Applied Soft Computing Journal 93 (2020) 106336

Contents lists available at ScienceDirect

Applied Soft Computing Journal

journal homepage: www.elsevier.com/locate/asoc

Developing two heuristic algorithms with metaheuristic algorithms to improve solutions of optimization problems with soft and hard constraints: An application to nurse rostering problems

Ping-Shun Chen *, Zhi-Yang Zeng

Department of Industrial and Systems Engineering, Chung Yuan Christian University, Chung Li District, Taoyuan City, 320, Taiwan, ROC

ARTICLE INFO

Article history:
Received 29 August 2019
Received in revised form 11 March 2020
Accepted 19 April 2020
Available online 1 May 2020

Keywords:
Nurse rostering problem
Decision tree
Greedy search algorithm
Bat algorithm
Particle swarm optimization

ABSTRACT

Many researchers have studied optimization problems with soft and hard constraints, such as school timetabling, nurse rostering, vehicle routing with soft time window, and job/machine scheduling. Nurse rostering problem (NRP) is the research problem in this paper. This study proposes two heuristic algorithms, which are the decision tree method and the greedy search algorithm, to integrate with metaheuristic algorithms in order to generate better initial solutions in less time and to improve solutions' quality. This research examines the algorithms' performance based on two scenarios and two metaheuristic algorithms: bat algorithm (BA) and particle swarm optimization (PSO). For the two scenarios, BA (or PSO) with the decision tree method outperforms BA (or PSO) without the decision tree method, and BA (or PSO) with the greedy search algorithm outperforms BA (or PSO) without the greedy search algorithm. Furthermore, the results show that BA (or PSO) with the decision tree method and the greedy search algorithm can generate better initial solutions in less time and improve solutions' quality.

1. Introduction

Mathematicians endeavor to find the best solution to optimization problems. When solving a nondeterministic polynomial-time hard (NP-hard) problem, a mathematician can use mathematical programming methods, heuristic algorithms, or metaheuristic algorithms [1,2]. However, the time required to solve NP-hard problems increases with the scale of the problem, making it extremely difficult to identify the optimal solution within a reasonable amount of time [3]. For this reason, mathematicians prefer metaheuristic algorithms for solving NP-hard research problems, such as healthcare [4,5], facility layout [6], and job shop scheduling [7]. Although this type of algorithm cannot guarantee an optimal solution, it can rapidly identify a near-optimal one. This makes them suitable for solving large-scale NP-hard optimization problems.

In optimization problems with soft and hard constraints, hard constraints are mandatory constraints that must be satisfied. Solutions cannot violate any hard constraint. Soft constraints are nonmandatory constraints. However, a penalty is incurred when soft constraints are violated, but the solution may remain feasible [8–11]. This study solves an NP-hard optimization problem that has soft and hard constraints. The optimal solution to such a problem can be discovered by minimizing soft constraint violations or by allocating relative weights to the soft constraints and minimizing total penalties for soft constraints while satisfying all hard constraints [9].

Optimization problems containing soft and hard constraints could involve school timetabling [12–14], employee/nursing rostering [15–22], vehicle routing with soft time window [23–25], and job/machine scheduling [7]. Several studies have adopted metaheuristic algorithms to solve these types of problems [26–30]. Jans and Degraeve [27] applied several metaheuristic algorithms, such as tabu search, genetic algorithm, and simulated annealing, to solve dynamic lot sizing problems; their success validated the effectiveness of metaheuristic algorithms in solving packing problems. In this study, a nurse rostering problem (NRP) containing soft and hard constraints was selected as the optimization problem to test the proposed algorithms.

Metaheuristic algorithms have several flaws when solving multi-constraint problems. Beheshti [26] highlighted the most common, including (1) the tendency to produce local optimal solutions, (2) slow convergence rate, (3) long computation time, (4) excessive parameter adjustments, and (5) different algorithms require different programming logics. To resolve the slow convergence rate and long computation time when using metaheuristic algorithms to solve optimization problems containing soft and hard constraints, this study applied three strategies. First, decision trees were developed to analyze the soft and hard

* Corresponding author.
E-mail address: pingshun@cycu.edu.tw (P.-S. Chen).

https://doi.org/10.1016/j.asoc.2020.106336
1568-4946/

constraints, facilitating the rapid identification of initial feasible solutions. Second, for overcoming the tendency to produce local optimal solutions, a greedy search algorithm was developed to improve solution quality. Third, two scenarios were evaluated to compare the differences between with and without the proposed heuristic algorithms based on two metaheuristic algorithms: the bat algorithm (BA) and particle swarm optimization (PSO). The results validated the robustness and effectiveness of the algorithms.

The rest of this article is organized as follows. Section 2 reviews the literature on the NRPs by using metaheuristic algorithms. Section 3 defines an optimization problem with soft and hard constraints and develops methodology procedures to solve it. Section 4 uses NRPs (i.e., radiological technologist rosters) as a case study. Furthermore, this study adopts the BA and PSO to validate the decision tree method and the greedy search algorithm. Finally, Section 5 concludes and suggests directions for future research.

2. Literature review

2.1. Nurse rostering problems

A literature review was conducted to determine the methods that previous scholars have used to solve NRPs and identify the types of problems, soft constraints, and hard constraints. This study summarized the NRP literature from 2013 to 2018, by using metaheuristic algorithms. The methodology of NRPs can be classified as metaheuristic algorithms and variable neighborhood searches.

For the applications of the first class, Todorovic and Petrovic [15] proposed a bee colony optimization (BCO) for an NRP comprising three hard and 14 soft constraints. They penalized violations of soft constraints. The objective was to minimize the penalties. Finally, the findings were compared with those produced using a memetic algorithm, a shift sequence approach, a scatter search, and a variable depth search to validate the effectiveness of the proposed BCO.

Lin et al. [31] studied an NRP with joint normalized shift and day-off preference satisfaction, which contained manpower, day-off, and shift requirements. They used the work shift weight and day-off weight for each nurse to calculate his or her shift preferences. Furthermore, Lin et al. [31] developed a genetic algorithm with immigrant scheme (GAIS) and compared the results of the genetic algorithm (GA), GAIS, and GA with recovery scheme based on 20 to 100 nurses. Their results showed that GAIS was better than GA with recovery scheme.

Wu et al. [16] proposed a PSO approach for an NRP. They began with a mathematical model and then developed an algorithm for generating workstretch patterns which could be applied as templates in searches for initial solutions. Finally, they applied PSO to solve the NRP. However, their study was limited to hard constraints. Their findings validated PSO as way to solve NRPs with 10, 13, and 20 nurses.

Jin et al. [32] proposed two hybrid metaheuristic algorithms that combined the harmony search (HS) and artificial immune systems (AIS) into the hybrid HSAIS (HHSAIS) and cooperative harmony search and artificial immune systems (CHSAIS). They used the he First International Nurse Rostering Competition (INRC2010) testing datasets to compare their algorithms' performance. The results showed that the CHSAIS performed better than HS, AIS, HHSAIS, and GA.

Rajeswari et al. [11] proposed a multi-objective directed BCO algorithm to solve the NRP, containing five hard and 14 soft constraints. They used the INRC2010 testing datasets and compared their results with the artificial bee colony (ABC) algorithm, the hybrid ABC algorithm, the global best HS, the HS with hill climbing, and the integer programming technique for NRP. The numerical results showed that their proposed algorithm outperformed all of the other algorithms.

For the applications of the second class, Wong et al. [8] proposed a two-stage heuristic approach for an NRP in the emergency department of a hospital. A mathematical programming model was developed for the NRP. In the first stage, a schedule assignment algorithm was used to search for an initial solution that satisfied all of the hard constraints. In the second, a local search algorithm was applied to search for the neighborhood solutions to reduce the penalties incurred from soft constraint violations. The findings of the study validated that a two-stage heuristic approach could solve NRPs.

Tassopoulos et al. [10] proposed a two-phase variable neighborhood approach (VNA) for NRPs. They applied this approach to all NRP instances proposed in the INRC2010 testing datasets. The results were then compared with those proposed in published studies. The study confirmed that the two-phase VNA solved NRPs with a maximum of 50 nurses.

Zheng et al. [33] studied an NRP that contained two hard and 18 soft constraints. They developed a randomized variable neighborhood search algorithm. Based on the INCR2010 testing datasets, their results showed that the proposed algorithm was better than the two-phase variable neighborhood approach [10], the adaptive neighborhood search algorithm, the two-phase hybrid algorithm by the winner of the competition, the branch and price algorithm, the constraint programming, and the hyper-heuristic algorithm. The results showed that the proposed algorithm outperformed all of the others.

Liu and Xie [34] proposed two methods to estimate patient waiting time and formulated two corresponding physician staffing models for emergency departments with time-varying demand. They developed a variable neighborhood search (VNS) algorithm, consisting of shaking and local search mechanisms, to solve the proposed models. They compared patient waiting time of two models to that of a constructed simulation model. Based on the numerical tests, the results showed that patient waiting time obtained by solving each model is close to that obtained by running the constructed simulation model, thus validating the two proposed patient waiting time approximations.

2.2. Bat algorithm

Bats, the only mammals capable of powered flight, use a special echolocation principle to avoid obstacles and detect prey in the dark. Yang [35], from the University of Cambridge, simulated the relationship between bats' preying behavior and optimization, thereby formulating BAs. By observing minor changes in the temporal difference between the sound waves they generate and the echoes they detect with their ears, bats can accurately detect the locations of prey, even tiny insects. The animal identifies their target insects by identifying Doppler effect changes caused by their fluttering wings.

A bat is a tuple: '$\mathbf{x}$', 'A', 'f', and 'r'. $\mathbf{x}$ is the position of a bat. A is the loudness of a bat. f is the frequency of a bat and falls within a range of $[f_{min}, f_{max}]$. r is the pulse rate of a bat. The initial steps are to generate these tuples of bats and to calculate each bat's objective function value, $F(\mathbf{x})$. For the main loop of the BA, the global search updates each bat's frequency, velocity, and position. The frequency of a bat, f, is calculated by determining the difference between the maximum and minimum of the frequency, multiplying β, and adding the minimum of the frequency Eq. (1). Here, β follows a uniform distribution with the parameters (0, 1). The velocity of a bat is calculated by determining the difference between the current position of the bat and the current best position of bats, multiplying f, and adding the current velocity

Eq. (2). The position of a bat is calculated by adding the velocity of the bat to the current position Eq. (3).

For the local search, the pulse rate of a bat, r, is the threshold for a random procedure to run a local research step. If the random variable generated is greater than the pulse rate of the bat, the local search will generate the bat's new position Eq. (4) and objective function value. Here, the new position of the bat is calculated using its old position, adding the value of ε and multiplying the loudness of the bat (A). Otherwise, nothing is done. Next, each bat generates a new position by flying randomly. The loudness, A, and objective function value, $F(\mathbf{x})$, of the bat are used to determine whether or not to accept the better bat. If the better bat is accepted, the loudness of the bat will decrease according to Eq. (5) and the pulse rate of the bat will increase according to Eq. (6). The iteration number will increase by one. If the maximum number of iterations is reached (the terminating condition), the main loop of the BA will stop. The current best bat ($\mathbf{x}_*$) is the best found solution.

$$f_i^t = f_{\min} + (f_{\max} - f_{\min}) \times \beta, \text{ where } \beta = \mathrm{U}(0, 1). \tag{1}$$

$$v_i^t = v_i^{t-1} + (\mathrm{x}_i^t - \mathrm{x}_*) \times f_i^t. \tag{2}$$

$$\mathrm{x}_i^t = \mathrm{x}_i^{t-1} + v_i^t. \tag{3}$$

$$\mathrm{x}_{\text{new}} = \mathrm{x}_{\text{old}} + \varepsilon \times A^t, \text{ where } \varepsilon = \mathrm{U}(-1, 1), \text{ and } A^t \text{ is the average loudness of all bats at iteration } t. \tag{4}$$

$$A_i^{t+1} = \alpha \times A_i^t, \text{ where } \alpha \in \text{uniform } [0, 1]. \tag{5}$$

$$r_i^{t+1} = r_i^0 \times (1 - \exp^{-\gamma t}), \text{ where } \gamma \text{ is a positive constant.} \tag{6}$$

BA can be used to solve integer optimization (or discrete variable) problems; the results of Yang's [35] research showed that, for the 10 testing benchmarking functions, PSO had better solutions than the GA, while BA surpassed the GA and PSO methods in terms of solving time and efficiency. Since its development, BA has been used to solve optimization problems involving continuous variables [36].

2.3. Particle swarm optimization

Kennedy and Eberhart [37] proposed particle swarm optimization (PSO) in 1995 to solve continuous nonlinear functions. This method was based on bird flocking, fish schooling, and swarming theory. PSO consisted of two searches, exploration (global optimal) and exploitation (local optimal), and required few parameters. There were advantages to using PSO to solve optimization problems. Wu et al. [16] have used PSO for solving the NRPs.

A particle represents a pair of location '**x**' and its local attractor '*y*'. The initial procedures of PSO generate new particles, **x**, and the objective function value, $F(x)$. Before reaching the maximum number of iterations (the terminating condition), each particle must update its velocity and location by using Eqs. (7) and (8). Here, ω represents the inertia weight, c_1 represents the cognitive parameter, c_2 represents the social parameter, and r_1 and r_2 are random numbers that fall within a range of (0, 1). The local attractor '*y*' of a particle is the second item of Eq. (7), implying that the new location is affected by the current best solution of a particle. The global attractor of a particle is the third item of Eq. (7), implying that the new location is affected by the best group solution of all particles (or the swarm).

If a better solution of particles is obtained, the current best solution (*pBest*) of this particle will be updated. If the updated solution (*pBest*) of the particle is better than the current best group solution (*gBest*) of all particles, the current best group solution (*gBest*) of all particles will be updated. Finally, after PSO stops, the current best group solution (*gBest*) is the best found solution.

$$v_i(t+1) = \omega * v_i(t) + c_1 r_1(pBest - \mathrm{x}_i(t)) + c_2 r_2(gBest - \mathrm{x}_i(t)). \tag{7}$$

where $r_1 = \mathrm{U}(0, 1)$, $r_2 = \mathrm{U}(0, 1)$.

$$\mathrm{x}_i(t+1) = \mathrm{x}_i(t) + v_i(t+1). \tag{8}$$

2.4. A summary

Based on this literature review, NRPs are NP-hard optimization problems that contain both soft and hard constraints [1,2]. Mathematicians have adopted heuristic/metaheuristic algorithms [38,39] and popular mathematical software [40,41], such as CPLEX, GAMS, and LINGO, to solve NRPs or employee scheduling problems. However, the best solution obtained by heuristic or metaheuristic algorithms are not always the optimal solution. Beheshti [26] claimed that all algorithms have advantages and disadvantages. This study proposes a way to improve the effectiveness of metaheuristic algorithms in optimizing soft and hard constraints and generating initial solutions, and to resolve the problem of local searches generating ill-suited neighborhood solutions.

