International Journal of
Mathematical Modelling & Computations
Vol. 15, No. 03, 2025, 101- 129
DOI: 10.71932/ijm.2025.1201016

Nurse Scheduling Problem Optimization based on Water Flow-like, Vibration Damping and Bee Colony Algorithms

Parisa Shahnazari-Shahrezaei[a,*]

[a] *Department of Industrial Engineering, Central Tehran Branch, Islamic Azad University, Tehran, Iran*

Abstract. This research involves a nurse scheduling problem which is formulated as an integer programming model dealing with diverse constraints such as multi-skilled nurses' requirements with different preferences and availabilities plus the hours and shifts related regulations. Due to its complex nature, two new meta-heuristic algorithms, namely, Water Flow-like Algorithm (WFA) and Vibration Damping Optimization (VDO) and another well-known meta-heuristic called Bee Colony Optimization (BCO) are developed. Two problem instances with up to 50 nurses and 35 days are defined considering a real case study via the data extracted from Isfahan-based Sina Hospital's Infant Ward in Iran, the problems are solved to verify the effectiveness of the proposed algorithms and some comparison metrics are applied to evaluate their reliability. As the first problem yielded results displayed, 17 Pareto solutions have been gained for each objective function by the WFA algorithm, 2 Pareto solutions by VDO algorithm and 4 Pareto solutions through the BCO algorithm. And for the second problem, 50 Pareto solutions have been achieved by the WFA algorithm, 6 Pareto solutions by the VDO algorithm, and 9 Pareto solutions using the BCO algorithm. According to the results, the three approaches have the potential to solve the real nurse scheduling problems in large scale optimally (near-optimal) within minutes for both instances. Besides, the reported results show that compared to the other two algorithms, the WFA algorithm discovers the solutions with higher reliability.

Received: 19 March 2025; Revised: 01 May 2025; Accepted: 08 May 2025.

Keywords: Nurse Scheduling; Multi-objective Model; Water Flow-like Algorithm; Vibration Damping Optimization; Bee Colony Optimization.

Index to information contained in this paper

1. Introduction
2. Proposed Nurse Scheduling Problem's Formulation
3. Meta-Heuristic Solution Methods
4. Computational Results
5. Managerial Perceptions
6. Conclusions

1. Introduction

Nurse scheduling process is a type of manpower assignment problem the planning process is under the effect of plenty of conflicting factors. In contrast to production environments which follow standard shifts, hospitals function for an entire 24-hour day while taking on the fluctuations of demand on successive days and shifts. As this problem dictates, hospital managers are obliged to develop an efficient schedule benefitting from the available nurses in the best manner. One of the most challenging problems in hospitals is to come up with the nursing services of high quality at reduced costs. The nurse rostering problem seeks to generate appropriate shift schedules by taking hospital management requirements and individual nurses' preferences as much as possible, to create the opportunity for a safe social life for all nurses and to consider hospital policies, labor laws, and different nursing skill levels while guaranteeing satisfactory coverage at all shifts. However, such factors

*Corresponding author. Email: parisa.shahnazari@iau.ac.ir; parisa_shahnazari@yahoo.com

often contradict each other. Consequently, nurses are not entirely satisfied with their individual preferences and hospital managers end in paying some penalties.

Until now, numerous nurse scheduling models and problem-solving approaches have been designed. Substantial surveys have been performed by Cheang et al. [1], Burke et al. [2], Ernst et al. [3] and Ernst et al. [4] on nurse scheduling problem. Mathematical modeling, meta-heuristics, and AI modeling are three broadly used ones for NSP problem-solving.

In order to come up with an optimal solution for the nurse scheduling process, mathematical models are applied which include mixed linear programming, integer programming, and so on. For solving four nurse roistering benchmark problems, Glass and Knight [5] used a mixed integer linear programming and reduced the problems' solution space.

Due to laborious computations that mathematical programming approaches undergo in real-size problems, meta-heuristic algorithms were built for solving this problem. Despite not being able to generate optimal solutions, meta-heuristics are equipped with the potential to come up with discover the solutions of high quality within a reasonable time period. During the last decade, meta-heuristic algorithms have been applied for nurse scheduling problem-solution process. A novel heuristic algorithm was introduced by Chiaramonte and Chiaramonte [6] using a competitive agent-based negotiation system called competitive nurse rostering (CNR) focusing on nursing personnel's shift preferences. Applying genetic algorithm (GA), Tsai and Li [7] designed a two-stage mathematical modeling for nurse scheduling system. Through utilizing integer programming (IP) and variable neighborhood search (VNS), Burke et al. [8] managed to solve highly-constrained nurse scheduling model. Employing a real case, i.e., a hospital's maternity ward, Shahnazari-Shahrezaei et al. [9] presented a NSP and a differential evolution algorithm (DE) and a greedy randomized adaptive search procedure (GRASP) as two meta-heuristics for the problem-solving process. An Artificial Bee Colony (ABC) algorithm approach was used by Buyukozkan and Sarucan [10] for NSP and its performance was evaluated by real data. In order to deal with nurse scheduling problems a spreadsheet-based two-stage heuristics was applied by Wong et al. [11] in a local emergency department. To deal with a nurse scheduling problem, Constantino et al. [12] proposed a novel deterministic heuristic algorithm known as MAPA for discovering more feasible solutions. Proposing a multi-objective model and a simple flexible heuristic approach, Legrain et al. [13] studied scheduling process for two kinds of nursing teams, i.e., regular teams and float teams. A mathematical programming model was proposed by Jafari et al. [14] considering nurse scheduling problem to maximize the nurses' preferences and to minimize the total extra nurses. Pursuing the goal to minimize demand efficiency, overtime and idle time for scheduling operating room(OR) nurses, Lim et al. [15] proposed a column generation algorithm and a two-stage swapping heuristics using real data and scheduled surgery cases and lunch break for nurses. Proposing a stochastic programming model, Bagheri et al. [16] applied a sample average approximation (SAA)method addressing the real world applications of nurse scheduling problem so that to get the regular and overtime assignment costs minimized considering the sudden changes in nurses' personal preferences and unexpected events. Presenting a new bi-objective approach solving the problem faster and more accurately, Di Martinelly and Meskens [17] considered the entire assignment of nurses to surgical operations while creating teams with strong affinities and tried to minimize nurse idle time. Focusing on a personalized nurse scheduling problem under uncertainty, Legrain et al. [18] developed a robust online stochastic algorithm inspired by the primal-dual algorithm for online optimization and the SAA targeting to create a feasible and near-optimal schedule at the end of the planning horizon. Considering a long term nurse scheduling approach on a linear integer programming and a mathematical model within a specific time horizon, Zanda et al. [19] addressed, the set of feasible scheduling which meet a set of constraints associated with contractual rules, certain local requirements, and various nursing working conditions. Formulating a general integer programming regarding multi-skilled nurses and different real-world constraints spotted in real-case problems and several strategies for nurse re-rostering problem, Wickert et al. [20] designed a variable neighborhood descent heuristic to tackle the problem without using a solver. Hamid et al. [21] developed a multi-objective mathematical model for nurse scheduling using three meta-heuristic algorithms; i.e., multi-objective Tabu search (MOTS), multi-objective Keshtel algorithm(MOKA), and non-dominated sorting genetic algorithm I(NSGA-II) for the problem solution. Combining a mathematical model and self-scheduling for defining nursing shifts towards minimizing the number of shifts and staffing level requirements, Svirsko et al. [22] investigated nurse scheduling models in terms of target performance and their time complexity by the Gurobi optimizer in Python. Using pooling resource concept for allocating medical staff in hospitals and defining their monthly schedules, Chen et al. [23] suggested a two-phase strategy, where the first phase proposed three heuristic algorithms, namely,HRA1,HRA2 and HRA3 for human resource allocation and the second phase applied an improved Particle Swarm Optimization (PSO) algorithm for rational scheduling of the medical staff ; Their research induced results can boost decision making about the medical staff planning and allocation. Taking the preferences of nurses in scheduling shifts into account, Khalili et al. [24] proposed a multi-objective model of nurses' scheduling which focused on their fatigue reduction using a real case of Labbafinejad Hospital. Due to NP-hard nature of nurse scheduling problem, Khalili et al. [25] applied a Gray Wolf algorithm solving the large-sized problem based on designing a new chromosome concerning service level under uncertainty. In order to maximize the nurses' preferences for the shifts and address their preferences' uncertainty, Jafari [26] proposed a mathematical programming model and a fuzzy model based on the Werner's fuzzy. Besides, the researcher evaluated the fuzzy model by some random test problems being solved the fuzzy model and he executed a sensitivity analysis to examine the effects of parameter variations on the results. Pursuing the objective to promote productivity, Benzaid et al. [27] introduced a two-step approach for new patients' treatment scheduling, nurse requirement planning, and the daily patient-nurse allocation by a waiting list and last-minute cancellations. Proposing a particular multi-objective mathematical model for nurse scheduling, Hamid et al. [28] focused on nurses' decision-making habits regarding the human factors and paid attention to some restrictions like individual preferences for days off, the individuals having special conditions including breastfeeding mothers, or those on leave because of a disease or other incidents. Consistent with Taiwan's new labor law affecting nurse scheduling problem, Zhuang and Yu [38] applied a binary goal programming (BGP) model to a case in a hospital where a different old law-abiding model was obeyed and obtained useful concessions for the new law (e.g., how to reduce the tension from the extra overtime evening shifts during scheduling) ;Their comparisons revealed that the new law significantly affected scheduling considering number of working days, nursing workload, and relevant issues. Pursuing the goal to design an automatic nurse roster scheduling system by open-source operational research tools, Leung et al. [39] developed an economical and highly efficient user friendly solution to nurse scheduling system via artificial intelligence (AI) and end user tools. Through introducing an augmented mixed-integer linear programming (MILP) for the NSP, Guo and Bard [40] attempted to minimize the sum of the weighted disclosed demand and nurse preference violations within a month. Benefitting from yearly questionnaire data on work schedule and prescribed hypnotics from 2028 Norwegian nurses taking part in the Survey of Shift work, Sleep and Health (SUSSH), Forthun et al. [41] proposed a longitudinal research and figured out the relationships by a random effects model and a fixed effects regression model; they found out relationship between shift work without nights and non-statistically meaningful sleep medicine use reduction in the fixed effects regression model in comparison with shift work with nights, that is, giving up night shift will enhance sleep and as a result, lower taking hypnotics. Concentrating on two major dimensions of nurse rostering problem (NRP), model and solution method, Turhan and Bilgen [42] suggested a new model considering unit assignments which help the model to be more accurate in terms of real-world scenarios. Finally, they presented a new mathematical based heuristics combining Integer Programming (IP) to create initial schedules and Discrete Particle Swarm Optimization (PSO) to further boost the schedule. In the process, IP corrects any sort of infeasibility. The test data related computational experiments indicated that the proposed algorithm resulted in near optimal solutions. Table 1 displays the previous significant studies. Presenting multiple modified variable neighborhood search (VNS) methods, Chen et al. [43] solved the NSP by applying the greedy concept. In this research, they designed three greedy-neighborhood-swapping mechanisms in order to perform local searches considering one-, two-, or three-neighborhood structures. These mechanisms aimed to minimize soft-constraint violations, such as nurses' preferences while adhering to government and hospital regulations. As revealed by the results, the proposed VNS approaches produced optimal or near-optimal solutions, and the optimal number of neighborhood structures was two. This study verified that increasing the number of neighborhoods does not necessarily improve the ability to escape local optima. Pursuing the aim to introduce a foundational framework for dealing with staff scheduling, Thomas [44] concentrated on staff's shift assignment optimization, a task with complexities as a result of factors like contractual duties and commissioned rest periods; This research also focused on the nurse rostering problem common in healthcare environments undergoing more severe staffing challenges since the COVID-19 pandemic ,which require evaluating staffing needs and optimizing shift allocations for working efficiently under such critical conditions. Identifying various preferences among nursing personnel in terms of shifts, shift rotations, and days off, Okoro et al. [45] proposed a mathematical model based on integer programming for optimizing nursing personnel's satisfaction through prioritizing their preferences; They implemented the model at the Federal University Teaching Hospital called Alex Ekwueme in Nigeria, which resulted in fair and fulfilling shift and day-off allocations consistent with the nursing staff's preferences. Presenting a multi-objective optimization framework integrating multiple perspectives for nurse preferences, Torres Ramos et al. [46] used a Mixed Integer Linear Programming (MILP) model within an ε-constraint method for weighing up the trade-offs in nurse preferences, which is often overlooked in traditional approaches for staff scheduling. Testing their approach on real and synthetic instances led to significant improvements in workload balance and nurse satisfaction and boosted workload balance by 60% for months. Dealing with the challenges behind generating optimal schedules for personal support workers (PSWs)in supported living housing, Ghavampour [47] investigated heuristic approaches and formulated the problem as a Mixed Integer Linear Programming (MILP) model, combining particular objectives and constraints which assign PSWs to patient demands. and applied d heuristic methods such as Logic-Based Benders Decomposition and Lagrangian Relaxation to overcome the problem complexity. As their results indicated, Benders Decomposition outperformed Lagrangian Relaxation, obtaining a solution within 0.2% of the best results yielded by the commercial solver GUROBI.

