# Nurse scheduling problem: investigating the principles of operators in evolutionary algorithm for small size population

*Huai Tein* Lim [1*], *Irene-Seok Ching* Yong[2], *Peh Sang* Ng[1], and *Poh Choo* Song[1]

[1]Department of Physical and Mathematical Science, Faculty of Science, Universiti Tunku Abdul Rahman, 31900 Kampar, Perak, Malaysia
[2]Centre for the Initiation of Talent and Industrial Training, University of Malaya, Kuala Lumpur, 50603, Malaysia.

**Abstract.** Developing an effective nurse shifts assignment system, that considers diverse nurse preferences and fairness as well as ward coverage in practical operational scenarios, is a complex and time-consuming task. Failing to address various constraints with different levels of precedence can lead to undesirable nurse schedules. The efficiency of such a system relies heavily on the attributes of an automated scheduling approach or the proficiency of a head nurse. Therefore, this paper investigates the principles of designing artificial computing operators for a matrix representation solution in the evolutionary algorithm hybridization. Several parent selections, each with different selection intensities that prioritize elitism and dissimilarity characteristics, are reviewed. Additionally, the integration of parent selection intensities with specific fragment sizes of crossovers are studied when designing a well-performing algorithm. The evaluation criteria encompass algorithm reliability, accuracy, effectiveness, and efficiency. The study reveals that the modified Maximax and Maximin parent selection with Block-wise crossover achieved a higher quality schedule with the lowest fitness value. In conclusion, a small-sized population proves suitable for addressing the complex computational problem that consist of heavy constraints. The selection intensity should strike a balance between elitism and dissimilarity intensities when combined with a smaller fragment size of mating strategy.

## 1 Introduction

In the ever-dynamic healthcare environment, the ability of a head nurse to make prompt and informed decisions is crucial, especially during unexpected challenges and workforce shortages. The research delves into the multifaceted nature of the nurse workforce's unhealthy working environment, extending beyond mere shortages to encompass factors such as changes in skill mix [1], nurse burnout [2, 3, 4], nurse preferences and fairness in shift [5, 6, 7], as well as understaffing due to uncertain absenteeism [5, 8]. Therefore, nurse scheduling problem (NSP) emerges as a critical challenge, involving the allocation of

*Corresponding author: limhuait@utar.edu.my

nurses to perform shift duties within a specified timeframe while adhering to various constraints. In other words, the complexity of nurse shift assignment problems, where shifts need to be allocated to nurses in a way that meets both hard constraints (essential criteria) and soft constraints (desirable criteria) to ensure a viable and high-quality schedule.

NSP is a multifaceted nature of nurse shift assignment problems. According to [9-11], it is evident that the constraints encompass various factors. Firstly, there are the regulations specific to the wards, which include aspects such as nurse workload, required skills, hospital policies, contractual agreements, and shift requirements. Secondly, real-time working practices come into play, accounting for demand fluctuations and seasonal variations that impact scheduling. Lastly, the preferences of the nurses themselves must be factored in, ensuring that their individual needs and desires are accommodated within the scheduling framework.

Practically, the manual production of these schedules by head nurses is highlighted as a challenging task, requiring significant paperwork, time, and effort. However, when automated approaches are employed, the difficult constraints of nurse scheduling add complexity, illustrating the real-world implications of these challenges. Due to that fact that automated approaches must possess the intelligence to handle subjective nurse preferences and dynamic staffing size akin to an experienced head nurse.

Technically, mathematical programming, although a conventional automated approach, proves less effective in solving NSP due to its classification as an NP-hard problem [9-11]. Therefore, evolutionary computation become imperative [12-14]. Evolutionary algorithm (EA) is considered as a method of searching optimization that inspired by the principles of natural selection [13, 14]. The algorithm maintains a population of individuals (solutions), and through a series of iterative processes that involve parent selection, crossover, and mutation operations, thereby mimic the evolution of biological organisms. Evolutionary algorithms possess stochastic characteristics, ensuring adaptability and randomness in the processes to search approximate solutions.