This concept can be applied to metaheuristic algorithms for generating medical staff rosters that satisfy government regulations and hospital regulations, and reduce unsatisfied shifts of medical staff. Two algorithms proposed in this study can generate better solutions and greatly reduce calculation time. Section 4 will present the numerical results of this study.

3. Methodology

3.1. Mathematical programming model

The mathematical programming model consisted of decision variables, hard constraints, soft constraints, and an objective function. Each decision variable, X_i, was either 0 or 1. The hard constraints depended on the decision variables. For example, functions associated with any variable X_i must be c, $\leq c$, or $\geq c$. Therefore, three types of hard constraints, Eqs. (9)–(11), were relevant:

$$H_1(X_i) = c \tag{9}$$

$$H_2(X_i) \leq c \tag{10}$$

$$H_3(X_i) \geq c \tag{11}$$

The soft constraints depended on the decision variables and had three forms. Therefore, three types of soft constraints, Eqs. (12)–(14), were relevant:

$$S_1(X_i) = c. \tag{12}$$

$$S_2(X_i) \leq c. \tag{13}$$

$$S_3(X_i) \geq c. \tag{14}$$

In Eq. (15), P_1^+ and P_1^- were the penalties for violating soft constraint (12). P_2 in Eq. (16) and P_3 in Eq. (17) were the penalties for violating soft constraints (13) and (14), respectively. P_i was a decision variable that could only take nonnegative values.

$$S_1(X_i) - P_1^+ + P_1^- = c. \tag{15}$$

$$S_2(X_i) - P_2 \leq c. \tag{16}$$

$$S_3(X_i) + P_3 \geq c. \tag{17}$$

The mathematical model proposed in this study contained soft constraints. Such models tend to minimize the weighed violations of soft constraints by minimizing total penalties. In the proposed model, P_i represents the penalty incurred for violating soft constraint i, W_i represents the weight of violating soft constraint i, and n represents the number of soft constraints. Therefore, the objective function of the proposed mathematical programming model for optimization containing soft and hard constraints was (18), where Eqs. (9)–(11) were the hard constraints and Eqs. (15), (16), and (17) were the soft constraints associated with penalties. The decision variables were X_i and P_i.

Mathematical model:

$$\text{Minimize } \sum_{i=1}^{n} P_i \times W_i \tag{18}$$

Subject to

Constraints (9)–(11) and (15)–(17).

3.2. Mathematical programming of the NRPs

3.2.1. Notations of the NRPs

Parameters:

d: index of working days, and $d = 1, 2, 3, \ldots, D$; D is the number of monthly working days.

s: index of shifts, and $s = 1$ (day shift: 8:00–16:00), 2 (evening shift: 16:00–24:00), and 3 (night shift: 24:00–8:00).

u: index of subunits.

n: index of nurses, and $n = 1, 2, 3, \ldots, N$; N is the number of nurses.

$n_{u,s,d}$: number of nurses required for the subunit u, shift s, on day d.

d_1: maximal consecutive days off for each nurse by hospital.

d_2: maximal consecutive working days for each nurse by government.

ω_{ij}: penalty for a nurse working shift i followed by shift j the next day, where $i, j \in \{1, 2, 3\}$ and $i < j$.

ω_{RiR}: penalty for a nurse working patterns of "off–shift i–off" on three consecutive days, where $i = 1, 2, 3$.

ω_{3R1}: penalty for a nurse working patterns of "shift 3–off–shift 1" on three consecutive days.

Decision variables:

$W_{n,u,s,d} = 1$ if nurse n is assigned to subunit u, shift s, on day d, and 0 otherwise.

$R_{n,d} = 1$ if nurse n is assigned a day off on day d, and 0 otherwise.

$Pij_{n,d} = 1$ if nurse n works shift i on day d followed by shift j the next day, and 0 otherwise, where $i, j \in \{1, 2, 3\}$ and $i < j$.

$PRiR_{n,d} = 1$ if nurse n works patterns of "off–shift i–off" on days d, $d+1$, and $d+2$, and 0 otherwise, where $i = 1, 2, 3$.

$P3R1_{n,d} = 1$ if nurse n works patterns of "shift 3–off–shift 1" on days d, $d+1$, and $d+2$, and 0 otherwise.

3.2.2. Hard constraints of the NRPs

Hard constraints consist of hospital regulations and government regulations.

- Each nurse either works in only one subunit for the day or has the day off (hospital regulations).

$$\sum_{u}\sum_{s=1}^{3} W_{n,u,s,d} + R_{n,d} = 1, \quad \text{for } d = 1, \ldots, D,\ n = 1, \ldots, N. \tag{19}$$

- Each subunit requires $n_{r,s,d}$ nurses for subunit r, shift s, and day d (hospital regulations).

$$\sum_{n=1}^{N} W_{n,u,s,d} = n_{u,s,d}, \quad \text{for } u, s{=}1, 2, 3, \text{ and } d = 1, \ldots, D. \tag{20}$$

- Nurses must not take more than d_1 consecutive days off (hospital regulations).

$$\sum_{d'=0}^{d_1} R_{n,d+d'} \le d_1, \quad \text{for } n = 1, \ldots, N,\ d = 1, \ldots, D - d_1. \tag{21}$$

- Nurses must take at least d_2 days off for every 7-day period (government regulations).

$$\sum_{d'=0}^{6} R_{n,d+d'} \ge d_2, \quad \text{for } n = 1, \ldots, N, d = 1, \ldots, D - 6. \tag{22}$$

- Nurses must not take shift i the day after taking shift j (government regulations). This is because the government requires that a nurse needs to rest at least 12 h between shifts.

$$\sum_{u} W_{n,u,j,d} + \sum_{u} W_{n,u,i,d+1} \le 1, \quad \text{where } i, j \in \{1, 2, 3\} \text{ and } i < j,$$
$$\text{for } n = 1, \ldots, N, d = 1, \ldots, D - 1. \tag{23}$$

3.2.3. Soft constraints of the NRPs

Soft constraints, Eqs. (24) through (26), represent nurses' preferences.

(1) No nurse takes shift j the day after taking shift i for special pair of (i, j). This soft constraint means that, if a nurse takes shift j the day after taking shift i, the penalty value will be 1.

$$\sum_{u} W_{n,u,i,d} + \sum_{u} W_{n,u,j,d+1} - Pij_{n,d} \le 1,$$
$$\text{where } i, j \in \{1, 2, 3\} \text{and } i < j, \tag{24}$$
$$\text{for } n = 1, \ldots, N, d = 1, \ldots, D - 1.$$

(2) An "off–shift i–off" order does not appear in a nurse's shift schedule. This soft constraint means that, if a nurse works on the "off–shift i–off" pattern, the penalty value will be 1.

$$R_{n,d} + \sum_{u} W_{n,u,i,d+1} + R_{n,d+2} - PRiR_{n,d} \le 2, \quad \text{where } i = 1, 2, 3,$$
$$\text{for } n = 1, \ldots, N, d = 1, \ldots, D - 2. \tag{25}$$

(3) A "shift 3–off–shift 1" order does not appear in a nurse's shift schedule. This soft constraint means that, if a nurse works on the "shift 3–off–shift 1" pattern, the penalty value will be 1.

$$\sum_{u} W_{n,u,3,d} + R_{n,d+1} + \sum_{u} W_{n,u,1,d+2} - P3R1_{n,d} \le 2,$$
$$\text{for } n = 1, \ldots, N, d = 1, \ldots, D - 2. \tag{26}$$

3.2.4. Objective function and mathematical model of the NRPs

The objective function (27) is to minimize the sum of penalties caused by violating the soft constraints/nurses' preferences. Each soft constraint includes a corresponding penalty decision variable, and the associated cost of these penalty decision variables has been added to the objective function.

$$\text{Min} \sum_{\substack{i,j\in\{1,2,3\}\\ \text{and } i<j}} \sum_{n=1}^{N} \sum_{d}^{D-1} (\omega_{ij} \times Pij_{n,d}) + \sum_{i=1}^{3} \sum_{n=1}^{N} \sum_{d}^{D-2} (\omega_{RiR} \times PRiR_{n,d}) + \sum_{n=1}^{N} \sum_{d=1}^{D-2} (\omega_{3R1} \times P3R1_{n,d}) \tag{27}$$

Based on the descriptions in Sections 3.2.1–3.2.4, the mathematical model of the NRPs in this study can be summarized as follows:

Minimize the objective function = Minimize Eq. (27)

Subject to:

Constraints (19)–(23) for hard constraints.

Constraints (24)–(26) for soft constraints.

3.3. Research procedures

In problems with multiple hard constraints, randomly generating initial or feasible solutions for each generation may be time-consuming because the violation of even one hard constraint results in the collapse of the entire solution. To eliminate infeasible solutions, researchers have two methods: to omit current infeasible solutions, generate a new set of solutions, and repeat this process until reaching a feasible solution; or to convert infeasible into feasible solutions by repairing the violated constraints and repeating this process until a feasible solution is generated. The first method takes a long time to generate a feasible solution and is not guaranteed. Therefore, this study adopts the second method.

The proposed heuristic algorithms are: generate initial solutions by using a decision tree method, and generate a local search by running a greedy search algorithm. In the first part, decision trees formulated a suitable set of rules to prevent violations of hard constraints while solutions are being generated. By reducing the number of hard constraint violations, the likelihood of generating infeasible solutions can be greatly decreased, minimizing the time required to revise infeasible solutions. In the second part, the greedy search algorithm was developed based on the weight of violating soft constraints.

To validate the proposed heuristic algorithms, the outcomes of three strategies were analyzed. The first strategy comprised BA (or PSO) with and without the decision tree method; the second comprised BA (or PSO) with and without the greedy search algorithm; the third comprised BA (or PSO) with the decision tree method and the greedy search algorithm. The procedures of BA and PSO are presented next. The quality of the initial solutions and the obtained solutions are subsequently analyzed in Section 4.4.

To encode and decode BA and PSO, this study uses a three-dimensional array to record the nurses' monthly shift schedule. A 16-nurse monthly schedule (e.g., June) will be a 16*3*30 array (nurses * shifts * days), which can be viewed as the position of a bat or a particle. For example, 'array[1, 1, 3] = 1' represents that nurse 1 is assigned to a day shift on day 3 while 'array[10, 2, 5] = 0' represents that nurse 10 is not assigned to an evening shift on day 5.

To generate the position of a bat or a particle, the values of the 16*3*30 array by the day shift, evening shift, night shift, or day-off must be filled. To satisfy Eqs. (19) and (20), the initial position (i.e., solution) of a bat or particle generates the required number of each shift and day-off and then randomly assigns a shift/day-off to one nurse. For example, a unit has 16 nurses to roster over three day shifts (8:00–16:00, shift value = 1), three night shifts (16:00–24:00, shift value = 2), and three evening shifts (24:00–8:00, shift value = 3) that occur on weekdays, implying a total of nine work shifts for each weekday. Government regulations require that each nurse who works one shift must subsequently have a 12-hour rest period, meaning each nurse either works one shift or has a day off on any particular day. The 16 nurses will be assigned to the nine work shifts and seven day-off shifts for weekdays. The BA or PSO will generate an initial position $\mathbf{x} = (x_{1,s,d}, x_{2,s,d}, x_{3,s,d}, \ldots, x_{16,s,d})$, where the value of $x_{i,s,d}$ represents that nurse i is assigned or not assigned to shift s on day d. The 16 positions will have three day, evening, and night shifts and seven day-off shifts for weekdays. Here, the initial position of a bat or a particle will comply with Eqs. (19) and (20), thereby leaving less time to rectify the violations caused by these two equations.

To update the position, the velocity of a bat is calculated using Eqs. (1) and (2). The new position of a bat is calculated using Eq. (3). In order to simplify the complexity of the example problem, we reduce the schedule from a 30-day to a one-day problem for 16 nurses. Before using a BA, we assume that $f_{min} = 0$ and f_{max} = the number of days in a given month (i.e., $f_{max} = 30$ in June). The initial v_0 value of bat i is assumed to be 0. β is randomly generated by a uniform distribution with parameters between 0 and 1. If β is 0.5, using Eq. (1) to calculate the frequency of bat i at iteration 1 will lead to 0 + (30–0) * 0.5 = 15.

In this research, we define that the sum of different values between two position representations based on a modified XOR gate operator is the distance between the original and best found positions of the bat. For example, (1, 1, 1, 0, 2, 2, 2, 0, 3, 3, 3, 0, 0, 0, 0, 0) represents the assignment of 16 nurses to shifts 1, 2, or 3 on day 1 at iteration 0. We assume that (1, 1, 0, 0, 1, 0, 2, 2, 0, 2, 3, 3, 3, 0, 0, 0) is the current best bat's assignment on day 1. The computation results from the modified XOR gate operator between two representations are (0, 0, 1, 0, 1, 1, 0, 1, 1, 1, 0, 1, 1, 0, 0, 0), totaling 8. Using Eq. (2), the velocity of bat i at iteration 1 is 0 + 8 * 15 = 120 (new velocity value). The remainder of the new velocity divided by the number of nurses is calculated as 120 % 16 = 8. Here, the '%' notation represents the remainder operator. Furthermore, we define the calculated remainder as the new velocity of bat i. According to Eq. (3), the new position of bat i will be the original position plus the value of the new velocity. In order to calculate the new position of bat i, we transform Eq. (3) to a modified XOR gate operator by randomly exchanging eight values of (1, 1, 1, 0, 2, 2, 2, 0, 3, 3, 3, 0, 0, 0, 0, 0) on day 1. If the positions of (3, 4), (7, 8), (11, 12), and (1, 16) are exchanged with each other, the new position of bat i on day 1 at iteration 1 becomes (0, 1, 0, 1, 2, 2, 0, 2, 3, 3, 0, 3, 0, 0, 0, 1).