Table 1: Literature review

References	No	Logic			Objective		Problem	Approach			Method
		Fuzzy	Robust	Gray	Single	Multi		Heuristic	Meta-heuristic	Exact	
Chiaramonte & Chiaramonte	6				*		NSP	*			Agent-based Negotiations
Tsai and Li	7				*		NSP		*	*	GA
Burke et al.	8						NSP			*	Mathematical
Shahnazari et al.					*		NSP	*			Differential
Buyukozkan	10				*		NSP		*		Bee colony
Constantino et al.	12				*		NSP		*	*	Artificial Bee colony
Legrain et al.	13					*	NSP	*			-
Jafari et al.	14	*				*	NSP			*	-
Lim et al.	15				*		NSP			*	-
Martinelly & Meskens	17		*			*	NSP			*	-
Khalili et al.	25			*	*		NSP		*		Wolf Algorithm
Chen et al.	43					*	NSP		*		Variable Neighborhood Search (VNS)
Thomas	44					*	NSP		*		Supply Chain Approach
Okoro et al.	45					*	NSP			*	Integer Programming
Torres Ramos et al.	46					*	NSP			*	Mixed Integer Linear Programming (MILP)
Ghavampour	47						Home Health Care	*			Logic-Based Benders Decomposition
Current study						*	NSP		*		WFA,VDO,BCO

The complexity of NSP as a popular combinatorial optimization scheduling problem is of NP-hard nature [29, 30]. Therefore, applying exact solution methods are not so effective from the computational perspective, the issue which requires using meta-heuristic algorithm for solving the nurse scheduling problem. The current research applies three approximate meta-heuristics, namely, Water Flow-like Algorithm (WFA), Vibration Damping Optimization (VDO) and Bee Colony Optimization (BCO) in solving the given problem. However, the major research gap is as the following:

- Not focusing on nursing staff scheduling for various work shifts so that to come up with balanced work pressure on nurses in providing patients with the medical care of higher quality.

Also, the main contribution of the current research is as the following:

- Proposing an integer mathematical programming as an optimization method for NSP considering the regulations of working hours, work shifts and rest time.

Finally, the main achievement of the research is as the following:

- Determining the working schedule of nurses on work shifts and increasing the quality of services.

The rest of the present article is organized as follows. The proposed NSP's model formulation is described in Section 2. Three efficient approximate meta-heuristics, namely, Water Flow-like Algorithm (WFA), Vibration Damping Optimization (VDO) and Bee Colony Optimization (BCO) are presented in Section 3 for the problem-solving process and the achieved results undergo comparisons. The computational results on problem instances, validation and analysis of the developed solution methods are discussed in Section 4. Eventually, the conclusions are given in Section 5.

2. Proposed Nurse Scheduling Problem's Formulation

2.1. *Problem Definition*

This section presents a scheduling problem which serves to allocate the nurses working in the Infant Ward of Sina Hospital in Isfahan. The considered nurse scheduling model includes three non-equal shifts: 6-h morning shift, 6-h afternoon shift and 12-h night shift. A 35-day scheduling period is taken into account. The nurses are categorized in three levels of proficiency (i.e., advanced practice registered nurse (APRN), regular registered nurse(RN), and nurse practitioner(NP)). In each shift, the number of the required nurses is assigned by supervisor. The number of nurses is assumed to be fixed. In addition, scheduling involves shift allocation regarding nurses' priorities, through which they have the choice to be allocated to work within their actual proficiency level or any other lower proficiency levels, of course not more than one proficiency level simultaneously per shift. Nurses prefer to take a rest on days or shifts which they specify at the beginning of scheduling.

2.2. *System of Notation*

The current study model related notations are as the following:

Indexes

i	Set index for nurses, (i=1, …, I)
k	Set index for days, (k=1, …, K)
j	Set index for shifts, (j=1, …, J); in the study model: (j = 1: *Morning*, 2: *Afternoon*, 3: *Night*)
s	Set index for proficiency levels, (s=1, …, S); in this model: (s = 1: *advanced practice registered nurse*, 2: *regular registered nurse,* 3: *nurse practitioner)*

Parameters

h_{kj}	Length of shift j per day k
$dh_{\min}$	Lower limit for total hours due to be covered by each nurse / day
$dh_{\max}$	Upper limit for total hours due to be covered by each nurse /day
$wh_{\min}$	Lower limit for total hours due to be covered by each nurse / week
$wh_{\max}$	Upper limit for total hours due to be covered by each nurse / week
$fh_{\min}$	Lower limit for total hours due to be covered by each nurse Fridays
$fh_{\max}$	Upper limit for total hours due to be covered by each nurse Fridays
$mh_{\min}$	Lower limit for total hours due to be covered by each nurse during whole scheduling period
$mh_{\max}$	Upper limit for total hours due to be covered by each nurse during whole scheduling period

RN_{kjs}	Total number of required nurses equipped with proficiency level *s* on shift *j*/ day *k*
RSL^i	Actual proficiency level of nurse *i*
Max_Night	Maximum number of night shifts which can be allocated to each nurse to serve during whole scheduling period
$Penl$	A penalty coefficient for serving at a proficiency level lower than actual one
k_i	Set of days when nurse *i* tends to have a rest during some/all shifts
j_{k_i}	Set of shifts of day k_i nurse *i* tends to have a rest
ASL^i_{kjs}	If nurse *i* can be allocated to work compatible with their actual proficiency level *s* or at any other ones lower than that on shift *j* / day *k*, then 1, otherwise 0.
d^i_k	Off-on-off deviation for nurse *i*
d^i_{kj}	Days/shifts deviation for nurse *i* tending to take a rest

Decision Variables

x^i_{kjs}	If nurse *i* is allocated to serve at proficiency level *s* on shift *j* / day *k*; then 1, otherwise 0.
y^i_k	If nurse *i* is allocated to serve for more than 12 h/ day *k*; then 1, otherwise 0.
q^i_k	If nurse *i* is allocated to serve on three consecutive night shifts per days *k-2*, *k-1*, and *k*; then 1, otherwise 0.

2.3. *Objective Functions*

The 1st objective function, Z_1, minimizes total off-on-offs-induced non-conformities for nurses:

$$Z_1 = Min \sum_i \sum_{k=2}^{34} d^i_k \tag{1}$$

The 2nd objective function, Z_2, minimizes total deviations from days/shifts when nurses tend to have a rest:

$$Z_2 = Min \sum_i \sum_{k \in k_i} \sum_{j \in j_{k_i}} d^i_{kj} \tag{2}$$

The 3rd objective function, Z_3, minimizes nurse allocation to proficiency levels lower than their actual one:

$$Z_3 = Min \sum_i \sum_k \sum_j \sum_s [(s - RSL^i) * x^i_{kjs} * Penl] \tag{3}$$

2.4. *Constraints*

Constraints are classified as hard and soft in the study model.