However, an often-problematic aspect in Evolutionary algorithm studies is setting operators parameters and balancing exploration and exploitation search [11-12, 15-17]. The setting of population size in nurse scheduling problem varies within the range 10-200, for instance, population size of 160 in a tuning range of 20-160 [18], 50-200 [19], 80 [20], 30 [21], 10 in range of 10-50 [22], and 15 [23]. Hence, the lower bound population size of approximately 10 can be considered small for EA matrix representation in NSP. There is a lack of studies on population sizing for population-based metaheuristics, and inappropriate sizes may impact convergence speed and model accuracy [24]. Heavy computation time exits when population size increased in a highly constrained problem [12]. Moreover, selection pressure dictates which individuals are chosen for reproduction in each generation. With high selection pressure, individuals undergo a strong competitive process among the population, resulting in only a small proportion of individuals is being selected for reproduction, typically those with the highest fitness values.

Therefore, careful consideration of factors such as population size, selective pressure, diversity, and randomization is vital for achieving a delicate balance in EA. This paper underscores the cooperation of reproduction operators and parent selection on the potential pitfalls of high selection intensity, which may lead to rapid convergence in scenarios involving a small population size and high selection pressure, trapping the algorithm in local optima. This aspect becomes pivotal in exploring the elitism and dissimilarity intensities of parent selection operator, impacting the flexibility of reproduction operators. Consequently, the paper emphasizes the importance of designing integration principles that focus on combining selection intensity with specific fragment sizes of crossovers to address the challenges posed by small population size in EA for NSP.

The main objective of the research is to enhance nurse scheduling through Evolutionary Algorithms which can generate desirable nurse workforce schedules, accommodating diverse nurse preferences and fairness while ensuring essential ward coverage. The proposed model seeks to minimize penalty values associated with constraint violations, striving to create schedules that optimally satisfy various constraints with different levels of precedence. To establish more accurate constraints by aligning with real-world conditions, head nurses were interviewed to obtain essential data on daily nurse coverage in each shift. In this research, the data of a lower bound number and an ideal number of nurses needed for the Cardiac Rehabilitation Ward (CRW) and Emergency Department (ED) at Hospital Sultanah Bahiyah (HSB) in Malaysia were collected which shown in Table 1.

**Table 1.** Number of nurses needed in daily basis.

| Nurse Coverage | CRW | ED | Model's Benchmark |
|---|---|---|---|
| No. of avalable nurses | 24 | 34 | 39 |
| Senior nurse needed for each shift | ≥ 1 | ≥ 1 | ≥ 1 |
| Nurses needed per *M,* morning shift | 4 - 5 | 5 - 7 | 6 - 8 |
| Nurses needed per *E*, evening shift | 4 - 5 | 5 - 7 | 6 - 8 |
| Nurses needed per *N*, night shift | 3 - 4 | 5 - 7 | 5 - 7 |

To generate a model, the constraints are usually focused on ward regulation, capacity (nurse coverage and competency), and preference (Nurse preference on shift ordering and nurse preference on off duty in terms of fairness [11]. Based on the interviews, the preferences which subject to nurse preferences on shift ordering and nurse preferences on off duty in terms of fairness are classified, but quite hard to be fulfilled in practical working conditions. The capacity is subject to nurse coverage and nurse competency. For this N-P hard problem, hard constraint was used to strict policies that determined the feasibility of a schedule and soft constraint was conformed to subjective personnel desires which were not necessarily be satisfied [1, 7, 8]. Semi-hard constraint is employed to alleviate the stringent nature of hard constraint, but it also may cause infeasible solution if many semi-hard constraints have been violated [11].

The ward regulations in this scheduling model are all classified as the stringent hard constraints. For instances, each nurse is limited to undertaking a single shift per day; nurses are obligated to work six days a week; and it is strictly forbidden for any nurse to work more than six consecutive days. In the aspect of comprehensive nurse coverage (refer to Table 1), it covertly involves several constraints categorized as hard, semi-hard, and soft. The hard constraints mandate a minimum of 6 nurses for both the Morning ($M$) and Evening ($E$) shifts, with the Night ($N$) shift requiring at least five nurses, including assigning at least one senior nurse to each shift. Semi-hard constraints set a tolerable coverage range of 16% to 21% of the total nursing staff for both the Morning ($M$) and Evening ($E$) shifts. Soft constraints specify an ideal coverage of 21% of total nurses for Morning ($M$) and Evening ($E$) shifts, with a slightly lower ideal allocation of 19% for a Night ($N$) shift in a ward.