Similarly, to update the position, the velocity of a particle is calculated using Eq. (7). The new position of a particle is calculated by Eq. (8). We assume that ω is 0.8, c_1 is 2, c_2 is 2, r_1 is 0.3, r_2 is 0.5, and $v_i(0)$ is 1.3, where r_1 = U(0, 1), r_2 = U(0, 1), and v_i = U(1, 2) in the example. At iteration 1, the original position of (1, 1, 1, 0, 2, 2, 2, 0, 3, 3, 3, 0, 0, 0, 0, 0) represents that the assignment of 16 nurses to shifts 1, 2, or 3 on day 1 at iteration 0. We assume that *pBest* is same as the original position and *gBest* is (1, 1, 0, 0, 1, 0, 2, 2, 0, 2, 3, 3, 3, 0, 0, 0). We use the modified XOR gate operator to calculate the distance between two positions, such as the distance between the original and *pBest* positions and the distance between the original and *gBest* positions. Using Eq. (7), the new velocity of particle i, v_i (1), is

Generating Initial Bats

1. Generate initial bats. Each bat represents by $\mathbf{x} = (x_1, x_2, \ldots, x_d)^T$. The objective function value of a bat is $F(\mathbf{x})$, where $\mathbf{x}_*$ is the current best bat.
2. Parameter initialization for each bat: position $\mathbf{x}_i^0$, velocity v_i^0, and loudness A_i^0, pulse rate r_i^0, and pulse frequency f_i^0; Here, the pulse frequency f falls in a range of $[f_{\min}, f_{\max}]$; $f_{\min}$ is the minimal pulse frequency; and $f_{\max}$ is the maximal pulse frequency.
3. **for** each bat in the population
4. Generate a shift schedule by the ***Decision Tree Method*** for each bat;
5. **if** the current bat violate any one of the hard constraints (i.e., Equations (21), (22), and (23))
6. Apply the corresponding shift schedule correction mechanisms to repair the current bat;
7. **end if**
8. **end for**
9. Output the feasible solution of each bat;

Searching Procedures

10. $t = 1$;
11. **Loop**
12. **for** each bat in the population at the t iteration
13. Update the shift schedule for each bat, $\mathbf{x}_i^t$, by using Equations (1), (2), and (3);
14. **if** ($rand > r_i$) at the t iteration
15. Generate an adjacent bat for a local search by using Equation (4);
16. Generate a new bat for the ***Greedy Search Algorithm***;
17. **end if**
18. Generate a new bat by flying randomly;
19. **if** the current bat violates any one of the hard constraints, Equations (21), (22), and (23)
20. Apply the corresponding shift schedule correction mechanisms to repair the current bat;
21. **end if**
22. **if** ($rand < A_i$ and $F(\mathbf{x}_i) < F(\mathbf{x}_*)$) at the t iteration
23. Accept the new bat ($\mathbf{x}_i^t$) as the current best bat ($\mathbf{x}_*$);
24. Reduce A_i^t and increase r_i^t by using Equations (5) and (6);
25. **end if**
26. Rank the bats' solutions by objective function values and update the current best bat ($\mathbf{x}_*$);
27. **end for**
28. $t = t + 1$;
29. **do while** ($t <$ the maximum number of iterations or the current best bat ($\mathbf{x}_*$) is equaled to the optimal solution obtained by IBM ILOG CPLEX)
30. Output the current best bat ($\mathbf{x}_*$);

Fig. 1. The pseudo code of BA with both heuristic algorithms.

calculated as 0.8 * 1.3 + 2 * 0.3 * 0 + 2 * 0.5 * 8 = 9.04 and the ceiling of 9.04 is 10. Therefore, v_i (1) is 10. Using Eq. (8), the new position of particle i is the original position plus the value of the new velocity. In order to calculate the new position of particle i, we transform Eq. (8) to a modified XOR gate operator by randomly exchanging ten values of (1, 1, 1, 0, 2, 2, 2, 0, 3, 3, 3, 0, 0, 0, 0, 0) on day 1. If the positions of (1, 4), (2, 5), (3, 6), (9, 12), and (10, 13) are exchanged with each other, the new position of particle i on day 1 at iteration 1 becomes (2, 2, 2, 0, 1, 1, 1, 0, 0, 0, 3, 3, 3, 0, 0, 0).

This study proposes the decision tree method and greedy search algorithm and integrates them into BA and PSO. Fig. 1 depicts the pseudo code of BA with the decision tree method and greedy search algorithm. To generate initial bats, Step 4 uses the proposed decision tree method for each bat to generate a shift schedule. Step 16 uses the proposed greedy search to generate a new bat. The remaining steps are the same as those of the traditional BA. Similarly, Fig. 2 depicts the pseudo code of PSO with the decision tree method and greedy search. Step 4 uses the proposed decision tree method for each particle to generate a shift schedule. Step 21 uses the proposed greedy search to generate a new particle. The remaining steps are the same as those of the traditional PSO.

For BA (or PSO) with the greedy search algorithm, the random generation method is used to replace the decision tree method in Step 4 to generate a shift schedule for each bat and particle. For BA (or PSO) with the decision tree method, Step 16 (or Step 21) is canceled. The details of the decision tree method and greedy search algorithm will be presented in Section 4.2.

4. Results and discussions

4.1. Radiological technologist rosters in the research hospital

The subjects of this study were 16 radiological technologists on rotation in the emergency diagnostic radiology unit of the research hospital. They performed detailed examinations and analyses using X-ray, portable X-ray, and computed tomography scanners; and uploaded patients' images to the hospital's picture archiving and communication system.

Numerous studies on NRPs have adopted number of staff per shift, number of staff per day, and special shift patterns as the hard constraints [9,11,17,32,33]. The diagnostic radiology unit analyzed in this study operated on a 24-hour rotation split into three shifts. The unit was split into three subunits, which were the X-ray unit (R), the portable X-ray unit (P), and the computed

Generating Initial Particles

1. Generate initial particles. Each particle represents by $\mathbf{x} = (x_1, x_2, \ldots, x_d)^T$. The objective function value of a particle is $F(\mathrm{x})$, where and $\mathbf{x}_*$ is the current best particle;
2. Parameter initialization for each particle: position $\mathbf{x}_i^0$, velocity v_i^0, inertia weight ω, cognitive parameter c_1, social parameter c_2, and random variables r_1 and r_2;
3. **for** each particle in the population
4. Generate a shift schedule by the ***Decision Tree Method*** for each particle;
5. **if** the current solution violate any one of the hard constraints (i.e., Equations (21), (22), and (23))
6. Apply the corresponding shift schedule correction mechanisms to repair the current particle;
7. **end if**
8. **end for**
9. Output the feasible solution of each particle;

Searching Procedures

10. $t = 1$;
11. **Loop**
12. **for** each particle in the population at the t iteration
13. Calculate fitness value of each particle;
14. **if** the fitness value is better than the best fitness value (*pBest*)
15. Set this fitness value as the new *pBest* of this particle;
16. **end if**
17. **if** the best fitness value of all the particle is better than the best group fitness value (*gBest*)
18. Set this best fitness value of all the particle as the current best particle ($\mathbf{x}_*$);
19. **end if**
20. Update the shift schedule for each particle, $\mathbf{x}_i^t$, by using Equations (7) and (8);
21. Generate an adjacent particle for the ***Greedy Search Algorithm*** and repeat steps 12 to 19 and go to step 22;
22. **if** the current particle violates any one of the hard constraints, Equations (21), (22), and (23)
23. Apply the corresponding shift schedule correction mechanisms to repair the current particle;
24. **end if**
25. **end for**
26. $t = t + 1$;
27. **do while** (t < the maximum number of iterations or the current best particle ($\mathbf{x}_*$) is equaled to the optimal solution obtained by IBM ILOG CPLEX)
28. Output the current best particle (x_*);

Fig. 2. The pseudo code of PSO with both heuristic algorithms.

tomography unit (C). The shifts were the day (08:00–16:00), evening (16:00–24:00), and night (24:00–08:00). In addition to complying with government regulations, rosters were required to satisfy hospital policies. Therefore, the rostering of radiological technologists is a type of NRP. In this case study, the hard constraints were:

1. Each radiological technologist may be allocated to either one shift or a day off.
2. Two radiological technologists are required to work the night shift in the P subunit on Sundays.
3. With the exception of Constraint 2, one radiological technologist is required to work in each shift of each subunit daily.
4. Radiological technologists shall receive no more than four consecutive off days.
5. Radiological technologists shall receive at least one day off for every seven consecutive work days.
6. Radiological technologists shall not work a day shift following a night shift.
7. Radiological technologists shall not work a day shift following an evening shift.
8. Radiological technologists shall not work a night shift following an evening shift.

To simplify the random generation of feasible solutions, this study presents an initial solution mechanism that satisfies Constraints 1 to 3. If a generated initial solution violates at least one of Constraints 4 to 8, a corresponding recovery mechanism will be applied. Some problems require adjustments to the hard constraints; the numbers of radiological technologists required in Constraint 2 and Constraint 3 depend on the problem definition. Different testing instances in Section 4.4 require different numbers of radiological technologists for each subunit, each shift, and each day.

4.2. Generation of initial and neighborhood solutions

The procedures of this research consisted of two stages. The first stage was the generation of the initial solution set using a decision tree method, and the second was the generation of neighborhood solutions using a greedy search algorithm. Decision trees were developed and applied in the first stage to apply radiological technologist roster information and generate an initial solution set that met the radiological technologist staffing requirements of the research hospital. In the second stage, a greedy search algorithm was performed at the end of each generation to improve the solutions' quality, thereby meeting the radiological technologists' expectations.

The decision tree method developed in this study was different from those proposed in published studies that have randomly generated initial solutions [10,16]. In this study, decision trees were developed to analyze the details of the problem and the set of feasible solutions, thereby reducing the time required to generate the initial solution set.

Table 1
Weight of violating each soft constraint by using the AHP.

Factors	(24)(1)	(24)(2)	(24)(3)	(25)(1)	(25)(2)	(25)(3)	(26)	Weight
Each worker does not take evening shift the day after taking day shift (24)(1)	0.04	0.05	0.10	0.04	0.03	0.03	0.04	0.055
Each worker does not take night shift the day after taking day shift (24)(2)	0.03	0.04	0.06	0.03	0.02	0.04	0.10	0.051
Each worker does not take night shift the day after taking evening shift (24)(3)	0.02	0.03	0.04	0.04	0.03	0.05	0.06	0.046
An "off–day shift–off" order does not appear in a worker's shift schedule (25)(1)	0.16	0.22	0.16	0.16	0.16	0.13	0.16	0.170
An "off–evening shift–off" order does not appear in a worker's shift schedule (25)(2)	0.14	0.22	0.20	0.12	0.13	0.11	0.12	0.153
An "off–night shift–off" order does not appear in a worker's shift schedule (25)(3)	0.32	0.30	0.23	0.31	0.31	0.26	0.21	0.269
An "night shift–off–day shift" order does not appear in a worker's shift schedule (26)	0.29	0.13	0.20	0.29	0.31	0.38	0.30	0.257

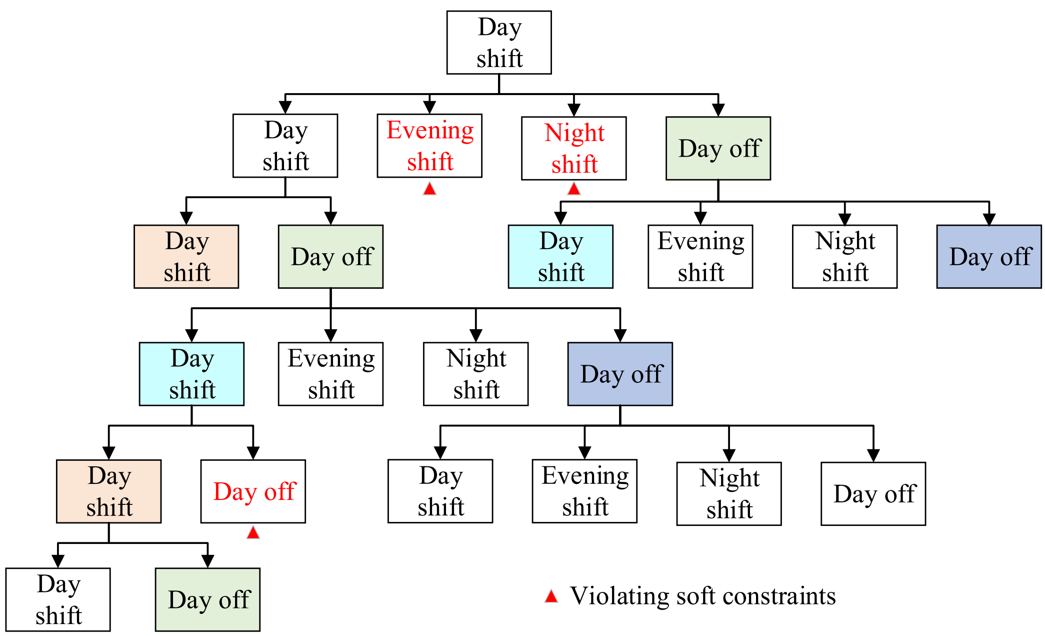

Fig. 3. "Day shift" pattern's decision tree.

For the soft constraints of the radiological technologist rostering problem, this study adopted Eqs. (24)–(26) as its soft constraints, and extended them into seven constraints by shifts, Eqs. (24)(1), (24)(2), (24)(3), (25)(1), (25)(2), (25)(3), and (26), as shown in Table 1. Then, this study calculated the corresponding weights of the seven constraints by the analytic hierarchy process (AHP). Based on the results of the questionnaires for investigating six radiological technologists' preferences, Eq. (28) was used to calculate a pairwise comparison matrix, A, of the seven factors. Here, a_{ij} represented the average score of factor i compared to factor j evaluated by the six radiological technologists. By using Eq. (29), this study obtained the weight of the seven key factors, which were 0.055, 0.050, 0.046, 0.170, 0.153, 0.269, and 0.257. The details of the AHP method can be found in the Refs. [9, 42–44].

$$A = [a_{ij}] = \begin{bmatrix} 1 & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ \frac{1}{a_{1n}} & \cdots & 1 \end{bmatrix}. \tag{28}$$

$$W_i = \frac{(\prod_{j=1}^{n} a_{ij})^{1/n}}{\sum_{i=1}^{n}(\prod_{j=1}^{n} a_{ij})^{1/n}} \quad \forall i = 1, 2, \ldots, n. \tag{29}$$

For the weights of the AHP, the consistency of questionnaires must be checked by calculating the consistency ratio (*C.R.*) value. In this case, the *C.R.* value of the data was 0.05 and less than 0.1, indicating that the six radiological technologists agreed on the weight of the seven factors. Hence, the AHP could determine the weights of the seven soft constraints in the objective function, Eq. (27), of the proposed NRP.

Decision trees were developed to analyze the soft and hard constraints in the radiological technologist rostering problem. Four rules were identified by analyzing the decision trees illustrated in Figs. 3–5: (1) radiological technologists should be allocated to fixed daily shifts; penalties are incurred when the previous and current shifts are different; this difference may even lead to hard constraint violation; (2) the shift logic resets once a radiological technologist has an off day (except night shift–off day–day shift); (3) radiological technologists may be allocated into any subunit on any shift after two consecutive off days; and (4) a radiological technologist should be assigned to a shift on the current day if he/she had only worked one day since the last off day; otherwise, this situation may incur a penalty (off day–work day–off day).

An initial solution set was generated based on these rules. The solution set was tested to ensure that it met all hard constraints.