2.4.1. *Hard Constraints*

Lower and upper limit on overall hours served by each nurse / day:

$$\sum_j \sum_s h_{kj} x^i_{kjs} \geq dh_{min} \quad ;\forall i, k \tag{4}$$

$$\sum_{j}\sum_{s} h_{kj}x^{i}_{kjs} \leq dh_{max} \quad ;\forall i,k \quad (5)$$

Lower and upper limit on overall hours served by each nurse / week:

$$\sum_{k=7k_2-6}^{7k_2}\sum_{j}\sum_{s} h_{kj}x^{i}_{kjs} \geq wh_{min} \quad ;\forall i,k_2 \in \{1,2,\ldots,5\} \quad (6)$$

$$\sum_{k=7k_2-6}^{7k_2}\sum_{j}\sum_{s} h_{kj}x^{i}_{kjs} \leq wh_{max} \quad ;\forall i,k_2 \in \{1,2,\ldots,5\} \quad (7)$$

Lower and upper limit on overall hours served by each nurse during entire scheduling period:

$$\sum_{k}\sum_{j}\sum_{s} h_{kj}x^{i}_{kjs} \geq mh_{min} \quad ;\forall i \quad (8)$$

$$\sum_{k}\sum_{j}\sum_{s} h_{kj}x^{i}_{kjs} \leq mh_{max} \quad ;\forall i \quad (9)$$

Lower and upper limit on total hours served by each nurse on Fridays during whole scheduling period:

$$\sum_{k\in 7\cup 14\cup 21\cup 28\cup 35}\sum_{j}\sum_{s} h_{kj}x^{i}_{kjs} \geq fh_{min} \quad ;\forall i \quad (10)$$

$$\sum_{k\in 7\cup 14\cup 21\cup 28\cup 35}\sum_{j}\sum_{s} h_{kj}x^{i}_{kjs} \leq fh_{max} \quad ;\forall i \quad (11)$$

Total number of nurses required at any proficiency level / shift per day:

$$\sum_{i} x^{i}_{kjs} = RN_{kjs} \quad ;\forall k,j,s \quad (12)$$

Each nurse can be allocated to serve compatible with their actual proficiency level or any lower level on a shift per day:

$$x^{i}_{kjs} \leq ASL^{i}_{kjs} \quad ;\forall i,k,j,s \quad (13)$$

Each nurse should not be allocated to serve at more than one proficiency level on a shift per day:

$$\sum_{s} x^{i}_{kjs} = 1 \quad ;\forall i,k,j \quad (14)$$

Highest number of night shifts each nurse can be allocated to serve during whole scheduling period:

$$\sum_{k}\sum_{s} x^{i}_{k(j\in Night)s} \leq Max\_Night \quad ;\forall i \quad (15)$$

Highest number of consecutive night shifts each nurse can be allocated to serve:

$$\sum_{k=k_1}^{k_1+3}\sum_{s} x^{i}_{k(j\in Night)s} \leq 3 \quad ;\forall i,k_1 \in \{1,2,\ldots,32\} \quad (16)$$

After serving within the maximum number of consecutive night shifts, at least 2 days off should be allocated to each nurse:

$$q^{i}_{k} - \sum_{s} x^{i}_{(k-2)(j\in Night)s} \leq 0 \quad ;\forall i,k \in \{3,\ldots,35\} \quad (17)$$

$$q_k^i - \sum_s x_{(k-1)(j\in Night)s}^i \leq 0 \qquad ;\forall i, k \in \{3, \ldots, 35\} \qquad (18)$$

$$q_k^i - \sum_s x_{k(j\in Night)s}^i \leq 0 \qquad ;\forall i, k \in \{3, \ldots, 35\} \qquad (19)$$

$$q_k^i - \sum_s x_{(k-2)(j\in Night)s}^i - \sum_s x_{(k-1)(j\in Night)s}^i - \sum_s x_{k(j\in Night)s}^i \geq -2 \qquad ;\forall i, k \in \{3, \ldots, 35\} \qquad (20)$$

Constraints (17) to (20) guarantee that:
$q_k^i = 1$, if nurse *i* is allocated to serve during three consecutive night shifts on days *k-2*, *k-1*, and *k*; otherwise 0.

Thus, the governing law gets verified through adding the subsequent constraints:

$$\sum_s x_{(k-2)(j\in Night)s}^i + \sum_s x_{(k-1)(j\in Night)s}^i + \sum_s x_{k(j\in Night)s}^i + \sum_s \sum_j x_{(k+1)js}^i + q_k^i \leq 4 \qquad ;\forall i, k \in \{3, \ldots, 34\} \qquad (21)$$

$$\sum_s x_{(k-2)(j\in Night)s}^i + \sum_s x_{(k-1)(j\in Night)s}^i + \sum_s x_{k(j\in Night)s}^i + \sum_s \sum_j x_{(k+2)js}^i + q_k^i \leq 4 \qquad ;\forall i, k \in \{3, \ldots, 33\} \qquad (22)$$

Consider this matter that one nurse cannot be allocated to serve for more than 12 hours successively:

$$\sum_s \sum_{j\in Afternoon} x_{kjs}^i + \sum_s \sum_{j\in Night} x_{kjs}^i \leq 1 \qquad ;\forall i, k \qquad (23)$$

$$\sum_s \sum_{j\in Night} x_{kjs}^i + \sum_s \sum_{j\in Morning} x_{(k+1)js}^i \leq 1 \qquad ;\forall i, k \qquad (24)$$

$$\sum_j \sum_s x_{kjs}^i \leq 2 \qquad ;\forall i, k \qquad (25)$$

A nurse who is allocated to serve for more than 12 hours per day should be allowed to take time off during the following day:

$$y_k^i - \sum_{j\in Morning} \sum_s x_{kjs}^i \leq 0 \qquad ;\forall i, k \qquad (26)$$

$$y_k^i - \sum_{j\in Night} \sum_s x_{kjs}^i \leq 0 \qquad ;\forall i, k \qquad (27)$$

$$y_k^i - \sum_{j\in Morning} \sum_s x_{kjs}^i - \sum_{j\in Night} \sum_s x_{kjs}^i \geq -1 \qquad ;\forall i, k \qquad (28)$$

Consequently, Constraints (26) to (28) guarantee this: $y_k^i = 1$, if nurse *i* is allocated to serve for more than 12 hours / day k; otherwise 0.

Now, the governing law is supported via adding the subsequent constraints:

$$\sum_{j\in Morning}\sum_{s} x_{kjs}^{i} + \sum_{j\in Night}\sum_{s} x_{kjs}^{i} + \sum_{j}\sum_{s} x_{(k+1)js}^{i} + y_{k}^{i} \leq 3 \qquad ;\forall i,k \tag{29}$$

2.4.2. *Soft Constraints*

Each nurse should not be allocated to serve between two days off (off-on-off is not permitted):

$$\frac{\sum_j \sum_s x_{(k-1)js}^{i}}{max(1,\sum_j \sum_s x_{(k-1)js}^{i})} - \frac{\sum_j \sum_s x_{kjs}^{i}}{max(1,\sum_j \sum_s x_{kjs}^{i})} + \frac{\sum_j \sum_s x_{(k+1)js}^{i}}{max(1,\sum_j \sum_s x_{(k+1)js}^{i})} + d_k^i \geq 0 \qquad ;\forall i,k \in \{2,3,\dots,34\} \tag{30}$$

Each nurse is concerned with having a rest on pre-arranged days/shifts:

$$\sum_{s} x_{kjs}^{i} - d_{kj}^{i} = 0 \qquad ;\forall i,k \in k_i, j \in j_{k_i} \tag{31}$$

$$d_k^i \geq 0 \qquad ;\forall i,k \in \{2,3,\dots,34\} \tag{32}$$

$$d_{kj}^i \geq 0 \qquad ;\forall i,k \in k_i, j \in j_{k_i} \tag{33}$$

2.5. *Simplification Steps of Presented NSP*

Certainly, the present study NSP's objective functions and soft constraints can be rephrased as depicted below:

$$Z_1 = Min\sum_{i}\sum_{k=2}^{34}[max\{0,(-\frac{\sum_j \sum_s x_{(k-1)js}^{i}}{max(1,\sum_j \sum_s x_{(k-1)js}^{i})} + \frac{\sum_j \sum_s x_{kjs}^{i}}{max(1,\sum_j \sum_s x_{kjs}^{i})} - \frac{\sum_j \sum_s x_{(k+1)js}^{i}}{max(1,\sum_j \sum_s x_{(k+1)js}^{i})})\}] \tag{34}$$

$$Z_2 = Min\sum_{i}\sum_{k\in k_i}\sum_{j\in j_{k_i}}\sum_{s} x_{kjs}^{i} \tag{35}$$

$$Z_3 = Min\sum_{i}\sum_{k}\sum_{j}\sum_{s}[(s - RSL^i) * x_{kjs}^{i} * Penl] \tag{36}$$

Which are subject to Constraints (4) to (29). Three objective functions considered for the problem are not in the same direction. That is, if one objective function enhancement results in the exacerbation of another objective function; for example, if the rest time during working days and shifts increases based on the first objective function, the compatibility, employment and accessibility of a nurse may not be done according to the second objective function. Also, in this case, a nurse with sufficient skills may not be available at the desired working time and shift. Figure 1 displays the current research framework.

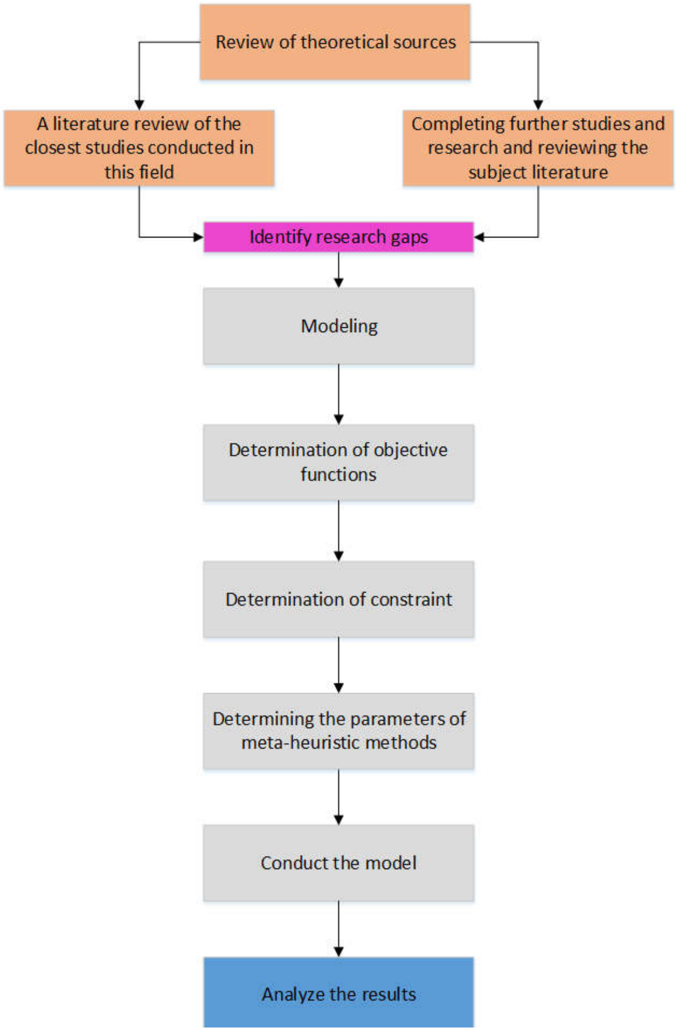

Figure 1. Research framework

2.6. *Preliminary Definitions*

- Pareto Set (Archive)

As Pareto criterion states: it is impossible to make anyone better –off without making someone worse -off. Thus, in case none of the feasible solutions leads to the improvement of any of the objective functions without the exacerbation of at least one of the other objective functions, then a non-dominated or Pareto optimal solution is feasible in the design space. To sum up, Pareto set contains all non-dominated feasible solutions The set of feasible solutions that are non-dominated is also known as Pareto optimal or non-dominated set. If a solution exists which does not belong to this set, it is called a dominated solution [41].