In considering nurse preferences for shift ordering, the scheduling model integrates various hard constraints. Nurses are entitled to two off days to pay compensation for three consecutive Night Shift duties (i.e., $N,N,N,$Off,Off). The model strictly forbids the sequence of a $N$ followed by a $M$ on the subsequent day as the consecutive work shifts. Furthermore, when reassigning shifts, the adherence to the rule of Forward Clockwise is preferably mandated (i.e., $M$<$M/E/N/$Off<$N/E/$Off<Off/$N$<Off). Additionally, the model incorporates semi-hard constraints discouraging instances of split off days or single workday for nurses, unless a nurse specifically requests the day off. These constraints are vital for work shift

ordering that underscore nurse preferences effectively. Moreover, there are semi-hard and soft constraints in this model designed to address nurse preferences for off-duty, emphasizing fairness and considering the individual needs and requests of nurses. A semi-hard constraint highlights the importance of providing timely off during weekends, guaranteeing each nurse at least one weekend off within a two-week schedule. Soft constraints underscore the approval of nurse requests for off days as a personal right. The model further ensures equity by practicing an equivalent total number of Morning ($M$) and Evening ($E$) shifts ($[M+E]\ /2 \pm [1 \text{ or } 2]$) for each nurse, maintaining a balanced distribution of these shifts. Additionally, the model encourages the provision of a stretch of consecutive off days for each nurse, promoting a more balanced and favourable schedule that aligns with individual preferences.

As a fitness evaluation in EA, the NSP model is optimized by minimizing the fitness value (see equation 1):

$$f(i) = \min \sum_{1}^{k} W_k C_k(i), \tag{1}$$

where $W_k$ be the weight of the violated constraint of type $k$, and $C_k(i)$ is the number of violated constraint of type $k$ in schedule $i$, an individual.

$$C_k(i) = \begin{cases} 1, & if\ constraint - type\ k\ exists\ for\ each\ nurse\ in\ a\ schedule\ i. \\ 0, & Otherwise. \end{cases} \tag{2}$$

To better monitoring and assessing a schedule's quality, the penalty values, $W_k$, for three types of constraint violation (i.e., hard, semi-hard, and soft constraints) are weighted in range of at least 1000000, at least 1000, and at least 1, respectively. This means that greater weightages were set for greater important constraints. The evaluation criteria cover the algorithm's reliability, measured by the feasible rate; accuracy, assessed through average fitness and standard deviation; effectiveness, determined by best fitness and convergence level; and efficiency, gauged by computation time.

## 2 The Operators and Intricacies of Evolutionary Algorithm

This paper aims to comprehend the intricacies of operators for balancing the exploration and exploitation in the search process. The main operators of EA consist of parent selection, crossover, and mutation. The parent selection operator is concerned with selecting certain individuals from the initial population that have the potential to produce good offspring for the next generation. However, high selection pressure in a small population may lead to premature convergence [11]. Therefore, developing a parent selection operator that provides enough diverse permutation space for the crossover operator to explore and preserves dissimilar individuals in a small-sized population are quite challenging.

To adapt to various population sizes and maintain population diversity, our focus was on preserving distinct solutions within a population. Therefore, dissimilarity intensity, and elitism intensity are incorporated and studied in the mating strategies. In essence, dissimilarity intensity influences how closely related the selected pair is, contributing to the maintenance of diversity within a population. Elitism intensity determines which individuals are selected based on superior genes, ensuring a promising solution space. As stated in [25], the contribution of high-quality genes is potentially leading to the development of a better new generation.