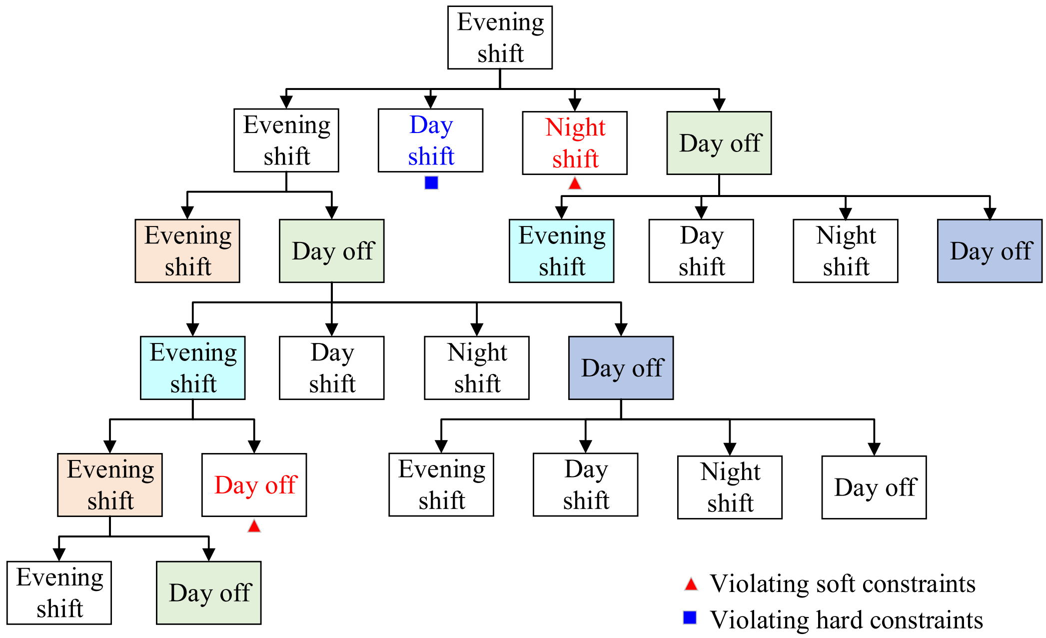

Fig. 4. "Evening shift" pattern's decision tree.

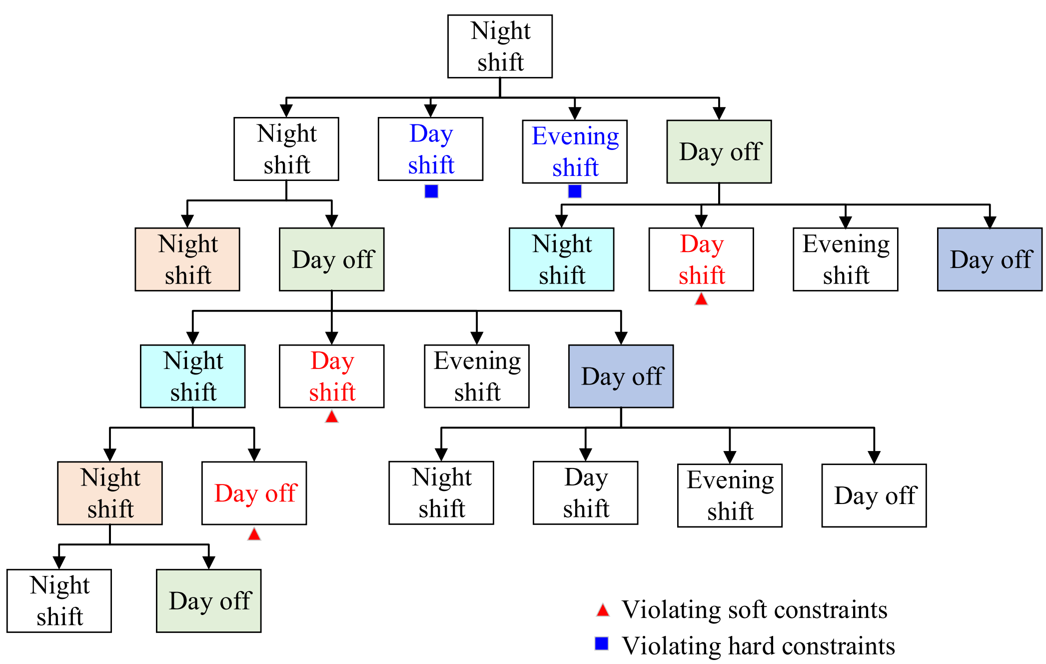

Fig. 5. "Night shift" pattern's decision tree.

When there was a violation, a recovery mechanism was applied to revise the initial solution set, thereby guaranteeing the feasibility of the initial solution set in the second stage.

In the second stage, the greedy search algorithm was used to search for neighboring solutions and generate a better solution. The details of the radiological technologist rostering problem were referenced to maximize the probability that the local search would generate a favorable solution set. The weights for soft constraint violations are obtained by the AHP method in Table 1. In Fig. 6, the greedy search algorithm applied the information in this table to ensure that the objective function value reflected the soft constraint violations of the current solution set before performing any search in the next generation, thereby increasing the chance of identifying a competitive solution. In other words, a schedule violating any soft constraints and the times of those violations were considered when searching for neighborhood solutions. For each iteration, running the greedy search algorithm changed only one of soft constraints' violations. The greedy search algorithm could reduce ineffective searches, enhance solution quality, and increase computation efficiency. The numerical results will be presented in Section 4.4.

In every round, when the neighborhood solution set yielded solutions that violated hard constraints, the recovery mechanism revised those solutions that violated the hard constraints 4 to 8 in Section 4.1. Any solutions that could not be revised were omitted from the solution set generated in that round. The same recovery mechanism was applied to the proposed heuristic algorithm, BA, and PSO in order to obtain a feasible solution. Due to space

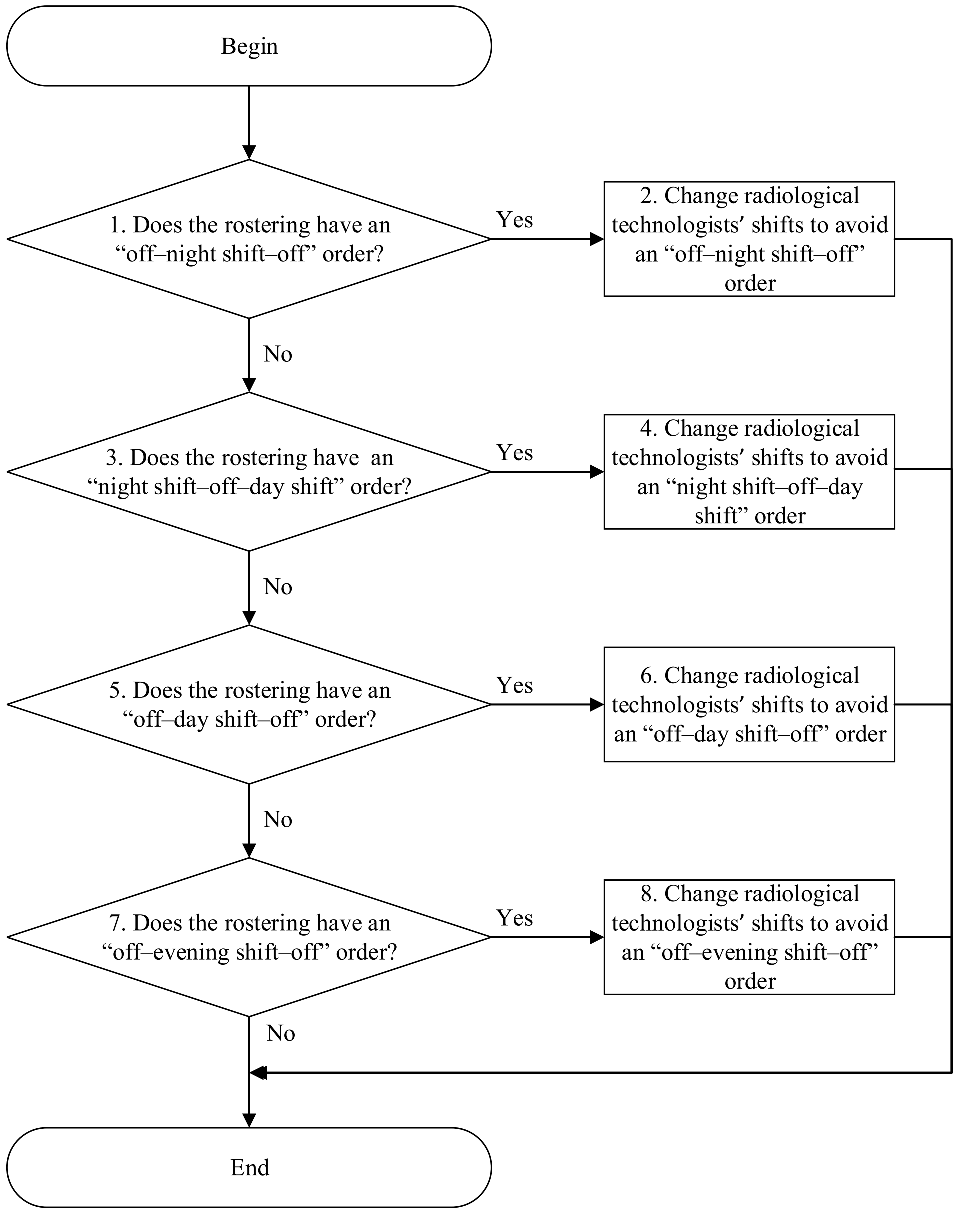

Fig. 6. Flowchart of the greedy search algorithm.

constraints, this study excluded the introductions of the recovery mechanism. This issue merits further exploration of different recovery mechanisms.

4.3. Parameter settings for the proposed BA and PSO

The PHP programming language was used to develop a version of the heuristic algorithm coupled with BA and a version coupled with PSO. The algorithms required metaheuristic parameters to be set before the applications were run. The PHP programming language was used to develop a user-friendly web interface to facilitate control and observation. The interface enabled users to select radiological technologist rosters by year, month, or number of radiological technologists, and set the number of iterations. After the parameters were set, the user pressed the compute button on the interface to run the algorithm. The system listed all solution details. Once each computation was complete, the interface displayed the optimal radiological technologist roster for the specified month, objective function value, items constituting the objective function value, and run time.

This study gives two examples of how to determine an appropriate number of iterations and population for the BA and PSO. Figs. 7(a) and 7(b) show that, for two runs of the 9 to 10 radiological-technologist setting, BA and PSO without both heuristic algorithms (decision tree method and greedy search algorithm) could not converge on the optimal solution, which was 0 in this setting, within 400 iterations based on the 10 bats/particles. This means that, if BA or PSO without both heuristic algorithms wants to search for the optimal solution, it should increase the number of iterations and/or population. However, integrating both heuristic algorithms with the BA or PSO could obtain the optimal solution within 330 iterations for the first run and within 385 iterations for the second run, as shown in Figs. 7(c), 7(d), 7(e), and 7(f).

Furthermore, the initial solution of BA (or PSO) with one or both heuristic algorithms was less than 3 for the first run and less than 4 for the second, which was much smaller than that (i.e., about 17 and 18.5 for the first and second runs) of BA (or PSO) without both heuristic algorithms. BA (or PSO) with one or both heuristic algorithms converged on a certain value about

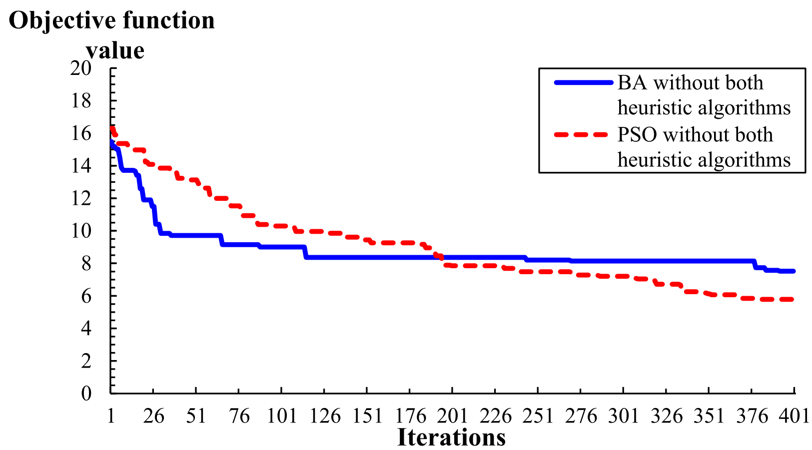

Fig. 7(a). Convergence curve of the BA and PSO without both heuristic algorithms for the first run.

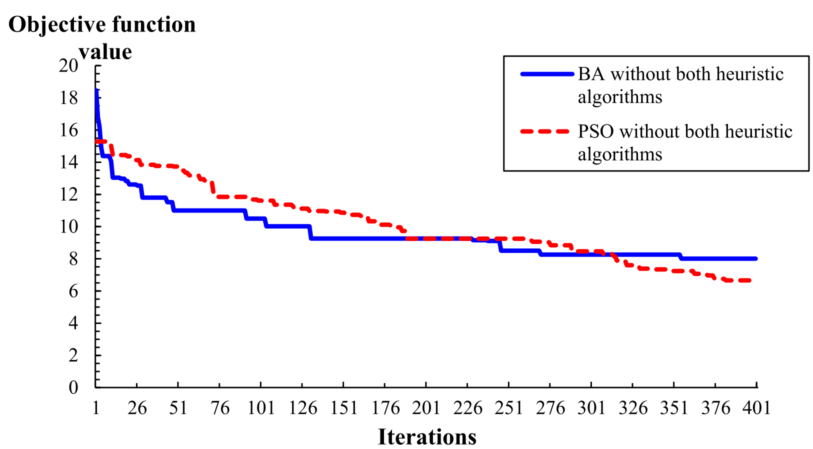

Fig. 7(b). Convergence curve of the BA and PSO without both heuristic algorithms for the second run.

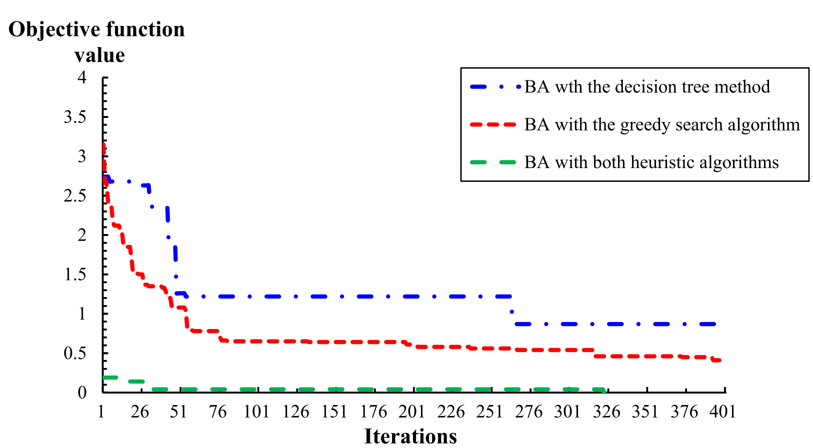

Fig. 7(c). Convergence curve of the BA with one or both heuristic algorithms for the first run.

350 and 380 iterations for the first and second runs, respectively. Therefore, the maximum number of iterations was set to 400, and population to 10 bats or particles based on the proposed two heuristic algorithms. Similarly, the remaining parameters in the BA and PSO were determined by trial and error of the references of BA [45–47] and PSO [48–50]. The parameter settings are tabulated in Table 2.