- Non-dominated solutions

A non-dominated solution is the one in which no one objective function can be enhanced without a simultaneous detriment to at least one of the other objectives [42].

3. Meta-Heuristic Solution Methods

In this research, three novel and well-known meta-heuristics, namely, Water Flow Algorithm (WFA), Vibration Damping Optimization (VDO) and Bee Colony Optimization (BCO) have been applied to optimize and solve the problem on a large scale. The proximity of the solution method of these three algorithms to each other is the reason for their applicability here. In fact, all of these three methods are based on random answer generation and neighborhood search. For example, the VDO method starts the move from a random initial solution and produces a random solution in each iteration and uses probability rules to search the neighborhood. Also, BCO algorithm is based on a local search algorithm. Finally, Water Flow Algorithm simulates how water flow moves to other watersheds. In such a way that by determining the local answers that include pits filled with water, it overflows and the water flow does not approach the water pits from a certain distance from the neighborhood. Rather, they move in new directions. In fact, according to this algorithm, duplicate answers are avoided.

3.1. *WFA*

Inspired by the shape of water flow in nature, WFA [31] is an optimization algorithm in which solution agents, solution space and objective function are considered as water flows, a territory and altitude of a flow (traversing the terrain mapped from the objective function), respectively. Flows' movements represent location changes directed by gravitation force and energy conservative law. Searching for solution begins with one water flow as a single agent and then that flow will be split into multiple sub-flows when passing through rough territories. In contrast, several flows will merge into one flow when meeting each other at the same location. Hence, those flows (simulated solution agents) vary dynamically for seeking optimal solutions. In this respect, Yang and Wang [31] applied WFA for bin packing problems. Wu et al. [32] also designed a heuristic algorithm using the WFA logic to solve the cell formation problem. In the continuation, the initial solution generation method, the operations of WFA based on several natural properties and behaviors of water flows and the proposed WFA's structure are detailed thoroughly.

3.1.1. Initialization Method

WFA begins with only one water flow (a single solution agent) whose location is created randomly. To generate the initial solution, the assignment starts from the first shift of the first day of scheduling horizon as shown below:

```
{
 For k = 1 to Max-k (e.g., 35 days)
    For j = 1 to Max-j (e.g., 3 shifts)
       For s = 1 to Max-s (e.g., 3 skill levels)
          Equal to the demand of shift j / day k for any proficiency level s, nurses are
          selected randomly out of those able to operate from feasibility perspective.
          The first priority is for nurses whose actual proficiency level is more tailored
          to the skill necessary for shift j / day k and the second priority is for nurses
          interested in working in shift j / day k.
       End for
    End for
 End for
}
```

3.1.2. *Splitting and Moving Operations of Flow*

In WFA, at first there exists only one water flow whose location is produced randomly and it starts flowing to new locations under the fluid momentum and potential barrier for searching the solution space seeking novice and better solutions. Two cases may happen while flowing: When meeting rough terrains, the water flow splits into sub-flows which are located adjacent to the initial water flow, or it keeps going as a single flow toward the best feasible adjacent location. A flow of higher momentum splits into more sub-flows to traverse more area. No flow splitting occurs to a flow of zero momentum ,which is regarded stagnant.

Take *N*, W_i and V_i as the number of water flows in the current iteration, the mass and velocity of flow *i* (*i*=1, 2, … , *N*), respectively. n_i represents the number of sub-flows branched from flow *i* specified by its momentum, W_iV_i. A flow can be split into sub-flows only when its momentum surpasses a predetermined definite limit, *T*. In order to avoid the number of sub-flows increasing in an exponential manner after proceeding WFA, an upper limit $\bar{n}$ is considered for the number of sub-flows branched from a flow in any iteration:

$$n_i = min\{ max\{ 1, int(\frac{W_iV_i}{T})\}, \bar{n}\} \tag{37}$$

According to Equation (37), if the momentum of a flow ranges from 0 to *T*, it will not split into sub-flows and will be considered a single one getting to be located somewhere new. Having generated sub-flows, the initial flow gets disposed of.

While exposed to splitting into n_i sub-flows, mass of flow *i* is distributed to sub-flows with respect to their ranks (*r*). Mass of sub-flow *k* split from flow *i* is calculated by Equation (38):

$$W_{ik} = (\frac{n_i+1-k}{\sum_{r=1}^{n_i} r})W_i, \qquad k = 1,2,\dots,n_i \tag{38}$$

Sub-flows related velocity is measured via energy conservation equation. To compute the velocity of sub-flow *k* split from flow *i*, Equation (39) is used:

$$\mu_{ik} = \begin{cases} \sqrt{V_i^2 + 2g\delta_{ik}} & if V_i^2 + 2g\delta_{ik} > 0 \\ 0, & otherwise \end{cases} \tag{39}$$

Where, *g* stands for gravity acceleration and δ_{ik} shows objective value changes from solution *i* to its adjacent solution, i.e., *k*. If $V_i^2 + 2g\delta_{ik} < 0$, then sub-flow *k* will get stagnant in its existing location without undergoing any splitting and moving; equivalently, the solution gets stuck in a local optimum. The stagnant sub-flow *k* merges into other sub-flows and is carried away or gradually evaporates into the air.

In the present study's WFA, a multi-objective multi-operator search procedure [33] is used to find neighboring solutions. The general structure of multi-objective multi-operator search procedure is as the following:

{Multi-objective multi-operator search structure
For each input solution *x*
  Set $P_{approx} = \{x\}$
  Repeat
    Randomly choose some *x* in P_{approx} subject to *nh(x)* not examined so far
      For all $NR = \{nh_1, nh_2, \dots, nh_r\}$
        Create $nh_i(x)$
        Apply feasibility check procedure on $nh_i(x)$
      End For
    Update P_{approx} with $nh_i(x)$
    If $x \in P_{approx}$
      Mark *x* as investigated
    End if
  Until the time when no uninvestigated element remains in P_{approx}

End for
Return P_{approx}
}

Where P_{approx} , *NR*, nh_i and $nh_i(x)$ stand for a set of solutions, a set of neighborhood search operators, the i^{th} neighborhood search operator and the i^{th} neighboring solution of x obtained by nh_i. In the aforementioned procedure, a set of solutions (P_{approx}) is constructed which solely includes the initial solution at first. An iteration involves randomly selecting an uninvestigated element of P_{approx} and generating its neighboring solutions by four neighborhood search operators (nh_1, nh_2, nh_3, and nh_4). Next, P_{approx} is updated by non-dominated relations and this process is repeated until the investigation of all elements ofP_{approx}gets over. Finally, P_{approx} including local optimal solutions in the adjacent initial solution returns. This procedure's induced output is a set of neighboring solutions from the initial solution, which are sub-flows split from the original flow. You can observe the four neighborhood search operators applied in the proposed WFA below:

- nh_1: Randomly nominating a nurse and in case of being allocated to serve in some shifts/days they tend to take time off, removing their allocation in the considered shift/day and replacing an eligible nurse.
- nh_2: Randomly nominating a nurse and checking their allocations in all shifts. In case of allocating the considered nurse to serve at proficiency level lower than the real one on a shift, removing their allocation in the considered shift and replacing an eligible nurse in the same shift if possible, or substituting their allocation with a nurse allocated to serve at a proficiency level closer to the required skill in the same shift.
- nh_3: Removing the allocation of nurses being allocated to serve between two off-days and substituting them with eligible nurses if possible.
- nh_4: Removing some allocations of nurses with the highest weekly work hours and replacing the nurses with the lowest weekly work hours regarding the feasibility conditions.

3.1.3. *Merging Operation of Flow*

When over two water flows transfer to similar location, they merge into a single flow of bigger mass and momentum and the adjacent solution search is reinforced. Due to this merging, stagnant flow gains enough energy to release itself from getting stuck. Flow merging operation regularly examines if a flow shares similar location with others; in case it is true, the next-coming flow merges into the previously-coming one. Take flows *i* and *j* as the ones sharing similar location, then flow *j* gets excluded when merging into flow *i*. Flow *i*'s mass and velocity are updated by Equations (40) and (41):

$$W_i = W_i + W_j \tag{40}$$

$$V_i = \frac{W_i V_i + W_j V_j}{W_i + W_j} \tag{41}$$

This operation prevents redundant searches by merging solution agents which have the same objective value.

Since the presented NSP is a multi-objective model, non-dominated relations will be checked when over two flows proceed to the very same location. Afterwards, the flow (i.e., solution agent) of the highest diversity [34] is selected, others are merged into it and mass and velocity updating is performed. As a matter of fact, all solution agents with the same quality are reduced to one solution agent.

3.1.4. *Water Evaporation Operation of Flow*

Naturally, part of water evaporates into the air and returns to earth through precipitation after a predetermined number of iterations. The earth-bound locations are stochastically specified from the locations of current flows. This process can help a solution agent not to get stuck in a local optimum and to search more solution spaces. In this operation, all flows' masses are updated according to Equation (42):

$$W_i = (1 - \frac{1}{t})\bar{W}_i, \quad i = 1,2,\ldots,N \tag{42}$$

In which, $\bar{W}_i$ shows the mass of flow i when initially generated or when merged with others. Whenever evaporation occurs, all flows' masses are reduced by the ratio of $\frac{1}{t}$. Consequently, in case a flow does not undergo merging or splitting, its total mass is reduced after *t* iterations.

3.1.5. *Precipitation Operation of Flow*

Once water evaporates into water vapor, it returns to earth through precipitation when vapor reaches a given level. WFA involves two types of precipitation: enforced and regular, where in the former type of precipitation, once all flows get located on earth with zero velocities (stagnant flows), all of them are forced to evaporate into the air and then return to earth. Poured-down flows equal the evaporated flows in number, but the locations they return to are stochastically deviated from the original ones. The initial mass of the original flow, W_0, is proportionally distributed to poured-down flows and the initial velocity, V_0, is assigned to them according to Equations (43) and (44):

$$W_i' = (\frac{W_i}{\sum_{k=1}^{N} W_k})W_0 \tag{43}$$

$$V_i^{'} = V_0 \tag{44}$$

And the latter type of precipitation as the regular one is performed in every *t* iteration to move the evaporated water back to earth. New poured-down flows join the current ones, which leads to increasing the current solutions in number. In this paper, *N* new flows pour down to earth and increase current flows to 2*N*. For this purpose, one of the neighborhood search operators ($nh_1, nh_2, nh_3,$ and nh_4) is chosen randomly and is applied on one of the *N* current solutions to generate a new solution (a non-repetitive solution agent adjacent to the current solution). This process is repeated *N* times to achieve *N* new solutions. The masses of poured-down flows are determined by Equation (45) and their velocity is initialized with the initial velocityV_0:

$$W_i^{'} = \frac{W_i}{\sum_{k=1}^{N} W_k}(W_0 - \sum_{k=1}^{N} W_k \tag{45}$$

Where, $W_0 - \sum_{k=1}^{N} W_k$ is the cumulative mass of evaporated water.