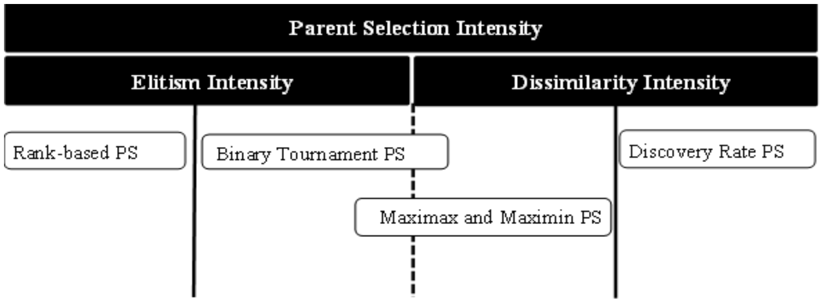


**Fig. 1.** Parent Selection Intensity in various parent selection operators in EAs.

Selective pressure in Fig. 1 is studied by considering the dual role (i.e., elitism intensity and dissimilarity intensity) respectively or simultaneously. To understand further, four parent selection operators in EA were experimented. Rank-based parent selection (Rk) is purely focused on elitism intensity whereas Discovery Rate parent selection (Dr) merely engrosses on dissimilarity intensity. In place of balancing exploration and exploitation in parent selection operator, Maximax and Maximin (MX), and Binary Tournament (T) execute elitism and dissimilarity intensities simultaneously but at different levels. Further details on the operators are shown in Fig. 2.

*Rk Parent Selection*

1) Rank individuals' fitness in a population in ascending order.
2) The 2 highest-ranked individuals will be selected and indicated as parent 1 and parent 2, representing elite parents.

*T Parent Selection*

1) Randomly pick 2 individuals.
2) Compare their fitness value.
3) Choose the individual with a lower fitness as parent 1.
4) Place the less fit individual to the group of remaining individuals in the population.
5) Repeat the whole process to get parent 2.

*MX Parent Selection*

1) Randomly partition half of the total population into a sub-population.
2) In the sub-population, rank the individuals in ascending order.
3) Classify the first half of the ranked sub-population as a group with the maximum outcome, and the second half as a group with the minimum outcome due to poorer fitness.
4) Select the best individual from each of the two outcome groups.
5) Define the selected individuals in step (4) as Parent 1 (i.e., the best in the first half, Maximax) and Parent 2 (i.e., the best in the second half, Maximin).

*Dr Parent Selection*

1) Select two individuals randomly from a population.
2) Determine a probability rate of discovery ($P_d$) and compute the Relative Difference, RD (x, y) = $\frac{|x-y|}{\max(x,y)}$
3) Verify the discovery by comparing the Relative Difference with the probability rate of discovery. If $P_d \leq$ RD, accept the two selected individuals as parents 1 and parent 2 for the next recombination purpose. If $P_d \geq$ RD, resume the two individuals to the population, and iterate through steps (i) to (iii) again.

**Fig. 2.** Procedure of various parent selection operators.

Rk parent selection is a selection based on the position of an individual in a particular sequence according to its fitness evaluation. A predominantly intensification is stimulated in this Rk selection scheme as suggested by [26]. Due to elitism element involved, Rk parent selection was implemented for our model validation purpose. The T parent selection, derived from [27], conducts a competition among two or more individuals randomly chosen from the population. This method is uncomplicated and avoids the need for fitness sorting, providing every individual with an equal opportunity for selection to uphold diversity. However, the tournament activity tends to emphasize elitism intensity, possibly impacting the speed of convergence. The proposed MX parent selection is inspired by the concept of Maximax and Maximin which is a decision-making approach under uncertainty [28]. MX is well-suited for application in an uncertain population pool with diverse individual characteristics. In this operator, incorporating dissimilarity intensity removes the need for filtering the initial population and subtly pressures on the selection of a superior individual within a subgroup. In this context, a slight distance between the complementary parents is ensured to uphold diversity in search. Dr that derived from [11] is a representative of dissimilarity intensity operator that is inspired by the process of discovering alien eggs in cuckoo search. Dr parent selection could be suggestive of the difference between selected parents. The diversity of a small-sized population should be maintained with at least 40% differences to offer a diverse permutation space for recombination operators.