4.4. Results and discussions

A personal computer, equipped with an Intel(R) Core(TM) i7-6700 CPU @ 3.40 GHz, 8GB of DDR3 memory, and Windows 7 Ultimate, was used for computation. The PHP programming language was used to develop the heuristic algorithms, BA, and PSO for the proposed radiological technologist rostering problem. The PHP programming language was also used to build a

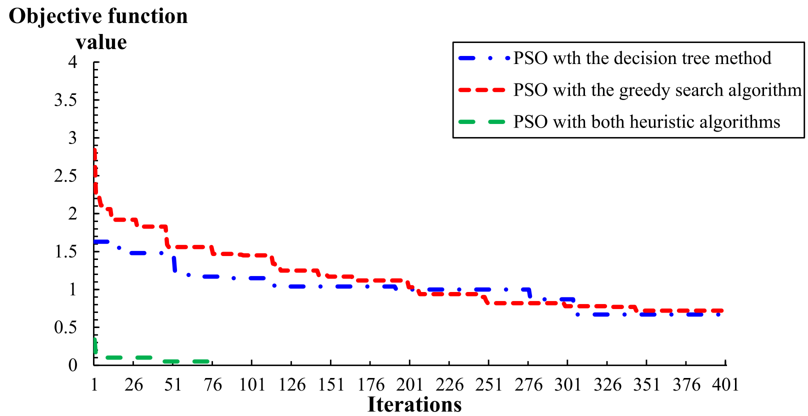

Fig. 7(d). Convergence curve of the PSO with one or both heuristic algorithms for the first run.

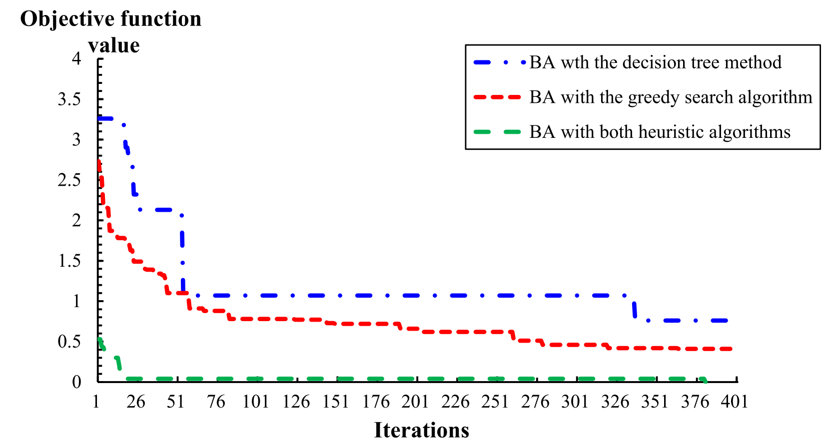

Fig. 7(e). Convergence curve of the BA with one or both heuristic algorithms for the second run.

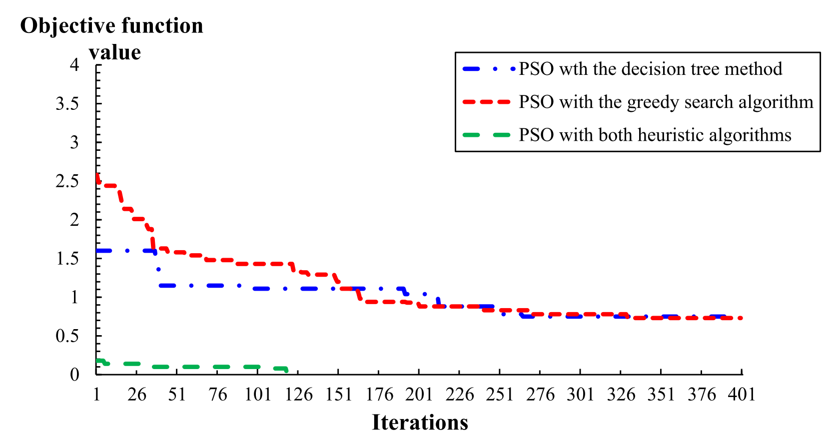

Fig. 7(f). Convergence curve of the PSO with one or both heuristic algorithms for the second run.

Table 2
Parameters of BA and PSO methods.

Parameters	BA	Parameters	PSO
Maximum number of iterations	400	Maximum number of iterations	400
Loudness	0.9∼1	Learning factors: c1 and c2	2
Pulse frequency	0.05∼0.2	Inertia weight	0.8
Number of bats	10	Number of particles	10

visual interface for the administrator. The scale of the radiological technologist rostering problem was based on the size of the research hospital, specifically, 16 radiological technologists, four daily groups of workers (three working shifts plus day off), three subunits, and one 30-day roster. Based on the scale of the problem, the decision variables (0–1 variables) for workers' working shifts defined a set of $2^{3\times3\times16\times30}= 2^{4,320}$ possibilities before generating an initial solution and using the decision tree method. In order to satisfy Eqs. (19) and (20), the possible sets of generating an initial solution reduced to$(C_3^{16} \times 3 \times C_3^{13} \times 2 \times C_3^{10} \times 1 \times C_7^7)^{30} = (115,315,200)^{30} \leq (2^{27})^{30} = 2^{810}$. Here, we picked up three of the 16 radiological technologists for one of three shifts (day, evening, and night), three from the remaining 13 for one of the two remaining shifts, three from the remaining 10 for the remaining shift, and seven from the remaining seven for the day-off shift. For the decision tree method, we first assigned the selected shift for each radiological technologist on the first day, we determined an appropriate shift for each radiological technologist on the second day based on the rules (the day shift followed by the day or day-off shift; the evening shift followed by the evening or day-off shift; the night shift followed by the night or day-off shift; and the day-off shift followed by the day, evening, night, or day-off shift) in Figs. 3–5, and the procedures were repeated until each radiological technologist was scheduled for all 30 days. Hence, for the worst case, the possible sets of solution spaces based on the decision tree method would reduce to 2^{697} calculated by Eq. (30), much less than the original possible sets, $2^{4,320}$. Therefore, the decision tree method not only generated a feasible solution, but also explored less searching spaces, leading to shorter solution time.

$$\begin{aligned}&(C_3^{16} \times 3 \times C_3^{13} \times 2 \times C_3^{10} \times 1 \times C_7^7) \times ((2 \times 2 \times 2)\\&\times (2 \times 2 \times 2) \times (2 \times 2 \times 2) \times (4 \times 4 \times 4 \times 4 \times 4 \times 4 \times 4))\\&\times \cdots \times ((2 \times 2 \times 2) \times (2 \times 2 \times 2) \times (2 \times 2 \times 2)\\&\times (4 \times 4 \times 4 \times 4 \times 4 \times 4 \times 4))\\&= (115,315,200) \times (2^{23})^{29} \leq 2^{27} \times 2^{667} = 2^{694}.\end{aligned} \tag{30}$$

The proposed heuristic algorithms were combined with metaheuristic algorithms, such as BA and PSO, and applied to the radiological technologist rostering problem to determine whether these combinations could achieve consistently superior convergence and generate useful near-optimal solutions. For the case hospital, the radiological technologist rostering problem required nine radiological technologists from Monday to Saturday and ten on Sunday. Three radiological technologists were required for each of the R, P, and C groups from Monday to Sunday and extra one was required for the night shift of the C group on Sunday. In actuality, the number of medical staff required per subunit on a daily basis depended on the number of patients in the department. Consequently, the required values of the roster may be affected by the allocation of radiological technologists. Therefore, two radiological technologist requirements were examined in this study: a daily requirement of 9 to 10 radiological technologists (research hospital) in Section 4.4.1 and a daily requirement of 20 to 38 radiological technologists in Section 4.4.2.

After a situation had been selected, a weekly roster was generated by randomly selecting the number of radiological technologists required for each day while satisfying all of the hard constraints. The weekly roster was copied, and then the remaining days were calculated to complete the roster for a full month.

4.4.1. The first testing scenario

The proposed decision tree method and greedy search algorithm were separately coupled with BA and PSO to calculate a radiological technologist rostering problem with a daily radiological technologist requirement of 9 to 10 people. Each of the 10 shift types was calculated 30 times. To validate the stability of the two algorithm combinations, the following comparisons were made: (1) the effects of the initial solution generated by the decision tree method on the objective function values produced using BA versus the effects of the randomly generated initial solution on the objective function values produced using BA; (2) the effects of the initial solution generated by the decision tree method on the objective function values produced using BA with the greedy search algorithm versus the effects of the initial solution generated by the decision tree method on the objective function values produced using BA without the greedy search algorithm; (3) the effects of the randomly generated solution on the objective function values produced using BA with the greedy search algorithm versus the effects of the randomly generated initial solution on the objective function values produced using BA without the greedy search algorithm; (4) the effects of the initial solution generated by the decision tree method on the objective function values produced using PSO versus the effects of the randomly generated initial solution on the objective function values produced using PSO; (5) the effects of the initial solution generated by the decision tree method on the objective function values produced using PSO with the greedy search algorithm versus the effects of the initial solution generated by the decision tree method on the objective function values produced using PSO without the greedy search algorithm; and (6) the effects of the randomly generated solution on the objective function values produced using PSO with the greedy search algorithm versus the effects of the randomly generated initial solution on the objective function values produced using PSO without the greedy search algorithm.

Table 3
Ten settings of daily required numbers of radiological technologists for "9- to 10-radiological technologist scenario".

Setting	Mon	Tue	Wed	Thu	Fri	Sat	Sun
1	3	3	3	3	3	3	4C
2	4C	3	4C	4C	4P	3	3
3	4R	4R	4C	4P	3	4R	4P
4	4R	4P	3	3	4P	4P	4C
5	4C	4P	3	4P	3	4P	4R
6	3	3	4C	4C	4P	3	3
7	4C	4R	4R	4P	3	4C	4P
8	3	3	4C	4P	4C	4C	3
9	4P	3	4R	4C	4R	3	4C
10	3	4C	3	4R	4P	4C	4R

The allocations for a pool of 9 to 10 radiological technologists for the 10 settings are tabulated in Table 3. In Setting 1, 3 means that three radiological technologists are allocated to the R, P, and C subunits from Monday to Saturday. On Sunday, 4C means that four radiological technologists are allocated to the C subunit, where one radiological technologist is allocated to the day and evening shift, and two radiological technologists are allocated to the night shift; and three to the P and C subunits, where one radiological technologist is allocated to the day, evening, and night shift. Similarly, 4P in Setting 2 means that four radiological technologists are allocated to the P subunit and three to the C and R subunits on Friday. 4R in Setting 3 means that four radiological technologists are allocated to the R subunit and three to the C and P subunits on Monday, Tuesday, and Saturday. The results are presented in Tables A.1–A.4.

This study uses one example to illustrate the best solution obtained by BA (or PSO) with the decision tree method or with the greedy search algorithm. The best solution obtained by BA with the decision tree in Fig. 7(c) was 0.87, which violated three of seven soft constraints: Eq. (25)(1) once, Eq. (25)(2) three times, and Eq. (26) once, as shown in Fig. 8(a). The best solution obtained by BA with the greedy search algorithm in Fig. 7(c) was 0.41, which violated Eq. (24)(1) twice, Eq. (24)(2) three times, and Eq. (24)(3) four times, as shown in Fig. 8(b). Similarly, the best solution obtained by PSO with the decision tree in Fig. 7(d) was 0.67, which violated Eq. (24)(1) once, Eq. (24)(2) twice, Eq. (25)(3) once, and Eq. (26) once, as shown in Fig. 8(c). The best solution obtained by PSO with the greedy search algorithm in Fig. 7(d)

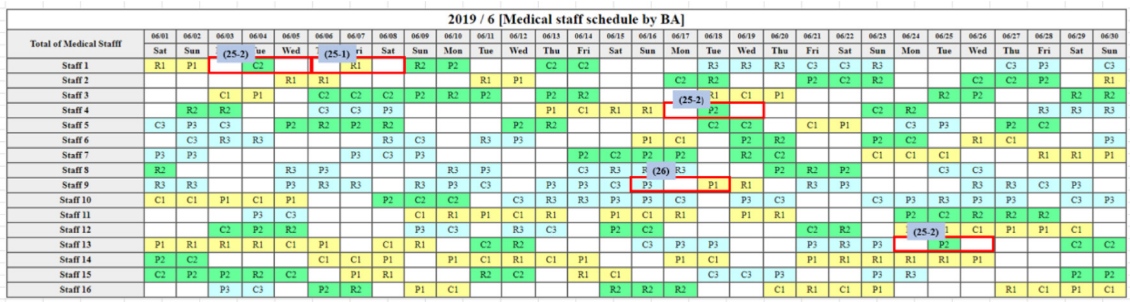

Fig. 8(a). The best solution by BA with the decision tree method for Fig. 7(c).

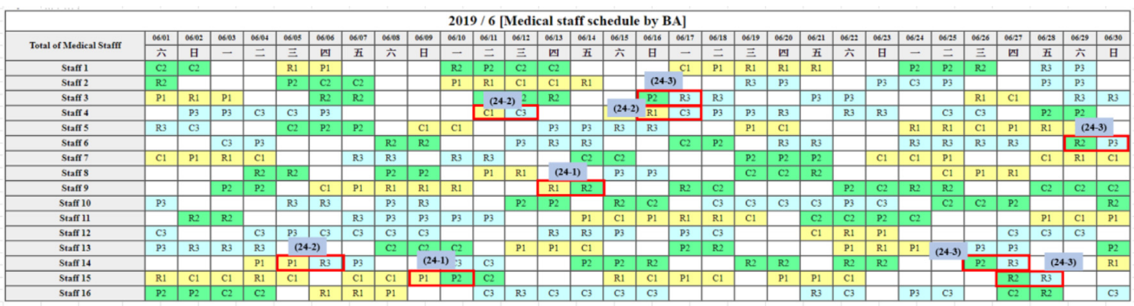

Fig. 8(b). The best solution by BA with the greedy search algorithm for Fig. 7(c).

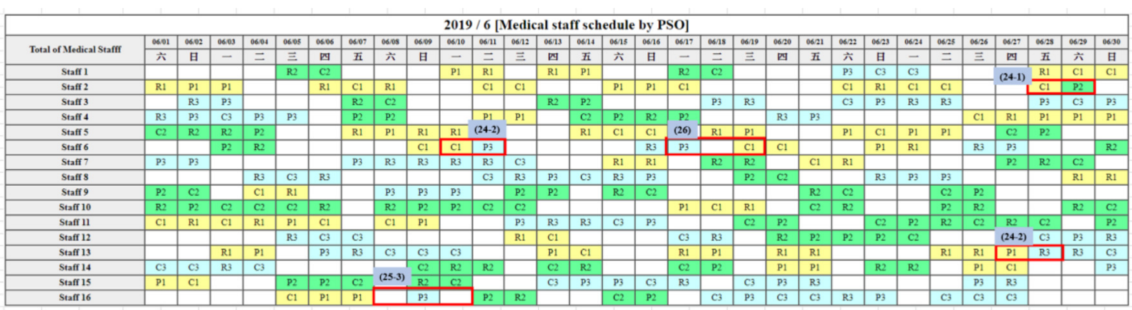

Fig. 8(c). The best solution by PSO with the decision tree method for Fig. 7(d).