Since some new flows might be produced in the same locations by the two mentioned types of precipitation, in order to get redundant flows discarded, a flow merging operation is run following precipitation operation.

3.1.6. *Solution Feasibility Assessment*

Implementing the proposed WFA algorithm results in the generation of new solutions, for which a feasible solution tests the satisfaction of all constraints for the developed solution and converts it into a feasible one in case of some constraints being violated. This procedure is given below:

Step1. If nurse *i* has been allocated to serve for over 12 hours successively or after operating in the highest number of night shifts consecutively, some of their allocations are cancelled regarding the violated constraint and then the cancelled assignments in the considered days, shifts and proficiency levels are modified consistent with demand level.

Step2. If nurse *I* has been allocated to serve at proficiency level s higher than their real level of proficiency:

Step2.1. If the number of assignments at s exceeds the demand level for that shift, the assignment of nurse i will be cancelled.

Step2.2. Else, if the number of assignments at s falls behind or equals the demand level for that shift, considering the satisfaction of the rest constraints, a nurse is selected and assigned to work out of the nurses able to work at s but not assigned in the given shift and then the assignment of nurse i will be cancelled. Otherwise, if a nurse has been allocated to serve at real proficiency level of nurse i or lower than the real proficiency of nurse i on the same shift, and also able to work at s, their allocation will be substituted with nurse i, provided that the rest constraints are satisfied. If none of the mentioned changes are possible, the allocation of a nurse able to serve at s but scheduled to serve in the previous or next shift, and the like, will change in order to allocate to serve at s in the considered shift if possible, and then the allocation of nurse i will be called off.

Step3. If the demand level for proficiency level s has not been taken into account for a shift:

Step3.1. If the number of allocations exceeds the demand level for the considered shift, some allocations will be cancelled until the equality relation is held.

Step3.2. Else, if the number of allocations is less than the demand level for the considered shift, some nurses are selected and assigned to work regarding all constraints. If no one is found to allocate, the demand level will be satisfied by changing the schedule of the previous or following shifts/days preferably.

Step4. If nurse i has been allocated to serve on a shift or shifts of a certain day while allocated to serve for over 12 hours during the previous day, efforts are made to cancel all their allocations in the given day and other nurses are allocated to serve that day regarding the feasibility conditions. If not possible, their allocation will be cancelled for one shift of the previous day and another nurse is allocated to work on the certain shift considering the feasibility conditions.

Step5. If weekly work hours of nurse i are less thanwh_{min}, allocating some nurses in the considered week is called off and nurse i will be allocated instead if possible. If weekly work hours of nurse i exceed wh_{max}, some allocations of nurse i in the considered week get cancelled and some nurses assigned to carry out some shifts of the given week are allocated instead if possible. These changes are performed concerning the feasibility conditions.

About the remaining constraints and their nature, nurses are selected this manner and the schedules are modified until the time when all constraints are fulfilled. For a constraint, if it is not possible to select a nurse considering all circumstances, it is impossible to convert its related solution into a feasible one and the relevant operation gets over.

3.1.7. *Procedure of Pareto Archive Update*

Pareto archive is adopted to store the non-dominated solutions. An iteration involves comparing the existing solutions in Pareto archive with the produced solutions in the current iteration via non-dominated solutions, which are considered as a new archive.

3.1.8. *WFA's Suggested Structure*

Based on the above-mentioned cases, the presented WFA's pseudo-code is as follows:

{

Procedure WFA

WFA parameter setting: the iteration limit G, the initial mass W_0, the initial velocity V_0, the basis momentum T, the number of the initial branch *N*=1, and the iteration count

Q=0. Initialize Pareto archive as an empty set.

While Q<=G
For each flow $i \in \{1,2,...,N\}$
Calculate the number of sub-flows n_i based on Equation (37).
Apply splitting and moving procedure for creating n_i sub-flows.
For each sub-flow $k \in \{1,2,...,n_i\}$
Calculate the mass of sub-flow W_{ik} split from flow *i* using Equation (38).
Calculate the velocity of sub-flow μ_{ik} split from flow *i* using Equation (39).

End for
End for
If flows have the same solutions,
Then execute flow's merging operation and have the subsequent mass and velocity updated based on Equations (40) and (41).
End if
Run water evaporation operation and get the subsequent mass updated for each water flow based on Equation (42).
Have the number of sub-flows updated for each flow *i*.
Perform the updating of total number of water flows: $N = \sum_{i=1}^{N} n_i$.
If precipitation condition is met, **then**
For each flow i
Figure out poured-down flows' mass ratios based on Equation (43) or (45) considering precipitation types and consider$V_i^{'} = V_0$.
If flows have identical solutions, **then**
Execute flow's merging operation and get the subsequent mass and velocity updated according to Equations (40) and (41).
End if
Run the updating of the total number of water flows *N*.
End For
End if
Update Pareto archive.
Q=Q+1
End While
Return Pareto archive.
End WFA
}

Pursuing the goal to investigate the quality of the generated solutions by the proposed WFA, VDO and BCO algorithms are also utilized to solve the given NSP. In the continuation, the suggested structure of VDO and BCO algorithms is addressed.

3.2. *VDO Algorithm*

The performance of Vibration Damping Optimization algorithm presented by Mehdizadeh and Tavakkoli-Moghaddam [35] is based on vibration damping features. The general structure suggested for this algorithm is illustrated in the following pseudo-code which has been fully detailed.

3.2.1. *VDO's Proposed Structure*

{Initializing:
Generate initial feasible solution.
Initialize algorithm parameters as A_0, l_{max}, γ, t_{max}, δ and t=1.
Initialize Pareto archive that involves the initial feasible solution produced in the first step.
For t=1 to t_{max}

Calculate objective function's value and name it u_0.
For l=1 to lmax
Apply neighborhood search operator on the existing current solution so that to produce a new solution.
Calculate the value of new solution's objectives function and name it u.
Apply acceptance procedure on current and new solution.
Update Pareto archive.
End for (internal loop).
Update amplitude for current algorithm iteration.
End for (external loop).
Return Pareto archive.
}

The vibration damping algorithm starts with a feasible solution and produces a solution in each replicate according to its previous solution. In the present study, considering the three-objective mathematical model, each replicate of VDO algorithm involves dealing with a set called Pareto Archive, keeping Pareto Boundary solutions, and reporting them as output.

3.2.2. *Solution Representation*

Since all meta-heuristic algorithms require feasible solution at the beginning, a feasible solution based on a certain structure known as Solution Representation is essential.

Regarding the considered problem, a feasible solution is represented as a 2-D matrix in which rows equal the number of nurses and columns equal the multiplication of the number of days and the number of shifts. In each cell of the above mentioned matrix, either 0 is placed (i.e., the nurse is not allocated to serve on that day and shift) or a number greater than or equal to that nurse's proficiency level, indicating the proficiency compatible with which the nurse is allocated to serve on that day and shift.

3.2.3. *Initial Solution Generation*

For generating an initial solution, here allocation starts from the 1st day and 1st shift. In each day and shift for each proficiency level, a nurse is selected out of the ones with the potential to be allocated. In the defined nurse selection procedure, the first priority is given to the nurses whose proficiency level is close to the required level and the second is given to those unwilling to work.

3.2.4 .*Initial Solution Selection*

As observed in the proposed structure of the algorithm, it initiates with the initial feasible solution, where first a number of certain feasible solutions are generated and then out of them, the solution of the highest quality and highest diversity is picked as the algorithm's initial solution.

Having generated a set of the initial solutions, all solutions' fitness is computed and the solution having the highest fitness is chosen. In order to compute fitness, first based on the non-dominated sorting-oriented MOEA known as non-dominated genetic algorithms II presented by Deb et al. [33], ranking all solutions is done and then the total value of crowding distance is calculated for each solution in its front. Finally, each solution's fitness calculation is performed by the following equation:

$$C_s = \frac{rank}{crowding\ distance} \tag{45}$$

In the above equation, *rank* is the frontier number where each solution is located. After $C_s C$ index computation for the solutions, the solution including the min $C_s$value is chosen as the one of THE highest quality and highest diversity.

3.2.5. *VDO Algorithm Parameters' Initialization*

Similar to other meta-heuristics, VDO algorithm has its own special parameters required to be initialized at the beginning, including vibration field A_t, maximum replicate per vibration field l_{max}, damping coefficient γ, standard deviation δ and maximum replicate of the main loop t_{max}.

As stated in the proposed structure, the algorithm has two loops, the inner (i.e., vibration field) and the outer loop. After the algorithm parameters' initialization, the outer loop of the algorithm starts. This loop has an inner loop. In each replicate of the algorithm (i.e., outer loop), the objective function's calculation of the current solution is performed and then in the inner loop, the current solution is exposed to neighborhood operator and the solution acceptance conditions are investigated. If the solution produced by neighborhood operator is eligible, it is accepted and is substituted with the current solution. Otherwise, it is rejected. After the inner loop termination, the vibration field is updated and the outer loop is replicated. In the following, the neighborhood operator performance, the acceptance conditions and the vibration field updating procedure have been described.

3.2.6. *Neighborhood Search*

The present study neighborhood search operator is inspired by Variable Neighborhood Search (VNS). Three neighborhood structures have been designed combined with VNS's structure, whose details are given in the following:

1st NS: This operator works this way: out of the nurses available, one is selected randomly and efforts are made to call off the selected nurse's allocations in the days when willing to have time off and to assign those allocations to the nurse able to work.

2nd NS: This operator works this way: a nurse is selected randomly and given all shifts, if the mentioned nurse has been allocated to a proficiency level lower than their real proficiency level, efforts are made to have the nurse's allocation closer to the actual one, if possible, by either cancelling the selected nurse's allocation and offering it to the eligible nurses or exchanging it with that of the nurse allocated to proficiency level closer to the selected nurse's proficiency level.

3rd NS: This operator works based on the weekly hours' number. Efforts are made to cancel the nurses' allocation with the max working hours and instead, the nurses with the min working hours be given the allocation subject to observing the feasible conditions.