Furthermore, the determinations of specific fragment sizes for crossover operators and the integration of selection intensities play crucial roles in designing a well-performing algorithm. In this paper, cut-off point in Row-wise crossover (Row) is a horizontal line that separates different type of nurse in this NSP. Thus, the whole senior nurse section in parent 1 may swap places with the senior nurse section in parent 2. Nevertheless, Block-wise crossover (Bw) involves setting vertical and horizontal cutoff points in a matrix representation of an individual. A section of a week (i.e., 7 days) is designated as the vertical cut-off point, where the senior nurse section has a horizontal cut-off point. In this setup, the swapping areas consist of half of the senior nurse section and half of the junior nurse section in a matrix representation between parent 1 and parent 2. In all, smaller fragment sizes of Bw crossover could enhance flexibility in swapping, hoping to achieve better exploitation and exploration.

Directed mutation operator implied in this study operated over each individual, also in a partially random manner, targeting specific objectives within segments of the individual. For instance, to adhere to the rule of forward clockwise scheduling, a random duty shift is swapped with an *E* shift followed by an *N* shift. Additionally, a random off day from a set of splits-off days is swapped with a single workday. The underlying reason for constructing directed mutation is to ensure that the constraints of an individual are not violated in the next generation. By considering a slightly changed concept of a mutation operator and avoiding a few instances of constraint violation in NSP, the directed mutation approach aims to enhance the robustness of the evolutionary process.

## 3 Experimental Result of Population Size

The EAs were programmed using MATLAB R2010a that equipped with an Intel® Core™ i5-11320H CPU @ 3.20 GHz and 12.0 GB of RAM. The parameter setting experiments were conducted to fit parent selections with Bw in different sizes of population, aiming to exhibit superior performance. The EA that combines MX with Bw is denoted as MX_Bw, and similar nomenclature conventions are applied to other combinations of operators discussed in this paper.

**Table 2.** Several Population Sizes for the two modified EAs.

| **Population size** | **10** | **12** | **14** | **16** | **18** | **20** | **30** | **40** |
|---|---|---|---|---|---|---|---|---|
| ***MX_Bw*** | | | | | | | | |
| Best Fitness | 4041 | **2046** | 4038 | 3046 | 3063 | 3049 | 4041 | 5043 |
| Average Fitness | 5051 | **4046** | 5048 | 6048 | 5050 | 4050 | 5051 | 6048 |
| STD ('000) | 0.58347 | 1.63479 | 0.53951 | 1.50539 | 1.88714 | 1.1877 | 0.69318 | 1.01497 |
| Feasible % | 4/20 | 6/20 | 8/20 | 11/20 | 8/20 | 15/20 | 7/20 | 20/20 |
| ***Dr_Bw*** | | | | | | | | |
| Best Fitness | 4040 | 3041 | 4044 | 3044 | 3045 | 3048 | **2062** | 4046 |
| Average Fitness | 5051 | **4047** | 5045 | 6050 | 5049 | 4050 | 6050 | 6052 |
| STD ('000) | 1.16397 | 0.91464 | 0.54791 | 1.78417 | 0.98651 | 0.99038 | 1.11732 | 1.36122 |
| Feasible % | 3/20 | 8/20 | 6/20 | 12/20 | 12/20 | 15/20 | 13/20 | 16/20 |

Based on the insights gleaned from Table 2, it is evident that a larger population size tends to enhance the success rate of feasibility but falls short in achieving effective optimal solutions. A comparative analysis, starting with a 10-unit difference in sizes (i.e., 10, 20, 30, 40), reveals that a population size of 20 emerges as the most favourable parameter within this range. This provisional optimal population size of 20 consistently yielded the best average fitness of 4050 and a slightly higher feasibility rate ((15/20) *100) = 75% in both MX_Bw and Dr_Bw. Furthermore, it resulted in the best fitness of 3049 in MX_Bw, although Dr_Bw obtained the second-best fitness of 3048. Notably, there was a discernible improvement from a size of 10 to 20, characterized by a lower best fitness and average fitness. Conversely, the transition from a size of 20 to 40 showcased inferior performance, marked by higher best fitness and average fitness, despite Dr_Bw achieving the best fitness of 2062 but high average fitness in a size of 30.