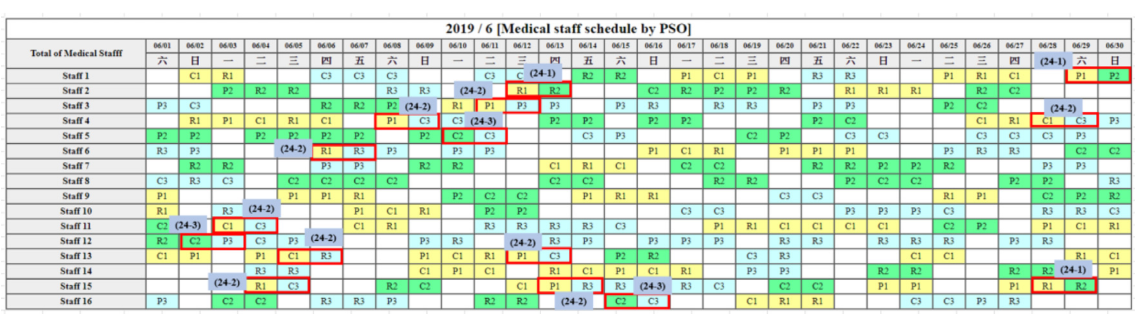

Fig. 8(d). The best solution by PSO with the greedy search algorithm for Fig. 7(d).

was 0.72, which violated Eq. (24)(1) three times, Eq. (24)(2) nine times, and Eq. (24)(3) three times, as shown in Fig. 8(d).

Based on the numerical results related to the seven soft constraints, three observations were found. The first observation was that BA with the decision tree method had fewer chances to violate Eqs. (24)(1), (24)(2), and (24)(3) than BA with the greedy search algorithm. The same observation existed between PSO with the decision tree method and PSO with the greedy algorithm. This was because the decision tree method was designed to consider the shift combinations, thereby leading to fewer chances to violate the two-day shift preferences, Eqs. (24)(1), (24)(2), and (24)(3).

The second observation was that BA (or PSO) with the greedy search algorithm had had fewer chances to violate Eqs. (25)(1), (25)(2), (25)(3), and (26) than BA (or PSO) with the decision tree. This was because the greedy search algorithm was designed to consider the weight of the seven soft constraints, and the rank of weights of the seven cost constraints was Eq. (25)(3), followed by Eqs. (26), (25)(1), (25)(2), (24)(1), (24)(2), (24)(3). Therefore, there

Table 4
Effects of two sources of initial solutions on the results of BA.

Setting (9 to 10 radiological technologists)	Optimal value obtained by IBM ILOG CPLEX	BA solutions with using the decision tree method for initial solutions				BA solutions with using random generation method for initial solutions			
		Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value	Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value
1	0	1.12	0.19	1.53	0.74	7.73	0.44	8.51	6.84
2	0	1.11	0.28	1.65	0.55	7.62	0.58	8.52	6.33
3	0	1.19	0.25	1.59	0.59	8.07	0.50	9.31	7.16
4	0	1.22	0.23	1.67	0.75	7.61	0.59	8.99	6.01
5	0	1.26	0.20	1.50	0.61	8.08	0.44	8.94	6.65
6	0	1.13	0.19	1.39	0.66	7.22	0.67	8.36	5.51
7	0	1.25	0.22	1.74	0.79	7.71	0.63	9.03	6.44
8	0	1.22	0.22	1.58	0.73	7.62	0.53	8.64	6.67
9	0	1.14	0.24	1.61	0.65	8.13	0.52	8.84	6.86
10	0	1.26	0.19	1.55	0.81	7.88	0.77	9.03	6.10

Table 5
Results for BA with initial solutions generated randomly with and without the greedy search algorithm.

Setting (9 to 10 radiological technologists)	Optimal value obtained by IBM ILOG CPLEX	BA solutions with using the greedy search algorithm				BA solutions without using the greedy search algorithm			
		Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value	Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value
1	0	0.20	0.09	0.43	0.05	7.73	0.44	8.51	6.84
2	0	0.22	0.08	0.48	0.13	7.62	0.58	8.52	6.33
3	0	0.32	0.12	0.58	0.09	8.07	0.50	9.31	7.16
4	0	0.25	0.07	0.40	0.13	7.61	0.59	8.99	6.01
5	0	0.22	0.10	0.48	0.04	8.08	0.44	8.94	6.65
6	0	0.13	0.09	0.37	0.00	7.22	0.67	8.36	5.51
7	0	0.26	0.11	0.55	0.08	7.71	0.63	9.03	6.44
8	0	0.18	0.07	0.36	0.09	7.62	0.53	8.64	6.67
9	0	0.21	0.08	0.42	0.05	8.13	0.52	8.84	6.86
10	0	0.27	0.11	0.52	0.05	7.88	0.77	9.03	6.10

were fewer chances to violate the three-day shift preferences, Eqs. (25)(1), (25)(2), (25)(3), and (26).

The third observation was that BA (or PSO) with both heuristic algorithms in Fig. 7(c) (or Fig. 7(d)) obtained the optimal solution, which was 0, leading to no violation of the seven soft constraints. This meant that integrating the decision tree method and the greedy search algorithm could comply with the medical staff preferences. Hence, the proposed both heuristic algorithms were validated.

Table 4 compares the effects of two sources of initial solutions on the results of BA. The initial solutions were generated by the decision tree method and by a random generation method. The table shows, that among the 10 settings, the average objective function values produced from randomly generated initial solutions ranged from 7.22 to 8.13. All settings achieved a minimum objective function value greater than 5.51. When using the initial solutions generated by the decision tree method, the average objective function values were between 1.11 and 1.26, and the minimum objective function values were between 0.55 and 0.81, suggesting convergence of the average objective function values and a significant shift toward the optimal solution of 0. The optimal solutions were calculated using IBM ILOG CPLEX.

Table 5 compares the results of BA with and without the greedy search algorithm. Among the 10 settings, all average objective function values and minimum objective function values produced using the randomly generated solution with the greedy search algorithm were better than those produced using the randomly generated solution without the greedy search algorithm. After 30 rounds, the settings that achieved an average objective function value of roughly 7.22 to 8.13 without the greedy search algorithm dropped to less than 0.4 with the application of the greedy search algorithm, suggesting that in addition to the initial solution generated by the decision tree method, the greedy search algorithm also improved solution quality.

Table 6 compares the optimal solution of BA without both heuristic algorithms, BA with the decision tree method, BA with the greedy search algorithm, and BA with both heuristic algorithms. The table shows that, among the 10 settings (9 to 10 radiological technologists), BA with both heuristic algorithms had better solution quality, followed by BA with the greedy search algorithm, BA with the decision tree method, and BA without both heuristic algorithms. This means that using the decision tree and greedy search algorithm could improve solution quality of BA, leading to obtain a near-optimal solution. In contrast, BA without both heuristic algorithms fell into the local optimal solution with the range of [7.22, 8.13].

Table 7 compares the solution time of BA without both heuristic algorithms, BA with the decision tree method, BA with the greedy search algorithm, and BA with both heuristic algorithms. The table shows that, among the 10 settings, BA with the decision tree method and BA with both heuristic algorithms had a shorter average solution time than BA and BA with the greedy search algorithm. This means that using the decision tree to generate an initial solution could reduce the average solution time much more than using a random way to generate an initial solution would. Furthermore, sometimes BA with both heuristic algorithms had the shortest average solution time. The reason for this finding was that, for some testing runs, BA with both heuristic algorithms obtained an optimal solution compared to the optimal solution obtained by IBM ILOG CPLEX and stopped the search procedures. However, compared to the solution time by using IBM ILOG CPLEX, BA with and without heuristic algorithms still took too long to obtain a near-optimal solution. It merits further research.

Table 6
Optimal solutions of BA with and without heuristic algorithms.

Setting (9 to 10 radiological technologists)	Optimal value obtained by IBM ILOG CPLEX	BA solutions without both heuristic algorithms		BA solutions with the decision tree method		BA solutions with the greedy search algorithm		BA solutions with both heuristic algorithms	
		Average objective function value	Standard deviation of objective function value	Average objective function value	Standard deviation of objective function value	Average objective function value	Standard deviation of objective function value	Average objective function value	Standard deviation of objective function value
1	0	7.73	0.44	1.12	0.19	0.20	0.09	0.03	0.05
2	0	7.62	0.58	1.11	0.28	0.22	0.08	0.05	0.07
3	0	8.07	0.50	1.19	0.25	0.32	0.12	0.06	0.03
4	0	7.61	0.59	1.22	0.23	0.25	0.07	0.05	0.05
5	0	8.08	0.44	1.26	0.20	0.22	0.10	0.05	0.02
6	0	7.22	0.67	1.13	0.19	0.13	0.09	0.02	0.02
7	0	7.71	0.63	1.25	0.22	0.26	0.11	0.07	0.07
8	0	7.62	0.53	1.22	0.22	0.18	0.07	0.03	0.03
9	0	8.13	0.52	1.14	0.24	0.21	0.08	0.07	0.10
10	0	7.88	0.77	1.26	0.19	0.27	0.11	0.05	0.07

Table 7
Solution time of BA with and without heuristic algorithms.

Setting (9 to 10 radiological technologists)	Solution time of obtaining optimal value obtained by using IBM ILOG CPLEX (s)	BA solutions without both heuristic algorithms		BA solutions with the decision tree method		BA solutions with the greedy search algorithm		BA solutions with both heuristic algorithms	
		Average solution time (s)	Standard deviation of solution time	Average solution time (s)	Standard deviation of solution time	Average solution time (s)	Standard deviation of solution time	Average solution time (s)	Standard deviation of solution time
1	2	109.53	10.58	57.03	2.06	112.57	8.65	31.20	24.83
2	3	109.33	9.57	54.87	2.27	113.03	8.68	51.23	22.37
3	2	120.43	9.94	53.47	2.49	132.43	9.32	55.67	14.74
4	2	114.73	10.39	53.87	2.22	124.43	10.91	57.33	15.91
5	2	115.80	9.60	54.50	2.85	128.00	11.78	57.13	14.30
6	2	100.90	6.01	54.37	2.41	122.50	12.25	49.23	21.13
7	2	125.80	10.64	53.47	2.90	149.03	15.94	56.90	12.95
8	2	106.73	7.80	54.77	3.78	112.73	7.69	38.03	22.95
9	2	129.70	13.06	56.63	4.58	121.87	7.35	53.10	20.16
10	2	116.33	11.42	56.23	2.61	121.07	7.37	47.17	18.72

Table 8
Optimal solutions of PSO with and without heuristic algorithms.

Setting (9 to 10 radiological technologists)	Optimal value obtained by IBM ILOG CPLEX	PSO solutions without both heuristic algorithms		PSO solutions with the decision tree method		PSO solutions with the greedy search algorithm		PSO solutions with both heuristic algorithms	
		Average objective function value	Standard deviation of objective function value	Average objective function value	Standard deviation of objective function value	Average objective function value	Standard deviation of objective function value	Average objective function value	Standard deviation of objective function value
1	0	7.66	0.56	1.05	0.28	1.48	0.18	0.14	0.11
2	0	7.97	0.47	0.92	0.26	1.48	0.23	0.12	0.10
3	0	7.64	0.49	0.93	0.19	1.46	0.18	0.15	0.11
4	0	7.75	0.73	0.98	0.24	1.60	0.23	0.15	0.09
5	0	7.81	0.62	0.89	0.23	1.53	0.22	0.15	0.08
6	0	7.78	0.57	0.98	0.22	1.51	0.27	0.12	0.08
7	0	7.51	0.64	0.88	0.21	1.36	0.21	0.15	0.09
8	0	7.52	0.69	0.86	0.25	1.62	0.28	0.12	0.08
9	0	7.64	0.46	0.96	0.25	1.55	0.15	0.15	0.11
10	0	7.68	0.71	0.94	0.24	1.51	0.21	0.19	0.11

Table 8 compares the optimal solution of PSO without both heuristic algorithms, PSO with the decision tree method, PSO with the greedy search algorithm, and PSO with both heuristic algorithms. The table shows that, among the 10 settings (9 to 10 radiological technologists), PSO with both heuristic algorithms had better solution quality, followed by PSO with the decision tree method, PSO with the greedy search algorithm, and PSO without both heuristic algorithms. This means that using the decision tree and greedy search algorithm could improve solution quality of BA, leading to obtain a near-optimal solution. In contrast, PSO without both heuristic algorithms fell into the local optimal solution with the range of [7.51, 7.97]. For PSO, the decision tree method had more impact on solution quality than the greedy search algorithm. This is different from the observation of BA.

Table 9 compares the solution time of PSO without both heuristic algorithms, PSO with the decision tree method, PSO with the greedy search algorithm, and PSO with both heuristic algorithms. The table shows that, among the 10 settings, PSO with the decision tree method had a shorter average solution time than PSO without the decision tree method. Using the decision tree method could generate a faster initial solution, thereby reducing the average solution time. Furthermore, among the 10 settings, PSO (with or without the decision tree method) without the greedy search algorithm had a shorter average solution time than PSO (with or without the decision tree method) with the greedy search algorithm. The reason for this finding was that the greedy search algorithm required an additional local search for each iteration of PSO. Furthermore, compared to the solution time by

Table 9
Solution time of PSO with and without heuristic algorithms.