As explained before, the three neighborhood structures have been combined as Variable Neighborhood Search (VNS) structure as follows:

```
  {For each input solution as s
K=1
  While the termination criterion is met do
      S1=Apply NSS type k
      S=Acceptance method (S, S1)
      If s is improved, then
      K=1
  Else
K=k+1
  If k=3 then
K=1
  End if
  End while}
```

By considering the aforementioned structure, having the solution exposed to NS, out of the two solutions at hand, one is picked as VNS's subsequent replicate solution. The solution goes through this selection process that according to the non-dominated equations, the dominant solution is chosen.

3.2.7. *Acceptance Requirements*

As stated earlier, after applying the neighborhood operation on the current solution and generating a new neighborhood, out of the two solutions, one is required to be picked as the current solution. For this purpose, VDO algorithm is subject to the acceptance requirements. Suppose that the values of the current solution's objective functions are represented as U_0 and those of the new solution's objective functions as U, then, the acceptance requirements will be as follows:

$$\Delta = \begin{cases} \Delta = 0 & if\ U\ dominate\ U_0 \\ \Delta = 1 & if\ U0\ dominate\ U \end{cases} \tag{46}$$

1. If $\Delta = 0,$ a new solution is selected to be substituted with the existing one.

2. If $\Delta = 1$, the value $r \in (0,1)$ is randomly generated, then if $r < 1 - \exp\left(-\frac{A^2}{2\delta^2}\right)$, the new solution is verified to be substituted with the existing solution. Otherwise, such solutions are taken as non-dominated. Thus, for both current and new solutions, Euclidean distance sum considering the Pareto archive included solutions is computed and the solution with min Euclidean distance is chosen.

3.2.8. *Vibration Field Update*

In each of the main replicates of the algorithm (i.e., outer loop), after the internal replicates termination, the vibration field value is updated. In fact, this value is used to reduce the vibration or oscillation field. The updating is done as the following:

$$A_t = A_0 \times \exp\left(-\frac{\gamma t}{2}\right) \tag{47}$$

Where, A_0 stands for the initial value of vibration field and γ is the coefficient of damping.

3.3. *BCO Algorithm*

Swarm algorithms are very effective in solving multivariate optimization problems. Bee Colony Optimization (BCO) algorithm was presented by Pham et al. [36] as a newly emerging one mimicking the behavior of honeybee for searching food.

3.3.1. *BCO Algorithm's Structure*

The present study BCO algorithm's proposed structure follows the pseudocode displayed down:

{Initialization:
  Initialize the algorithm parameters (population size: *PS*, the percentage of the 1st swarm: *PER*, number of the algorithm replicates: R_{max}).
  Initialize Pareto archive as empty set.
  Generate *N* feasible solutions as initial population.
    While criterion is met
      Compute each solution's fitness in current population.
      Choose best bees and their location as p1 set.
      Choose other bees and their location as p2 set.
      Utilize neighborhood search operator for p1 set,

Utilize feasibility check method for yielded solutions.
Allocate some bees to yielded solutions and compute their fitness.
Utilize random neighborhood search operator for p2.
Utilize feasibility check for yielded solutions.
Compute their fitness.
Choose *N* best bees per location as next generation's population.
Update Pareto archive.
End while
Return best solution.
}

BCO algorithm is supposed to consist of two swarms of bees, in which the 1st swarm is intelligently looking for food sources (i.e., locations or solutions) and the 2nd swarm of the bees do this task randomly. In the 1st stage of above structure, the algorithm parameters are initialized, each swarm bees' number is defined and the initial solutions' population is generated. In the loop related to the body of the algorithm replicated in a certain number, the fitness function is computed for the solutions (i.e., food sources) and the best locations are allocated to the 1st swarm represented with p1 and the other locations to the 2nd swarm (i.e., p2).

After allocating the locations to the bees, the designed Neighborhood Structures, described in the continuation, are applied on each swarm's solutions and a number of the solutions (i.e., food location or source) are generated. Out of the existing solutions, *N* superior solutions are chosen as their population of subsequent replicate of the body of the algorithm. Moreover, in this structure, the Pareto archive as a set of the Pareto frontier solutions is defined and is updated at the end of each algorithm replicate and ultimately, is reported as the algorithm output.

In above structure, criterion C_S described in VDO algorithm is used for computing fitness. It is worth mentioning that this algorithm follows the solution representation and the initial solutions generation similar to that of VDO algorithm.

Employing BCO, the problem here is solved based on local search through a new method the input of which is the swarm of p1 set solutions. The utilized method operates based on neighborhood search. In other words, it receives a set of solutions as input and tries to come up with good neighborhood solutions by boosting each of these solutions.

In developing the present study method mentioned above, a Variable Neighborhood Search (VNS) has been used. In fact, each of the solutions existing in p1 swarm is given as input to VNS's structure and a better solution is produced. The VNS's structure used in this section is the same elaborated in VDO algorithm section.

3.3.2. *Random Neighborhood Search (p2 Set of Swarm of Bees)*

For implementing Random Neighborhood Search for the 2nd swarm, a Parallel Neighborhood Search Operator has been employed. Parallel search structure results from simultaneously applying 4 described neighborhood search structures on the input solution as illustrated below:

Step 0: Start
Step 1: Set counter as 0.
Step 2: Apply 1st, 2nd, 3nd and 4nd NSs on input solution sequentially and name their outputs as s1, s2, s3 and s4, respectively.
Step 3: Out of 4 solutions s1, s2, s3 and s4, select best solution and substitute it with input solution.
Step 4: Add a unit to counter.
Step 5: If counter has reached acceptable limit, proceed to Step 6, otherwise return to

Step 2.
Step 6: End.

It is worth stating that in Step 3 of this structure, the best solution is the one with the highest quality and max diversity. The criterion for quality here is the solution being non-dominated and that for diversity is the Euclidian distance from the best solution yielded through the algorithm until now. Indeed, this step involves selecting a non-dominated solution with the highest Euclidian distance from the best one.

3.3.3. *Pareto Archive Update*

As it has been already discussed, the solution approach of the current research pursues Pareto Archive, which performs storing the non-dominated solutions the algorithms produced and it gets updated in each replicate of the algorithm. The update process goes this manner: the solutions generated in the current replicate and the solutions existing in Pareto Archive are added to a pool of solutions and ranked with respect to each other. Then, out of these solutions, the ones available in the 1st rank (i.e., frontier) dominating the solutions of other frontiers are nominated as the new Pareto archive.

3.3.4. *Solution Choosing*

In each replicate, an algorithm requires solutions' population. For choosing the subsequent replicate population, the solutions existing in the population of the current replicate and the solutions generated by the algorithm so far are added to the solution pool together. Afterwards, consistent with the rule stated by Deb et al. [33], the crowding distance is ranked and computed for each solution regarding that solution's rank is done. At last, N solutions of the highest quality and max diversity are chosen as the subsequent replicate population of the algorithm.

4. Computational Results

To analyze the effectiveness of the presented NSP formulated in line with a real case study, which has been carried out in the Infant Ward of Isfahan-based Sina Hospital in Iran, two numerical examples get solved through the presented WFA, BCO and VDO algorithms on an Intel(R) Core(TM) i3 of 4 GB of memory and 2.27 GHZ of dual core CPU and the MATLAB R2009a software. A scheduling period of 35 days (5 weeks) is taken into account.

The common features of both problem instances are shown by Table 2. Tables 5 to 9 display the remaining features of both problem instances. The input parameters of three algorithms, namely, WFA, BCO and VDO are given in Table 10.

Table 2. Common features of two defined examples based on the case study

dh_{min}	dh_{max}	wh_{min}	wh_{max}	fh_{min}	fh_{max}	mh_{min}	mh_{max}	Max_Night	$Penl(\$)$
0	18	24	90	0	90	128	252	20	10

Tables 3-7 display the predetermined parameters based on the current situation in Sina Hospital. In Table 3, the number of nurses is given based on their actual proficiency levels for the two considered problem instances. Table 4 lists the number of nurses required for each work shift. Also, Tables 5, 6, and 7 demonstrate nurses' rest time for the first and second problems, respectively. Eventually, Table 10 displays the implementation parameters for meta-heuristic methods.

Table 3. Actual Proficiency Level of Nurses

Real proficiency level	Nurses' ID (Problem instance 1)	Nurses' ID (Problem instance 2)
APRN	1-8	1-20
RN	9-14	21-35
NP	15-20	36-50

Table 4. Number of Required Nurses per Shift of Scheduling Period

No. of Required Nurses	APRN	RN	NP
Problem Instance 1	Morning: 2 Afternoon: 2 Night: 2	Morning: 2 Afternoon: 2 Night: 2	Morning: 2 Afternoon: 2 Night: 0
Problem Instance 2	Morning: 6 Afternoon: 6 Night: 4	Morning: 4 Afternoon: 4 Night: 4	Morning: 4 Afternoon: 4 Night: 4

Table 5. Preferred Days to Take a Rest (Problem Instance 1)

Nurses' ID	Preferred days	Nurses' ID	Preferred days
1	4,7,28	11	6,23,24
2	1,3,20,21,32	12	9,10,20,21
3	2,10,11,25,26,34	13	13,14,25,31
4	8,9,27	14	15,22,23
5	5,6,12,13,18,19	15	1,2,20,27,28,35
6	14,15,23,30	16	7,8,26,35
7	16,17,24,35	17	9,10,25
8	2,9,22	18	6,11,12,30
9	1,2,28,30	19	13,14,15,16
10	3,16,17,33	20	21,22,23,32

Table 6. Preferred Days to Take a Rest (Problem Instance 2)

Nurses' ID	Preferred days	Nurses' ID	Preferred days	Nurses' ID	Preferred days
1	17,18,24,25	11	7,8,28,29,30	21	23,24,25,26
2	1,2,4,23	12	9,13,15,19,20	22	7,22,23
3	5,6,19,28,29,30	13	1,3,31,32,33	23	1,2,3,15,16,17,18
4	3,21,22,23	14	5,6,21,22	24	4,20
5	1,2,14,17,25,27,28	15	4,5,6,11,12	25	9,10,11,12
6	9,10,11,34,35	16	18,26	26	5,6,12,13
7	12,13,14,15	17	16,17,18	27	14,17,18,19
8	25,26	18	25,26	28	26,27,28
9	8,12,13,14	19	23,27,28	29	4,5,32,33
10	18,19,20	20	2,10,23,24,25	30	23,24,27,28

Table 7. Preferred Days to Take a Rest (Problem Instance 2)

Nurses' ID	Preferred days	Nurses' ID	Preferred days
31	1,7,8	41	21,22,23
32	2,3,6,7,8	42	26,27,28
33	17,18,19	43	25,26,27
34	19,21,22	44	9,10
35	10,11,13,14	45	15,16,17
36	3,4,34,35	46	4,5,21,22
37	10,25,26	47	1,2,3,4,35
38	17,18,19	48	23,28,29
39	2,3,4,5	49	13,14,17,18
40	1,8,9,10	50	3,4,5,34

To define the required pre-parameters as indicated in Table 10, first off, for each meta-heuristic method, a suitable three-level design is considered for each experiment by the Taguchi method. Then, by running each experiment and measuring the MID index, signal-to-noise ratio (SNR) is calculated for each parameter. Table 8 shows the MID value of each experiment and Table 9 depicts SNR of each parameter level. It should be mentioned that the lower the SNR, the better level selected for the parameter. Also, level (1) is the lowest value, level (2) is the middle limit and level (3) is the highest acceptable value taken for each parameter.