To refine the precision of our investigation, the experiment strategically focused on outcomes within the range of population sizes 10 to 20. Two proposed parent selection approaches were closely tested across groups with sizes 10, 12, 14, 16, 18, 20, 30, and 40. Despite achieving the second-best fitness with a value of 3041 in Dr_Bw, a population size of 12 resulted in the lowest average fitness at 4047. Additionally, in MX_Bw, which was the superior model among all, the population size of 12 exhibited both the lowest fitness of 2046 and the lowest average fitness of 4046. Consequently, aiming to identify the optimal (lowest fitness) schedule across all population sizes, a population size of 12 was selected for subsequent experiments.

## 4 Experimental Result of EAs Comparison

Table 3 illustrates a comparative analysis of output across different parent selections and crossover operators. The experiment employed fixed parameters, including 100 generations, a population size of 12, and 30 runs. To ensure fair comparisons, the EAs maintained consistent operators which are directed mutation and steady-state replacement strategy. Convergence level is determined by assessing fitness improvement over successive iterations. For example, a convergence level of 9 indicates that no fitness improvement has been observed starting from the 9th iteration. This could monitor the effectiveness of a model optimization in terms of solution stability.

**Table 3.** EAs with different parent selections and crossovers.

| | > Elitism Intensity PS | | | | > Dissimilarity Intensity PS | | | |
|---|---|---|---|---|---|---|---|---|
| **Parent Selection** | **Rk_Row** | **Rk_Bw** | **T_Row** | **T_Bw** | **MX_Row** | **MX_Bw** | **Dr_Row** | **Dr_Bw** |
| Best Fitness | 4040 | 4062 | 3033 | 3040 | 3038 | 2046 | 5039 | 3041 |
| Time (Sec) | 189.2 | 124.55 | 114.35 | 129.77 | 129.40 | 193.55 | 352.60 | 135.23 |
| Convergence level | 9 | 8 | 98 | 15 | 25 | 97 | 40 | 93 |
| Average Fitness | 4040 | 6051 | 4039 | 5049 | 4041 | 5050 | 5039 | 5050 |
| STD ('000) | - | 1.64026 | 1.09946 | 1.78316 | 1.41775 | 1.78316 | 0 | 1.06143 |
| Feasible rate | 1/30 | 5/30 | 5/30 | 12/30 | 4/30 | 9/30 | 4/30 | 10/30 |

Overall, the models presented in Table 3 were able to produce a best-so-far solution, representing an acceptable schedule with the violation of several semi-hard and soft constraints. In comparison, MX_Bw demonstrated superior performance by achieving the lowest best fitness of 2046 and a compatible average fitness of 5050. Perhaps, MX_Bw's superiority suggests an optimal balance between integrating elitism and dissimilarity intensities in parent selection, along with diverse crossover using smaller fragment cells. The optimal schedule with the best fitness value (i.e., 2046) effectively accommodated the compensated days off for three consecutive night shift duties, and all requested days off for all 39 nurses. The specified nurse coverages outlined in the model's benchmark (see Table 1) have been successfully met in the nurse schedule. Moreover, none of the nurses are scheduled to work more than 6 consecutive days, and their shifts are allocated according to the forward clockwise direction rule, ensuring that no *N* shift is followed by a *M* shift in any adjacent work shifts.

As an enhancement over Rk parent selection, MX parent selection effectively addressed the issue of fast convergence observed in Rk models (i.e., 9 and 8 convergence level). This led to a notable increase in the feasible rate, progressing from 1/30 to 4/30 and 5/30 to 9/30, respectively. These enhancements highlight the critical role of dissimilarity intensity in mitigating the challenge of premature convergence. However, it's essential to note that despite these positive outcomes, MX parent selection exhibited weaknesses in terms of time efficiency, requiring 193.55 seconds of computational time, and achieving a slightly inferior feasibility rate compared to models which regards to T.