Setting (9 to 10 radiological technologists)	Solution time of obtaining optimal value obtained by using IBM ILOG CPLEX (s)	PSO solutions without both heuristic algorithms		PSO solutions with the decision tree method		PSO solutions with the greedy search algorithm		PSO solutions with both heuristic algorithms	
		Average solution time (s)	Standard deviation of solution time	Average solution time (s)	Standard deviation of solution time	Average solution time (s)	Standard deviation of solution time	Average solution time (s)	Standard deviation of solution time
1	2	30.83	0.87	26.07	1.26	37.03	0.72	29.17	11.64
2	3	30.70	0.92	28.27	0.83	36.00	0.53	31.60	8.71
3	2	30.70	1.15	28.50	0.90	37.02	1.40	31.20	8.16
4	2	29.87	0.68	27.03	0.67	37.30	0.60	32.93	10.30
5	2	28.57	0.57	26.20	0.48	39.23	1.10	37.20	7.08
6	2	28.13	0.57	27.67	0.80	37.63	0.85	34.67	10.16
7	2	26.33	0.55	24.83	0.87	37.33	1.12	34.47	6.67
8	2	29.60	1.35	27.30	0.75	39.33	0.92	33.03	8.49
9	2	28.87	1.01	25.80	0.55	37.53	0.82	33.00	7.71
10	2	29.53	0.97	26.37	0.56	38.07	0.94	34.57	5.30

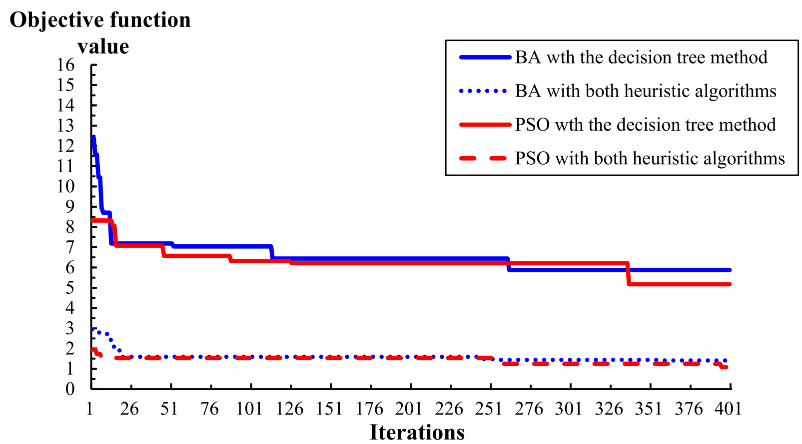

Fig. 9(a). Convergence curve of BA and PSO with the decision tree method and with both heuristic algorithms for the first run.

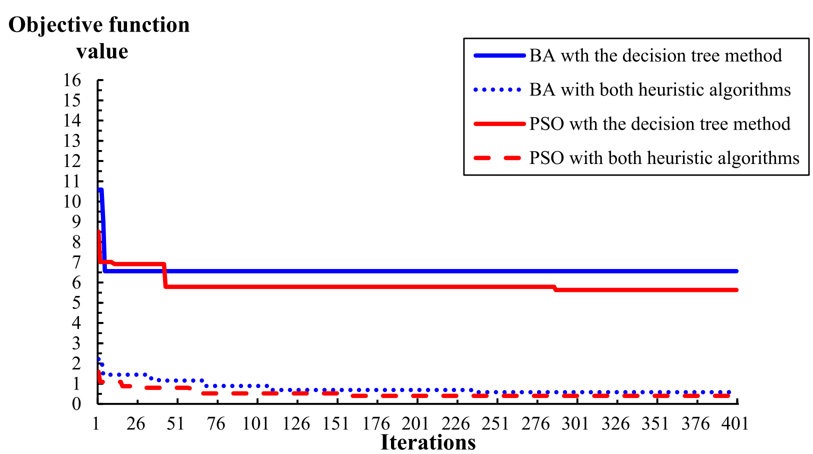

Fig. 9(b). Convergence curve of BA and PSO with the decision tree method and with both heuristic algorithms for the second run.

using IBM ILOG CPLEX, PSO with and without heuristic algorithms still took longer to obtain a near-optimal solution, but PSO with and without heuristic algorithms took less solution time than BA with and without heuristic algorithms.

4.4.2. The second testing scenario

To test large-scale rosters, there are 20 to 38 radiological technologists reported in the NRP literature [10]. For the daily requirement of 20 to 38 radiological technologists, the initial solutions generated by the random generation method take too long. Thus, this study performed only two analyses: BA with initial solutions generated by the decision tree method with and without the greedy search algorithm; and PSO with initial solutions generated by the decision tree method with and without the greedy search algorithm. The results are presented in Tables A.5 and A.6. Figs. 9(a) and 9(b) show that the convergence curve of the two sampled runs of BA and PSO with the decision tree method and with two heuristic algorithms. Although increasing the maximum number of iterations may improve solution quality in the second scenario (i.e., large-scale rosters), this research set the maximum number of iterations at 400 due to the trade-offs between solution quality and solution time.

Based on the numerical results in Table 10, this study validated that either the decision tree method for initial solutions or the

Table 10
Optimal solutions of BA and PSO, which was integrated with the decision tree method, with and without the greedy search algorithm.

Setting (20 to 38 radiological technologists)	Optimal value obtained by IBM ILOG CPLEX	BA solutions with the decision tree method		BA solutions with both heuristic algorithms		PSO solutions with the decision tree method		PSO solutions with both heuristic algorithms	
		Average objective function value	Standard deviation of objective function value	Average objective function value	Standard deviation of objective function value	Average objective function value	Standard deviation of objective function value	Average objective function value	Standard deviation of objective function value
1	0.2	6.22	0.62	1.13	0.16	5.87	0.36	1.05	0.15
2	0	6.51	0.56	0.62	0.15	4.98	0.42	0.31	0.10
3	0	5.77	0.43	0.66	0.14	5.43	0.35	0.53	0.11
4	0	5.96	0.50	0.57	0.14	5.31	0.40	0.42	0.11
5	0.2	7.33	0.43	1.18	0.15	7.32	0.40	1.22	0.17
6	0	6.21	0.54	0.70	0.18	5.49	0.41	0.42	0.13
7	0.2	7.02	0.54	1.28	0.14	6.09	0.46	1.14	0.15
8	0.2	8.61	0.52	1.84	0.17	7.19	0.26	1.78	0.26
9	0	7.44	0.48	0.91	0.19	6.20	0.41	0.55	0.13
10	0	6.80	0.37	1.28	0.18	5.54	0.45	1.13	0.21

Table 11
Solution time of BA and PSO, which was integrated with the decision tree method, with and without the greedy search algorithm.

Setting (20 to 38 radiological technologists)	Solution time of obtaining optimal value obtained by using IBM ILOG CPLEX (s)	BA solutions with the decision tree method		BA solutions with both heuristic algorithms		PSO solutions with the decision tree method		PSO solutions with both heuristic algorithms	
		Average solution time (s)	Standard deviation of solution time	Average solution time (s)	Standard deviation of solution time	Average solution time (s)	Standard deviation of solution time	Average solution time (s)	Standard deviation of solution time
1	73	228.73	13.96	234.10	13.23	230.97	16.44	265.90	15.86
2	6	228.47	13.67	237.87	14.84	226.47	18.22	270.13	4.99
3	4	231.33	12.27	224.37	8.16	242.87	7.38	250.20	20.22
4	5	231.40	10.76	226.37	9.18	221.73	14.70	234.27	5.81
5	8	238.77	12.68	245.23	12.91	206.17	2.76	233.07	5.91
6	6	213.07	12.05	220.07	10.15	195.70	4.48	215.23	5.95
7	103	233.13	11.19	227.53	13.15	237.47	4.32	279.47	4.20
8	57	240.40	10.91	239.00	9.20	230.23	17.33	242.90	9.32
9	14	227.37	16.29	215.03	9.07	221.03	16.29	231.20	8.95
10	5	238.57	14.11	237.67	11.90	210.23	4.21	243.87	13.32

greedy search algorithm for local searches enhanced solution quality for the BA and PSO. Furthermore, for the first testing scenario (9 to 10 radiological technologists), the BA and PSO combining the decision tree method for initial solutions with the greedy search algorithm for local searches could obtain the optimal solution. For the second testing scenario (20 to 38 radiological technologists), the BA and PSO combining the decision tree method for initial solutions with the greedy search algorithm for local searches indeed enhanced solution quality.

Table 11 compares the solution time of BA with the decision tree method, BA with both heuristic algorithms, PSO with the decision tree method, and PSO with both heuristic algorithms. The table shows that, among the 10 settings, PSO with the decision tree method had a shorter average solution time than PSO with both heuristic algorithms, and that BA with the decision tree method (or BA with both heuristic algorithms) had a shorter average solution time than PSO with both heuristic algorithms. But, BA with the decision tree method had average solution time close to BA with both heuristic algorithms. Furthermore, when two scenarios were compared (9 to 10 radiological technologists vs. 20 to 38 radiological technologists), the solution time of BA with two heuristic algorithms increased from 55 to 231 s, and the solution time of PSO with two heuristic algorithms increased from 30 to 247 s. Based on this analysis, when the scale of the problem increased, the solution time of PSO with two heuristic algorithms increased more than the solution time of BA with two heuristic algorithms did.

4.5. Summary

The proposed heuristic algorithms, which are the decision tree method and the greedy search algorithm, can be incorporated into two metaheuristic algorithms to generate better rosters that satisfied government regulations and hospital policies, and reduced unsatisfied radiological technologists' shifts. The contributions of this research can be summarized as follows:

(1) The initial solutions generated by the decision tree method can be applied to metaheuristic algorithms, such as the BA and PSO, to expedite search and convergence.
(2) The greedy search algorithm developed in this study can be applied to metaheuristic algorithms, such as the BA and PSO, to enhance solution quality.
(3) The outcomes of different testing instances of the radiological technologist rostering problems validated the feasibility and effectiveness of the proposed heuristic algorithms. The algorithms proposed in this study resolves several problems of manual rostering in the case hospital; not only does it shorten the time required to generate rosters, it also prevents manual errors. Therefore, it is an effective tool for administrators tasked with generating better monthly rosters.

5. Conclusions and future research

This study developed two heuristic algorithms integrated with metaheuristic algorithms to generate better initial solutions in less time and improve solutions' quality of solving optimization problems containing soft and hard constraints. The algorithms were coupled with BA or PSO to enhance solution quality. They were applied to a case radiological technologist rostering problem to confirm their feasibility.

Soft and hard constraints that satisfied government regulations and hospital policies, and reduced unsatisfied radiological technologists' shifts were formalized for the emergency diagnostic radiology unit of a research hospital. The proposed algorithms not only enhanced radiological technologists' rostering satisfaction but also reduced the time required to create rosters.

The proposed heuristic algorithms were applied to a radiological technologist rostering problem. The outcomes proved the feasibility of the algorithms. This study can be concluded with the following two points:

(1) The proposed heuristic algorithms can be applied in conjunction with two metaheuristic algorithms to solve radiological technologist rostering problems. These problems belong to typical rostering problems and include designing parameters, generating initial solutions, designing roster updates and recovery mechanisms, and applying greedy search. Results showed that applying the proposed heuristic algorithms to two metaheuristic algorithms for the radiological technologist rostering problem generated rosters that were more satisfactory to the radiological technologists.

(2) Generating radiological technologist rosters that satisfied various hard constraints, such as government regulations and hospital policies, reduced the human error that exists in manual rostering. The results presented in Section 4.4 show that, by applying the initial solution generated by the decision tree method and the greedy search algorithm developed in this study to BA and PSO, the algorithms were able to produce better solutions that approximated the optimal solution. These results proved the feasibility of the proposed heuristic algorithms.

This study has three limitations:

(1) Without loss of the generality, this research considers only the hard constraints, Eqs. (19) through (23), and the soft constraints, Eqs. (24) through (26), for a monthly medical staff schedule. In the literature review, other hard constraints adopted by the NRPs are the limitations on the number of the evening and night shifts for each nurse, number of the required senior nurses for each shift, the total number of shifts that each nurse has in two weeks, and the number of days off each nurse in four weeks. These hard constraints differ by medical department and hospital, making it difficult to develop a mathematical model or a software package that is capable of accommodating all hard and soft constraints for different NRPs. That is why many practical schedulers still prefer to generate their weekly or monthly schedules for medical staff manually.

(2) The proposed algorithms were applied to a radiological technologist rostering problem. They were not used to solve other problems, such as school scheduling/timetabling [12,13,29]. It may be more difficult to generate useful decision trees for school scheduling/timetabling problems because of the greater number of subjects, timeslots, and hard and soft constraints.

(3) Some constraints were excluded. For example, this study did not account for the conflicts or effects that arise when generating rosters for multiple months. This continuity limitation could be addressed in future research [1,51,52].

This study proposes two suggestions to facilitate future research:

(1) The proposed heuristic algorithms have been tested only on a radiological technologist rostering problem. Problems in applying the proposed algorithms to other similar problems, such as school timetabling [12,13], have yet to be addressed. Although the validation of the proposed algorithms was successful in this study, it should be applied to other problems to ensure reproducibility.

(2) The performance of the proposed heuristic algorithms has been measured for only one research hospital. Adjustments to the algorithms must be made when soft constraints change. Therefore, a universal model should be developed in the future so that the algorithms can be applied to more hospitals.

CRediT authorship contribution statement

Ping-Shun Chen: Funding acquisition, Project administration, Supervision, Writing - review & editing, Methodology, Conceptualization, Formal analysis. **Zhi-Yang Zeng:** Validation, Investigation, Resources, Data curation, Visualization, Writing - original draft, Software.

Declaration of competing interest

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

Acknowledgments

This research is supported by the Ministry of Science and Technology, Taiwan, R.O.C. under contract no. MOST 107-2221-E-033-053 and MOST 108-2221-E-033-008 .

Appendix

See Tables A.1–A.6.

Table A.1
Results for BA with initial solutions generated by the decision tree method with and without the greedy search algorithm.

Setting (9 to 10 radiological technologists)	Optimal value obtained by IBM ILOG CPLEX	BA solutions with using the greedy search algorithm				BA solutions without using the greedy search algorithm			
		Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value	Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value
1	0	0.03	0.05	0.25	0	1.12	0.19	1.53	0.74
2	0	0.05	0.07	0.32	0	1.11	0.28	1.65	0.55
3	0	0.06	0.03	0.10	0	1.19	0.25	1.59	0.59
4	0	0.05	0.05	0.32	0	1.22	0.23	1.67	0.75
5	0	0.05	0.02	0.10	0	1.26	0.20	1.50	0.61
6	0	0.02	0.02	0.05	0	1.13	0.19	1.39	0.66
7	0	0.07	0.07	0.42	0	1.25	0.22	1.74	0.79
8	0	0.03	0.03	0.13	0	1.22	0.22	1.58	0.73
9	0	0.07	0.10	0.48	0	1.14	0.24	1.61	0.65
10	0	0.05	0.07	0.35	0	1.26	0.19	1.55	0.81

Table A.2
Effects of two sources of initial solutions on the results of PSO.

Setting (9 to 10 radiological technologists)	Optimal value obtained by IBM ILOG CPLEX	PSO solutions with using the decision tree method for initial solutions				PSO solutions with using random generation method for initial solutions			
		Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value	Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value
1	0	1.05	0.28	1.59	0.30	7.66	0.56	8.85	6.78
2	0	0.92	0.26	1.52	0.51	7.97	0.47	8.81	6.88
3	0	0.93	0.19	1.27	0.43	7.64	0.49	8.53	6.25
4	0	0.98	0.24	1.34	0.45	7.75	0.73	8.98	5.85
5	0	0.89	0.23	1.28	0.51	7.81	0.62	9.04	6.43
6	0	0.98	0.22	1.50	0.49	7.78	0.57	8.94	6.17
7	0	0.88	0.21	1.39	0.43	7.51	0.64	8.58	5.73
8	0	0.86	0.25	1.38	0.45	7.52	0.69	9.09	5.26
9	0	0.96	0.25	1.69	0.48	7.64	0.46	8.38	6.32
10	0	0.94	0.24	1.43	0.48	7.68	0.71	8.80	6.45

Table A.3
Results for PSO with initial solutions generated randomly with and without the greedy search algorithm.