Table 8. MID index for each meta-heuristic method

Run	MID_{WFA}	MID_{BCO}	MID_{VDO}
1	0.541	0.632	0.701
2	0.361	0.298	0.369
3	0.556	0.432	0.601
4	0.444	0.398	0.564
5	0.398	0.588	0.401
6	0.601	0.574	0.465
7	0.598	0.439	0.369
8	0.365	0.333	0.369
9	0.457	0.401	0.367

Table 9. SNR

S/N	G	W_o	v_o	T	g	t	$\bar{n}$	Ps	Per	Ao	Imax	λ	δ	tmax
Level (1)	0.15	0.32	0.39	0.78	0.34	0.88	0.37	0.78	0.25	0.36	0.66	0.41	0.74	0.66
Level (2)	0.26	0.29	0.25	0.66	0.49	0.90	0.26	0.85	0.22	0.44	0.78	0.36	0.84	0.74
Level (3)	0.36	0.38	0.33	0.53	0.43	0.95	0.56	0.92	0.32	0.56	0.88	0.44	0.78	0.81

Table 10. Input parameters for Proposed WFA, BCO and VDO

Algorithm	Parameter	Value	Parameter	Value
WFA	G	100	g	10
	W_0	40	t	20
	V_0	15	$\bar{n}$	10 for Problem instance 1 5 for Problem instance 2
	T	100		
BCO	PS	200	R_{max}	500
	PER	50		
VDO	$A0$	6	δ	1.5
	l_{max}	40	t_{max}	500
	γ	0.05		

Tables 11 and 12 display the objective values of the non-dominated solutions achieved in the Pareto archives of the suggested WFA, BCO and VDO algorithms for the 2 stated examples.

The validity of the presented NSP is examined through a small-sized problem-solution (Problem instance 1) and the accrued results being compared with a shift schedule which has been manually written down by the mentioned Infant Ward's supervisor.

Table 11. Objective Values of Pareto Archive Set by Presented WFA, BCO and VDO (Problem Instance 1)

WFA									
Solution number	1	2	3	4	5	6	7	8	9
Z1	47	35	47	35	46	34	45	16	45
Z2	57	56	57	56	56	55	60	59	60
Z3	330	360	330	360	350	380	290	340	290
Solution number	10	11	12	13	14	15	16	17	
Z1	49	29	28	36	17	33	27	44	
Z2	59	56	56	62	58	64	57	57	
Z3	280	410	430	250	300	280	420	350	

VDO			
Solution number	1	2	
Z1	41	42	
Z2	57	56	
Z3	380	370	

BCO				
Solution number	1	2	3	4
Z1	40	52	41	41
Z2	58	53	55	58
Z3	400	330	390	380

Table 12. Objective Values of Pareto Archive Set by Presented WFA, BCO and VDO (Problem Instance 2)

WFA													
Solution number	1	2	3	4	5	6	7	8	9	10	11	12	13
Z1	33	49	36	50	55	56	37	122	122	39	38	123	120
Z2	84	85	82	84	84	83	84	72	72	82	82	74	75
Z3	560	500	560	520	470	480	540	170	170	540	550	160	170
Solution number	14	15	16	17	18	19	20	21	22	23	24	25	26
Z1	129	37	112	20	20	129	105	104	104	102	130	129	114
Z2	77	79	66	68	69	77	67	67	67	68	74	73	66
Z3	60	560	250	670	630	60	230	250	250	240	70	150	180
Solution number	27	28	29	30	31	32	33	34	35	36	37	38	39
Z1	40	45	24	128	112	111	23	17	15	42	45	24	49
Z2	72	70	74	75	64	65	66	69	75	73	69	67	69
Z3	590	570	570	90	290	280	660	640	690	580	620	640	560

Solution number	40	41	42	43	44	45	46	47	48	49	50
Z1	20	20	127	108	15	15	16	128	109	19	114
Z2	70	70	78	70	77	79	81	75	65	66	65
Z3	600	600	130	220	680	660	650	90	300	750	270

VDO

Solution number	1	2	3	4	5	6
Z1	142	148	145	143	141	144
Z2	161	156	157	160	163	159
Z3	2590	2590	2580	2620	2610	2610

BCO

Solution number	1	2	3	4	5	6	7	8	9
Z1	149	141	152	146	148	142	143	149	142
Z2	154	164	160	155	155	161	157	156	163
Z3	2600	2620	2570	2630	2590	2620	2580	2580	2610

Moreover, to verify the present study introduced algorithms' reliability, the comparative metrics put forward below have been applied:

a) **Quality Assurance (QA) Metric:** This metric combines non-dominated solutions obtained by three algorithms and calculates the ratios between them.

b) **Space Metric (SM):** This metric [37] is calculated via the following formula:

$$\boldsymbol{SM} = \frac{\sum_{i=1}^{N-1}|d_{mean}-d_i|}{(N-1)*d_{mean}} \tag{48}$$

In which, d_i stands for Euclidean distance between the consecutive solutions of the achieved non-dominated set of solutions and d_{mean} shows the average distance of all the distances. Space metric is used to measure the spread uniformity of the points in the set of solutions.

c) **Diversification performance metric:** This metric [37] defines the span of the set of solution via calculating the formula below:

$$\boldsymbol{DM} = \sqrt{\sum_{i=1}^{N} max(\left\|x_t^i - y_t^i\right\|)} \tag{49}$$

Where, $\left\|x_t^i - y_t^i\right\|$ calculates Euclidean distance between the non-dominating solutions x_t^i andy_t^i.

Table 13 presents the length of time for computing both algorithms and values of the above mentioned comparative metrics. Through analyzing the accrued comparative metrics, we could conclude the following:

- The introduced WFA is a more time-consuming algorithm compared with VDO and BCO algorithms in the study, but the consumed time is still reasonable for solving real world problems.
- On the other hand, the study WFA can achieve higher number of Pareto solutions of higher qualities than the suggested VDO and BCO algorithms.
- The research VDO needs non-dominated solutions with the min values of space metric for both problem instances 1 and 2. This matter illustrates that the study VDO-induced non-dominated solutions are distributed more uniformly compared to the WFA and BCO algorithms introduced here.
- The diversification performance metric values in the introduced WFA are meaningfully greater than those from the study VDO and BCO algorithms.

Table 13. Computation-induced Results

Problem instance	Computational time (second)			Quality assurance metric			Space metric			Diversification performance metric		
	WFA	VDO	BCO	WFA	VDO	BCO	WFA	VDO	BCO	WFA	VDO	BCO
1	75.70	2.29	6.46	%75	40%	60%	0.4634	0	0.9898	47.4665	4.49	15.96
2	220.62	26.18	67.11	%92.8	%25	%75	0.4497	0.281	0.425	166.7889	14.21	20.87

To test the calculated measurability, the Relative Absolute Error(RAE) applied by Abolghasemian et al. [48-49] is utilized. For this purpose, the ARE value has been estimated between the calculation time of the mathematical modeling and the meta-heuristic algorithms. As seen in Table 14, ARE value is less than 5%. Therefore, given the confidence interval (CI) 95%, it can be concluded that the calculations are of good capability.

Table 14. ARE results

Problem instance	Meta-heuristic (Seconds)			Mathematical (Second)	ARE		
	WFA	VDO	BCO	Exact	ARE_{WFA}	ARE_{VDO}	ARE_{BCO}
1	75.70	2.29	6.46	80.25	0.02	0.001	0.036
2	220.62	26.18	67.11	125.36	0.01	0.001	0.046

5. Managerial Perceptions

The current study remarkable managerial perceptions are as follows:

- All patient care programs are documented by nurses.
- The needs and problems of patients are properly solved on time.
- The required standards of the nursing profession are formulated and developed.
- The nursing performance monitoring process is easily implemented.
- The qualification and competence of all nurses is constantly reviewed.

6. Conclusions

The present article has introduced a nurse scheduling problem (NSP) pursuing a real case study in the Infant Ward of Sina Hospital in Isfahan, considering hospital management requirements, nurses' preferences, hospital policies, labor laws and some other realistic factors to the greatest extent possible. Due to the nursing scheduling problem being of NP-hard nature, three meta-heuristics, i.e., WFA, BCO and VDO algorithms have been proposed to solve the formulated NSP. Moreover, in order to measure the quality of model's constraints and the study algorithms, two problem instances have been designed. Comparing the results accrued from the small-sized problem (problem instance 1) with a shift schedule manually drawn by the mentioned Infant Ward supervisor has confirmed the performance of the given model. Furthermore, some practical comparative metrics, such as quality assurance metric, space metric, and diversification performance metric have been employed to confirm the effectiveness of the utilized algorithms. The computation drawn results have revealed the recommended WFA outperforming the other two algorithms, namely, BCO and VDO in this paper and equipped with the potential to supply more efficient solutions. The presented NSP can be employed in similar wards and hospitals, if it undergoes some changes. It is worth mentioning that this research has been conducted in a hospital and on nurses who are considered as the main community of the current study, that is, with respect to their vital role in improving patients' health; however, it has been tough to sufficiently evaluate their role in the hospital during the research, which is pointed out as the most important limitation in this study. Future studies would include studying the fuzzy nature of hospital managers' objectives and nurses' preferences, considering some other effective factors in the model, new solution approaches, and healthcare scheduling in newer areas. Besides, the future study requires developing hospital associated mechanism simulation model and implementing the would-be model's optimizer.

References

[1] B. Cheang, H. Li, A. Lim and B. Rodrigues, Nurse Rostering Problems-A Bibliographic Survey, European Journal of Operational Research, **151 (3)** (2003) 447-460.

[2] E.K. Burke, P. De Causmaecker, G. Vanden Berghe and H. Van Landeghem, The State of the Art of Nurse Rostering, Journal of Scheduling, **7 (6)** (2004) 441-499.

[3] A. Ernst, H. Jiang, M. Krishnamoorthy and D. Sier, Staff Scheduling and Rostering: A Review of Applications, Methods and Models, European Journal of Operations Research, **153 (1)** (2004), 3-27.

