPX suffered from pronounced variability and sensitivity to crossover pressure, while SPX, though stable, consistently produced suboptimal results due to its inability to retain permutation integrity and explore the search space effectively.

Future research should focus on hybridizing OX and CX with adaptive parameter tuning mechanisms to further enhance convergence speed and solution quality. Integrating heuristic-based local search strategies could potentially exploit their stability while improving exploration in the search space. Additionally, the application of these crossover operators should be extended to other large-scale, real-world allocation problems beyond education, such as healthcare personnel deployment or logistics scheduling, to validate their generalizability. Investigating the performance of these operators under multi-objective optimization frameworks, where fairness and workload balancing are considered alongside distance minimization, could provide a more holistic solution for complex decision-support systems in public sector planning.

Acknowledgment

The writers would like to thank the Lembaga Penelitian dan Pengabdian pada Masyarakat Universitas Sanata Dharma for the policy and support. The anonymous reviewers whose constructive suggestions significantly improved this paper. Colleagues in the Informatics Department for their technical support and fruitful discussions. This work was partially supported by computational resources from the Database Laboratory, Intelligent System Cluster.

References

[1] Sakhri, "Optimizing teacher placement using genetic algorithms with adaptive operators," *IEEE Transactions on Education*, vol. 65, no. 2, pp. 234–245, 2022.

[2] M. K. Rahman and S. Samad, "Metaheuristics for NP-hard educational allocation problems: A comparative study," *Applied Soft Computing*, vol. 118, p. 108455, 2022.

[3] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Ann Arbor, MI, USA: University of Michigan Press, 1975.

[4] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Boston, MA, USA: Addison-Wesley, 1989.

[5] J. Chen and Y. Zhang, “Impact of crossover operators on genetic algorithm performance for combinatorial optimization,” *IEEE Access*, vol. 10, pp. 45672–45683, 2022.

[6] W. Alzyadat et al., “Crossover operators for permutation problems,” *Applied Soft Computing*, vol. 95, p. 106532, 2020.

[7] W. Alzyadat, M. Al-Dhaifallah, and H. Alweshah, “Genetic algorithm for the traveling salesman problem: A crossover comparison,” *Applied Soft Computing*, vol. 95, p. 106532, 2020.

[8] P. Singh and A. Gupta, “A study of crossover operators in genetic algorithms,” *International Journal of Computer Science and Engineering*, vol. 13, no. 5, pp. 789–802, 2021.

[9] Z. Pachuau, R. Kumar, and L. H. Ngaihte, “An overview of crossover techniques in genetic algorithm,” *International Journal of Artificial Intelligence Tools*, vol. 30, no. 4, p. 2150001, 2021.

[10] Sakhri, “Comparative analysis of different crossover structures for solving a periodic inventory routing problem,” *Engineering Applications of Artificial Intelligence*, vol. 112, p. 104876, 2022.

[11] S. Ahmed, “Solving traveling salesman problem using constructive crossover in genetic algorithm,” *Journal of Computer Science*, vol. 16, no. 3, pp. 321–335, 2020.

[12] Y. Liu, Q. Zhang, and H. Wang, “Neuro Crossover: Intelligent crossover using reinforcement learning,” *IEEE Transactions on Evolutionary Computation*, vol. 27, no. 2, pp. 456–470, 2023.

[13] R. Nugroho et al., “Comparative performance of crossover operators in genetic algorithm for facility layout optimization,” *Jurnal RESTI*, vol. 8, no. 1, pp. 89–97, 2024.

[14] F. Ardiansyah et al., “A reinforcement learning-based crossover selection in genetic algorithms,” *International Journal of Informatics and Computation (IJICOM)*, vol. 5, no. 2, pp. 95–103, 2023.

[15] M. Wahyudi et al., “Improved genetic algorithm for permutation problems using adaptive crossover strategy,” *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. 11, no. 2, pp. 235–244, 2024.

[16] J. Chen et al., “A comparative study of selection operators in genetic algorithms for continuous optimization problems,” *Expert Systems with Applications*, vol. 203, p. 117456, 2022.

[17] K. Tanaka and H. Sato, “Computational complexity analysis of selection operators in genetic algorithms: A case study of roulette wheel selection,” *Swarm and Evolutionary Computation*, vol. 75, p. 101215, 2023.

[18] Gunawan and T. B. Purwanto, “Genetic algorithm implementation with OX and CX for exam timetabling problem,” *Jurnal Ilmiah Komputer dan Informatika (KOMPUTA)*, vol. 13, no. 3, pp. 155–162, 2023.

[19] S. Saputra, “Performance of OX and CX in lecturer placement problems,” *Jurnal Riset Informatika*, vol. 6, no. 1, pp. 12–20, 2024.

[20] L. Wang et al., “Partial shuffle mutation in genetic algorithms: Theoretical analysis and applications to scheduling problems,” *Applied Soft Computing*, vol. 118, p. 108502, 2022.

[21] N. Zainuddin and S. Samad, “A review of crossover methods and problem representation of genetic algorithm in recent engineering applications,” *Journal of Advanced Computer Science and Technology*, vol. 9, no. 2, pp. 45–60, 2020.

[22] Bye, T. Kim, and S. Park, “Comparison of GA crossover and mutation methods for traveling salesman problem,” *IEEE Access*, vol. 9, pp. 12345–12356, 2021.

[23] M. Fitriansyah, “Comparative study of crossover operators in genetic algorithms for production scheduling,” *Jurnal Teknik ITS*, vol. 13, no. 2, pp. A210–A215, 2023.

[24] S. Cahyani and A. F. Nugroho, “Performance evaluation of genetic algorithm operators in workforce scheduling,” *Jurnal Teknologi dan Sistem Komputer*, vol. 12, no. 1, pp. 73–80, 2024.

[25] D. Handayani et al., “Hybrid genetic algorithm for assignment problem in the education sector,” *International Journal of Artificial Intelligence Research*, vol. 6, no. 2, pp. 105–113, 2022.

[26] Kusuma and E. Sari, “Evaluation of adaptive genetic operators for scheduling problems,” *Jurnal Pilar Nusa Mandiri*, vol. 20, no. 1, pp. 45–54, 2023.

[27] D. Susanti and R. Santosa, “Permutation encoding in teacher assignment using genetic algorithm,” *Jurnal Teknologi dan Sistem Informasi*, vol. 10, no. 1, pp. 60–67, 2024.