Moreover, Dr_Bw performed mediocre outcomes, yet it outperformed the Rk parent selection across all aspects. This suggests that the substantial dissimilarity intensity implicitly emphasizes the importance of prioritizing exploration over exploitation in the design of evolutionary algorithms. Moreover, Dr_Bw showed a slight weak in terms of the best fitness compared to T_Row, T_Bw, MX_Row, and MX_Bw. For instance, when comparing the best fit from the proposed MX_Bw, Dr_Bw was defeated by 1 semi-hard constraint violated but slightly outperformed by 12.2% (i.e., (46-41)/41) on soft constraint.

As referred to the strong elitism intensity in the parent selection operator, it's noteworthy that the performances of Row models were better than Bw models, suggesting that these types of parent selection operators should inject more exploration elements into the crossover operator. On the other hand, the performance of Bw models surpassed that of Row models when their parent selection operators deviated towards dissimilarity intensity. This implies that there should be a greater emphasis on incorporating the exploitation element into the crossover process when coupled with a parent selection operator with heavy dissimilarity intensity.

Nevertheless, a shared drawback among all these models was their weak reliability, as evidenced by low feasible rates that did not exceed 40% in successfully generating feasible solution per run. Based on our findings, this condition could be ameliorated by enhancing the recombination operators, a prospect to be addressed in future research.

## 5 Conclusion

As a conclusion, the NSP model not only heightened the probability of nurses securing significant Request Off days and an even distribution of significant weekend off days but also ensured a well-balanced allocation of work shifts. The model further highlighted nurse workload fairness by maintaining a mean staffing level per shift category.

To enhance the performance of offspring production, the refinement of both parent selection and crossover operators should consider aspects such as population diversification, selective pressure, the randomization principle, and convergence issues. Specifically, the intensities of elitism and dissimilarity were addressed in the context of selective pressure. Consequently, classical parent selections in EAs (i.e., T, and Rk) and newly modified parent selections in EAs (i.e., MX, and Dr) were combined with two different fragment sizes of crossovers (i.e., Row, and Bw).

High elitism intensity helped find the best fit but may cause premature convergence in evolutionary algorithms. This high selection intensity may accelerate the convergence towards optimal solutions by favouring the selection of the fittest individuals, but it also increases the risk of losing diversity within the population. By this finding, the incorporation of dissimilarity intensity in selections was designed to actively shape population diversity by pinpointing unique and less-explored regions within the population. This strategic approach aimed to prevent the premature convergence of the algorithm.

Turning attention to the aspect of population size, conventional wisdom viewed a small population size unfavourably, anticipating a potential loss of diversity. Contrary to this convention, our research challenges this perspective, revealing that a smaller population size can be advantageous, especially for problems with heavy constraints leading to a low permutation in the initial population. As a result, a population size of 12 is recommended in this study. Perhaps, opting for a larger population size increases the risk of generating relatively similar individuals, signifying a reduction in diversity. It is imperative to emphasize the significance of flexibility in the search process facilitated by the crossover operator to mitigate the looming threat of premature convergence. This highlights the potential challenge for future research in exploring how a small-sized population, equipped with flexible search capabilities in the crossover operator, can effectively navigate and address this concern.

## References

1. J.F. Bard, H.W. Purnomo, Health Care Manag. Sci., **8**, 315 – 324 (2005)
2. B.J. Kalisch, M. Aebersold, Interruptions and multitasking in nursing care. Jt. Comm. J. Qual. Patient Saf., **36**(3), 126 – 132 (2010)
3. J. Sangai, A. Bellabdaoui, *Workload balancing in nurse scheduling problem models and discussion*, in 2017 International Colloquium on Logistics and Supply Chain Management (LOGISTIQUA), 82-87. IEEE (2017)
4. M. Shahriari, M. Shamali, A. Yazdannik, Iran. J. Nurs. Midwifery Res., **19**(4), 360 – 365 (2014)