[4] A. Ernst, H. Jiang, M.K.B. Owens, M. Krishnamoorthy and D. Sier, An Annotated Bibliography of Personnel Scheduling and Rostering, Annals of Operations Research, **127 (1)** (2004) 21-144.

[5] C.A. Glass and R.A. Knight, The Nurse Rostering Problem: A Critical Appraisal of the Problem Structure, European Journal of Operational Research, **202 (2)** (2010) 379-389.

[6] M.V. Chiaramonte and L.M. Chiaramonte, An Agent-Based Nurse Rostering System under Minimal Staffing Conditions, International Journal of Production Economics, **114 (2)** (2008) 697- 713.

[7] C.C. Tsai and S.H.A. Li, A Two-Stage Modeling with Genetic Algorithms for The Nurse Scheduling Problem, Expert Systems with Applications, **36 (5)** (2009) 9506-9512.

[8] E.K. Burke, J. Li and R. Qu, A Hybrid Model of Integer Programming and Variable Neighbourhood Search for Highly-Constrained Nurse Rostering Problems, European Journal of Operational Research, **203 (2)** (2010) 484-493.

[9] P. Shahnazari-Shahrezaei, R. Tavakkoli-Moghaddam, M. Azarkish and A. Sadeghnejad-Barkousaraie, A Differential Evolution Algorithm Developed for a Nurse Scheduling Problem, South African Journal of Industrial Engineering, **23 (3)** (2012) 68-90.

[10] K. Buyukozkan and A. Sarucan, Applicability of Artificial Bee Colony Algorithm for Nurse Scheduling Problems, International Journal of Computational Intelligence Systems, **7 (sup1)** (2014) 121-136.

[11] T.C. Wong, M. Xu and K.S. Chin, A Two-Stage Heuristic Approach for Nurse Scheduling Problem: A Case Study in an Emergency Department, Computers & Operations Research, **51** (2014) 99-110.

[12] A.A. Constantino, D. Landa-Silva, E.L. de Melo, C.F.X. de Mendonça, D.B. Rizzato and W. Romão, A Heuristic Algorithm Based on Multi-Assignment Procedures for Nurse Scheduling, Annals of Operations Research, **218 (1)** (2014) 165-183.

[13] A. Legrain, H. Bouarab and N. Lahrichi, The Nurse Scheduling Problem in Real-Life, Journal of medical systems, **39 (1)** (2015) 160.

[14] H. Jafari, S. Bateni, P. Daneshvar, S. Bateni and H. Mahdioun, Fuzzy Mathematical Modeling Approach for the Nurse Scheduling Problem: A Case Study, International Journal of Fuzzy Systems, **18 (2)** (2016) 320-332.

[15] G.J. Lim, A. Mobasher, J.F. Bard and A. Najjarbashi, Nurse Scheduling with Lunch Break Assignments in Operating Suites, Operations Research for Health Care, **10** (2016) 35-48.

[16] M. Bagheri, A.G. Devin and A. Izanloo, An Application of Stochastic Programming Method for Nurse Scheduling Problem in Real Word Hospital, Computers & Industrial Engineering, **96** (2016) 192-200.

[17] C. Di Martinelly and N. Meskens, A Bi-Objective Integrated Approach to Building Surgical Teams and Nurse Schedule Rosters to Maximise Surgical Team Affinities and Minimise Nurses' Idle Time, International Journal of Production Economics, **191** (2017) 323-334.

[18] A. Legrain, J. Omer and S. Rosat, An Online Stochastic Algorithm for a Dynamic Nurse Scheduling Problem, European Journal of Operational Research, **285 (1)** (2018) 196-210.

[19] S. Zanda, P. Zuddas and C. Seatzu, Long Term Nurse Scheduling via a Decision Support System Based on Linear Integer Programming: A Case Study at the University Hospital in Cagliari, Computers & Industrial Engineering, **126** (2018) 337-347.

[20] T.I. Wickert, P. Smet and G.V. Berghe, The Nurse Rerostering Problem: Strategies for Reconstructing Disrupted Schedules, Computers & Operations Research, **104** (2018) 319-337.

[21] M. Hamid, R. Tavakkoli-Moghaddam, F. Golpaygani and B. Vahedi-Nouri, A Multi-Objective Model for a Nurse Scheduling Problem by Emphasizing Human Factors, Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine, **234 (2)** (2019) 179-199.

[22] A.C. Svirsko, B.A. Norman, D. Rausch and J. Woodring, Using Mathematical Modeling to Improve the Emergency Department Nurse-Scheduling Process, Journal of Emergency Nursing, **45 (4)** (2019) 425-432.

[23] P.S. Chen, W.T. Huang, T.H. Chiang and G.Y.H. Chen, Applying Heuristic Algorithms to Solve Inter-Hospital Hierarchical Allocation and Scheduling Problems of Medical Staff, International Journal of Computational Intelligence Systems, **13 (1)** (2020) 318-331.

[24] N. Khalili, P. Shahnazari-Shahrezaei and A. Gholam Abri, A Multi-Objective Optimization Approach for a Nurse Scheduling Problem Considering the Fatigue Factor (Case Study: Labbafinejad Hospital), Journal of Applied Research on Industrial Engineering, **7 (4)** (2020) 396-423.

[25] N. Khalili, P. Shahnazari-Shahrezaei and A. Gholam Abri, Nurse Scheduling Problem Considering Service Level in Uncertain Conditions, Journal of Decisions and Operations Research, **7 (Special Issue)** (2022) 1-23.

[26] H. Jafari, Developing a Fuzzy Model for the Nurse Scheduling Problem, Journal of operational research in its applications (applied mathematics)-Lahijan Azad University, **17 (2)** (2020) 93-107.

[27] M. Benzaid, N. Lahrichi and L.M. Rousseau, Chemotherapy Appointment Scheduling and Daily Outpatient–Nurse Assignment, Health care management science, **23 (1)** (2020) 34-50.

[28] M. Hamid, R. Tavakkoli-Moghaddam, F. Golpaygani and B. Vahedi-Nouri, A Multi-Objective Model for a Nurse Scheduling Problem by Emphasizing Human Factors, Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine**, 234 (2)** (2020) 179-199.
[29] T. Osogami and H. Imai, Classification of Various Neighbourhood Operations for the Nurse Scheduling Problem, Lecture Notes in Computer Science, **1969** (2000) 72-83.
[30] M. Vanhoucke and B. Maenhout, On the Characterization and Generation of Nurse Scheduling Problem Instances, European Journal of Operational Research, **196 (2)** (2009) 457-467.
[31] F.C. Yang and Y.P. Wang, Water Flow-Like Algorithm for Object Grouping Problems, Journal of the Chinese Institute of Industrial Engineers, **24 (6)** (2007) 475-488.
[32] T.H. Wu, S.H. Chung and C.C. Chang, A Water Flow-Like Algorithm for Manufacturing Cell Formation Problems, European Journal of Operational Research, **205 (2)** (2010) 346-360.
[33] K. Deb, A. Pratap, S. Agarwal and T. Meyarivan, A Fast and Elitist Multi-Objective Genetic Algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation, **6 (2)** (2002) 182-197.
[34] M.J. Geiger, On Operators and Search Space Topology in Multi-Objective Flow Shop Scheduling, European Journal of Operational Research, **181 (1)** (2007) 195-206.
[35] E. Mehdizadeh and R. Tavakkoli-Moghaddam, Vibration Damping Optimization, In Proceedings of the International Conference of Operations Research and Global Business, Germany, (2008) 3-5.
[36] D.T. Pham, A. Ghanbarzadeh, E. Koc, S. Otri, S. Rahim and M. Zaidi, Bee Algorithm-A Novel Approach to Function Optimisation, Technical Note MEC 0501, (2005).
[37] A. Abraham, L. Jain and R. Goldberg, Evolutionary Multiobjective Optimization-Theoretical Advances and Applications, Springer-Verlag London Limited, United States of America, (2005).
[38] Z. Y. Zhuang and F. Y. Vincent, Analyzing the Effects of the New Labor Law on Outpatient Nurse Scheduling with Law-Fitting Modeling and Case Studies, Expert Systems with Applications, **180** (2021) 115103.
[39] F. Leung, Y. C. Lau, M. Law and S. K. Djeng, Artificial Intelligence and End User Tools to Develop a Nurse Duty Roster Scheduling System, International Journal of Nursing Sciences, **9 (3)** (2022) 373-377.
[40] J. Guo and J. F. Bard, A Column Generation-Based Algorithm for Midterm Nurse Scheduling with Specialized Constraints, Preference Considerations, and Overtime, Computers & Operations Research, **138** (2022) 105597.
[41] I. Forthun, S. Waage, S. Pallesen, B. E. Moen, and B. Bjorvatn, A Longitudinal Study of Work Time Schedule and Prescribed Sleep Medication Use in Norwegian Nurses, Sleep Medicine, **100 (S176)** (2022).
[42] A. M. Turhan and B. Bilgen, A Mat-Heuristic Based Solution Approach for an Extended Nurse Rostering Problem with Skills and Units, Socio-Economic Planning Sciences, (2022) 101300.
[43] P. S. Chen, W. T. Huang, G. Y.H. Chen, J. F. Dang and E. C. Yeh, Constructing Modified Variable Neighborhood Search Approaches to Solve a Nurse Scheduling Problem, International Journal of Production Research**, 62 (12)** (2024) 7186-7204.
[44] J. Thomas, Optimizing Nurse Scheduling: A Supply Chain Approach for Healthcare Institutions, Journal of Engineering and Science, **20 (6s)** (2024).
[45] C. N. Okoro, E. R. Egbuji and E. C. Mmahi, Integer Programming Approaches to Nurse Scheduling Problems (NSPs) in Management of Alex Ekwueme Federal University Teaching Hospital (AEFUTH), Abakaliki, Ebonyi State, Nigeria, Unpublished Manuscript, (2024).
[46] A. F. Torres Ramos, F. Castaño, N. Velasco and C. Montoya, A Multi-Objective Approach for the Nurse Scheduling Problem: A Home Health Care Delivery Case Study, SSRN Electronic Journal, (2025).
[47] A. Ghavampour, Heuristic Approaches for the Home Health Care Staff Scheduling Problem, University of Ottawa, Thesis, (2025).
[48] M. Abolghasemian, A. G. Kanai and M. Daneshmandmehr, A Two-Phase Simulation-Based Optimization of Hauling System in Open-Pit Mine, Iranian Journal of Management Studies, **13 (4)** (2020) 705-732.
[49] M. Abolghasemian and H. Darabi, Simulation Based Optimization of Haulage System of an Open-Pit Mine: Meta Modeling Approach, Organizational Resources Management Researches, **8 (2)** (2018) 1-17.