5. R. Otero, C. Montoya, C.B. Jaimes, C. Garzón, E. Vergel, J. Valdés, Oper. Res. Health Care., **38**(4), 100389 (2023)
6. A. Ben Said, E. A. Mohammed, M. Mouhoub, *An implicit learning approach for solving the nurse scheduling problem*, in Neural Information Processing. ICONIP 2021. LNCS, 13109. Springer, Cham (2021)
7. A. Youssef, S. Senbel, *A Bi-level heuristic solution for the nurse scheduling problem based on shift-swapping,* in 2018 IEEE 8th Annual Computing and Communication Workshop and Conference (CCWC), 72-78. IEEE (2018)
8. A. R. Clark, H. Walker, J. Appl. Oper. Res., **3**(3), 148 – 162 (2011)
9. I. Berrada, J. Ferland, P. Michelon, A multi-objective approach to nurse scheduling with both hard and soft constraints. Socio-Econ. Plan. Sci., **30**(3), 183 – 193 (1996)
10. B. Cheang, H. Li, A. Lim, Rodrigues, B., Eur. J. Oper. Res., **151**(3), 447 – 460 (2003)
11. H.T. Lim, R. Ramli, *Enhancements of evolutionary algorithm for the complex requirements of a nurse scheduling problem,* in AIP Conference Proceedings, vol. **1635**(1), 615 – 619. American Institute of Physics (2014)
12. N.V. Dharwadkar, G.G. Shingan, S.U. Mane, S. Joshi, Int. J. Swarm Intell. Res., **13**(1), 1 – 17 (2022)
13. G. Çetin, O. Özkaraca, E. Güvenç, M. Sakal, Genetic algorithm-based optimization for nurse scheduling problem. GJES, **9**(4), 31 – 38 (2023)
14. Y.K. Lin, C.H. Yen, Healthcare, **11**(5), 739 (2023)
15. T.H. Wu, J.Y. Yeh, Y.M. Lee, J. Comput. Oper. Res., **54**, 52 – 63 (2015)
16. S. Karmakar, S. Chakraborty, T. Chatterjee, A. Baidya, Acharyya, S., *Meta-heuristics for solving nurse scheduling problem: A comparative study*, in 2016 2nd International Conference on Advances in Computing, Communication, & Automation (ICACCA)(Fall), 1 – 5. IEEE September (2016)
17. B. Kazimipour, X. Li, A.Q. Qin, A *review of population initialization techniques for evolutionary algorithms*, in IEEE Congress on Evolutionary Computation (CEC), 2585 – 2592. IEEE (2014)
18. C.C. Lin, J.R. Kang, D.J. Chiang, C.L. Chen, Int. J. Distrib. Sens. Netw., **11**(7), 595419 (2015)
19. N. Khalili, P. Shahnazari Shahrezaei, A.G. Abri, J. Appl. Res. Ind. Eng., 7(4), 396 – 423 (2020).
20. Andriansyah, N. Alfadilla, P.D. Sentia, D. Asmadi, *Optimization of nurse scheduling problem using genetic algorithm: a case study*, in IOP Conference Series: Materials Science and Engineering **536**, 012131 (2019)
21. E.C. Yagmur, A. Sarucan, J. Intell. Syst., **28**(4), 633 – 647 (2019)
22. K. Leksakul, S. Phetsawat, Math. Probl. Eng., **1**, 246543 (2014)
23. L. Augustine, M. Faer, A. Kavountzis, R. Patel, *A brief study of the nurse scheduling problem (NSP)*. University of Pittsburgh Medical Center. (2009)
24. C.M. Fernandes, N. Fachada, J.L.J. Laredo, J.J. Merelo, A.C. Rosa, Swarm Evol. Comput., **58**, 100721 (2020)
25. J. Zhong, X. Hu, M. Gu, J. Zhang, *Comparison of performance between different selection strategies on simple genetic algorithms*, in International Conference on Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, IEEE. **2**, 1115 – 1121 (2005)

26. B. Maenhout, M. Vanhoucke, Omega, **41**, 903 – 918 (2013)
27. E.K. Burke, A.J. Smith, IEEE Trans. Power Syst., **15**(1), 122-128 (2000)
28. J. Heizer, B. Render, Operations management (8th Edition). (New Jersey: Prentice Hall, 2006)