Setting (9 to 10 radiological technologists)	Optimal value obtained by IBM ILOG CPLEX	PSO solutions with using the greedy search algorithm				PSO solutions without using the greedy search algorithm			
		Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value	Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value
1	0	1.48	0.18	1.81	0.95	7.66	0.56	8.85	6.78
2	0	1.48	0.23	2.01	1.04	7.97	0.47	8.81	6.88
3	0	1.46	0.18	1.83	1.11	7.64	0.49	8.53	6.25
4	0	1.60	0.23	2.05	1.21	7.75	0.73	8.98	5.85
5	0	1.53	0.22	1.98	1.12	7.81	0.62	9.04	6.43
6	0	1.51	0.27	2.05	1.01	7.78	0.57	8.94	6.17
7	0	1.36	0.21	1.77	0.76	7.51	0.64	8.58	5.73
8	0	1.62	0.28	2.09	1.07	7.52	0.69	9.09	5.26
9	0	1.55	0.15	1.87	1.25	7.64	0.46	8.38	6.32
10	0	1.51	0.21	1.97	1.17	7.68	0.71	8.80	6.45

Table A.4
Results for PSO with initial solutions generated by the decision tree method with and without the greedy search algorithm.

Setting (9 to 10 radiological technologists)	Optimal value obtained by IBM ILOG CPLEX	PSO solutions with using the greedy search algorithm				PSO solutions without using the greedy search algorithm			
		Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value	Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value
1	0	0.14	0.11	0.52	0	1.05	0.28	1.59	0.30
2	0	0.12	0.10	0.34	0	0.92	0.26	1.52	0.51
3	0	0.15	0.11	0.37	0	0.93	0.19	1.27	0.43
4	0	0.15	0.09	0.37	0	0.98	0.24	1.34	0.45
5	0	0.15	0.08	0.35	0	0.89	0.23	1.28	0.51
6	0	0.12	0.08	0.35	0	0.98	0.22	1.50	0.49
7	0	0.15	0.09	0.38	0	0.88	0.21	1.39	0.43
8	0	0.12	0.08	0.30	0	0.86	0.25	1.38	0.45
9	0	0.15	0.11	0.43	0	0.96	0.25	1.69	0.48
10	0	0.19	0.11	0.45	0	0.94	0.24	1.43	0.48

Table A.5
Results for BA with initial solutions generated by the decision tree method with and without the greedy search algorithm.

Setting (20 to 38 radiological technologists)	Optimal value obtained by IBM ILOG CPLEX	BA solutions with using the greedy search algorithm				BA solutions without using the greedy search algorithm			
		Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value	Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value
1	0.2	1.13	0.16	1.54	0.83	6.22	0.62	7.16	4.86
2	0	0.62	0.15	0.89	0.23	6.51	0.56	7.41	5.46
3	0	0.66	0.14	0.95	0.42	5.77	0.43	6.41	4.94
4	0	0.57	0.14	0.81	0.28	5.96	0.50	6.72	4.46
5	0.2	1.18	0.15	1.50	0.91	7.33	0.43	8.11	6.23
6	0	0.70	0.18	0.97	0.32	6.21	0.54	7.14	4.72
7	0.2	1.28	0.14	1.55	0.91	7.02	0.54	7.75	5.39
8	0.2	1.84	0.17	2.20	1.55	8.61	0.52	9.38	7.61
9	0	0.91	0.19	1.23	0.43	7.44	0.48	8.20	6.45
10	0	1.28	0.18	1.63	0.87	6.80	0.37	7.51	5.91

Table A.6
Results for PSO with initial solutions generated by the decision tree method with and without the greedy search algorithm.

Setting (20 to 38 radiological technologists)	Optimal value obtained by IBM ILOG CPLEX	PSO solutions with using the greedy search algorithm				PSO solutions without using the greedy search algorithm			
		Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value	Average objective function value	Standard deviation of objective function value	Maximum objective function value	Minimum objective function value
1	0.2	1.05	0.15	1.36	0.75	5.87	0.36	6.44	5.09
2	0	0.31	0.10	0.51	0.05	4.98	0.42	5.67	3.75
3	0	0.53	0.11	0.77	0.28	5.43	0.35	6.03	4.69
4	0	0.42	0.11	0.60	0.19	5.31	0.40	5.91	4.49
5	0.2	1.22	0.17	1.51	0.94	7.32	0.40	8.17	6.57
6	0	0.42	0.13	0.73	0.20	5.49	0.41	6.26	4.47
7	0.2	1.14	0.15	1.34	0.76	6.09	0.46	6.69	4.75
8	0.2	1.78	0.26	2.19	1.22	7.19	0.26	7.87	6.62
9	0	0.55	0.13	0.85	0.29	6.20	0.41	6.81	5.13
10	0	1.13	0.21	1.58	0.69	5.54	0.45	6.35	4.38

References

[1] P. De Causmaecker, G.V. Berghe, A categorisation of nurse rostering problems, J. Sched. 14 (2011) 3–16.
[2] T. Osogami, H. Imai, Classification of various neighborhood operations for the nurse scheduling problem, in: ISAAC 2000, in: LNCS, vol. 1969, Springer-Verlag Berlin Heidelberg, 2000, pp. 72–83.
[3] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algorithms, third ed., The MIT Press, Cambridge, Massachusetts, 2009.
[4] C.W. Tsai, M.C. Chiang, A. Ksentini, M. Chen, Metaheuristic algorithms for healthcare: Open issues and challenges, Comput. Electr. Eng. 53 (2016) 421–434.
[5] M. Defraeye, I. Van Nieuwenhuyse, Staffing and scheduling under nonstationary demand for service: A literature review, Omega-Int. J. Manag. Sci. 58 (2016) 4–25.
[6] A. Ahmadi, M.S. Pishvaee, M.R.A. Jokar, A survey on multi-floor facility layout problems, Comput. Ind. Eng. 107 (2017) 158–170.
[7] B. Calis, S. Bulkan, A research survey: Review of AI solution strategies of job shop scheduling problem, J. Intell. Manuf. 26 (2015) 961–973.
[8] T.C. Wong, M. Xu, K.S. Chin, A two-stage heuristic approach for nurse scheduling problem: A case study in an emergency department, Comput. Oper. Res. 51 (2014) 99–110.
[9] P.S. Chen, Y.J. Lin, N.C. Peng, A two-stage method to determine the allocation and scheduling of medical staff in uncertain environments, Comput. Ind. Eng. 99 (2016) 174–188.
[10] I.X. Tassopoulos, I.P. Solos, G.N. Beligiannis, A two-phase adaptive variable neighborhood approach for nurse rostering, Comput. Oper. Res. 60 (2015) 150–169.
[11] M. Rajeswari, J. Amudhavel, S. Pothula, P. Dhavachelvan, Directed bee colony optimization algorithm to solve the nurse rostering problem, Comput. Intell. Neurosci. (2017) 6563498, 1–26.
[12] N. Pillay, A review of hyper-heuristics for educational timetabling, Ann. Oper. Res. 239 (2016) 3–38.
[13] C.K. Teoh, A. Wibowo, M.S. Ngadiman, Review of state of the art for metaheuristic techniques in academic scheduling problems, Artif. Intell. Rev. 44 (2015) 1–21.
[14] L. Sayiniec, M.O. Santos, A.M. Costa, Parallel local search algorithms for high school timetabling problems, European J. Oper. Res. 265 (2018) 81–98.
[15] N. Todorovic, S. Petrovic, Bee colony optimization algorithm for nurse rostering, IEEE Trans. Syst. Man Cybern. Syst. 43 (2013) 467–473.
[16] T.H. Wu, J.Y. Yeh, Y.M. Lee, A particle swarm optimization approach with refinement procedure for nurse rostering problem, Comput. Oper. Res. 54 (2015) 52–63.
[17] M.A. Awadallah, A.L. Bolaji, M.A. Al-Betar, A hybrid artificial bee colony for a nurse rostering problem, Appl. Soft Comput. 35 (2015) 726–739.
[18] A. Parisi, C.N. Jones, A two-stage stochastic programming approach to employee scheduling in retail outlets with uncertain demand, Omega-Int. J. Manag. Sci. 53 (2015) 97–103.
[19] J.F. Bard, Z.C. Shu, D.J. Morrice, L.K. Leykum, Annual block scheduling for internal medicine residents with 4+1 templates, J. Oper. Res. Soc. 67 (2016) 911–927.
[20] G. Erdogan, M. Haouari, M.O. Matoglu, O.O. Ozener, Solving a large-scale crew pairing problem, J. Oper. Res. Soc. 66 (2015) 1742–1754.
[21] A. Leggate, S. Sucu, K. Akartunali, R. van der Meer, Modelling crew scheduling in offshore supply vessels, J. Oper. Res. Soc. 69 (2018) 959–970.
[22] B. Akbarzadeh, G. Moslehi, M. Reisi-Nafchi, B. Maenhout, The re-planning and scheduling of surgical cases in the operating room department after block release time with resource rescheduling, European J. Oper. Res. 278 (2019) 596–614.
[23] A.K. Beheshti, S.R. Hejazi, A novel hybrid column generation-metaheuristic approach for the vehicle routing problem with general soft time window, Inform. Sci. 316 (2015) 598–615.
[24] S. Iqbal, M. Kaykobad, M.S. Rahman, Solving the multi-objective vehicle routing problem with soft time windows with the help of bees, Swarm Evol. Comput. 24 (2015) 50–64.
[25] C.C. Lin, L.P. Hung, W.Y. Liu, M.C. Tsai, Jointly rostering, routing, and rerostering for home health care services: A harmony search approach with genetic, saturation, inheritance, and immigrant schemes, Comput. Ind. Eng. 115 (2018) 151–166.
[26] Z. Beheshti, S.M.H. Shamsuddin, A review of population-based meta-heuristic algorithm, Int. J. Adv. Soft Comput. Appl. 5 (2013) 1–35.
[27] R. Jans, Z. Degraeve, Meta-heuristics for dynamic lot sizing: A review and comparison of solution approaches, European J. Oper. Res. 177 (2007) 1855–1875.
[28] G.H.G. Fonseca, H.G. Santos, Variable neighborhood search based algorithms for high school timetabling, Comput. Oper. Res. 52 (2014) 203–208.
[29] V.I. Skoullis, I.X. Tassopoulos, G.N. Beligiannis, Solving the high school timetabling problem using a hybrid cat swarm optimization based algorithm, Appl. Soft Comput. 52 (2017) 277–289.
[30] E.K. Burke, T. Curtois, R. Qu, G. Vanden Berghe, A scatter search methodology for the nurse rostering problem, J. Oper. Res. Soc. 61 (2010) 1667–1679.
[31] C.C. Lin, J.R. Kang, D.J. Chiang, C.L. Chen, Nurse scheduling with joint normalized shift and day-off preference satisfaction using a genetic algorithm with immigrant scheme, Int. J. Distrib. Sens. Netw. (2015) 595419, 1-10.
[32] S.H. Jin, H.Y. Yun, S.J. Jeong, K.S. Kim, Hybrid and cooperative strategies using harmony search and artificial immune systems for solving the nurse rostering problem, Sustainability 9 (2017) 1090, 1–19.
[33] Z.R. Zheng, X.Y. Liu, X.J. Gong, A simple randomized variable neighbourhood search for nurse rostering, Comput. Ind. Eng. 110 (2017) 165–174.
[34] R. Liu, X.L. Xie, Physician staffing for emergency departments with time-varying demand, Inf. J. Comput. 30 (2018) 588–607.
[35] X.-S. Yang, A new metaheauristic bat-inspired algorithm, in: Nature Inspired Cooperative Strategies for Optimization (NICSO 2010), in: studies in Computational Intelligence, vol. 284, Springer Berlin Heidelberg, 2010, pp. 65–74.
[36] S. Yılmaz, E.U. Küçüksille, A new modification approach on bat algorithm for solving optimization problems, Appl. Soft Comput. 28 (2015) 259–275.
[37] J. Kennedy, R. Eberhart, Particle swarm optimization, in: IEEE International Conference on Neural Networks, Perth, Austria, 199..
[38] P. Smet, T. Wauters, M. Mihaylov, G. Vanden Berghe, The shift minimisation personnel task scheduling problem: A new hybrid approach and computational insights, Omega-Int. J. Manag. Sci. 46 (2014) 64–73.
[39] S. Agrali, Z.C. Taskin, A.T. Unal, Employee scheduling in service industries with flexible employee availability and demand, Omega-Int. J. Manag. Sci. 66 (2017) 159–169.
[40] P.D. Wright, S. Mahar, Centralized nurse scheduling to simultaneously improve schedule cost and nurse satisfaction, Omega-Int. J. Manag. Sci. 41 (2013) 1042–1052.
[41] S. Mattia, F. Rossi, M. Servilio, S. Smriglio, Staffing and scheduling flexible call centers by two-stage robust optimization, Omega-Int. J. Manag. Sci. 72 (2017) 25–37.
[42] S. Topaloglu, A multi-objective programming model for scheduling emergency medicine residents, Comput. Ind. Eng. 51 (2006) 375–388.
[43] M.G. Guler, K. Idin, E.Y. Guler, A goal programming model for scheduling residents in an anesthesia and reanimation department, Expert Syst. Appl. 40 (2013) 2117–2126.

[44] F.F.C. Pan, Using analytic hierarchy process to identify the nurses with high stress-coping capability: Model and application, Iranian J. Publ. Health 43 (2014) 273–281.
[45] I. Khennak, H. Drias, Bat-inspired algorithm based query expansion for medical web information retrieval, J. Med. Syst. 41 (2017) 1–16, 34.
[46] J.Y. Li, S. Fong, L.S. Liu, N. Dey, A.S. Ashour, L. Moraru, Dual feature selection and rebalancing strategy using metaheuristic optimization algorithms in X-ray image datasets, Multimedia Tools Appl. 78 (2019) 20913–20933.
[47] T. Kaur, B.S. Saini, S. Gupta, A novel feature selection method for brain tumor MR image classification based on the Fisher criterion and parameter-free bat optimization, Neural Comput. Appl. 29 (2018) 193–206.
[48] R. Sheikhpour, M.A. Sarram, R. Sheikhpour, Particle swarm optimization for bandwidth determination and feature selection of kernel density estimation based classifiers in diagnosis of breast cancer, Appl. Soft Comput. 40 (2016) 113–131.
[49] T.Y. Tan, L. Zhang, C.P. Lim, Intelligent skin cancer diagnosis using improved particle swarm optimization and deep learning models, Appl. Soft Comput. 84 (2019) 1–19, 105725.
[50] T.M.C. Jegan, D. Ravindran, Electrochemical machining process parameter optimization using particle swarm optimization, Comput. Intell. 33 (2017) 1019–1037.
[51] J. Van den Bergh, J. Beliën, P. De Bruecker, E. Demeulemeester, L. De Boeck, Personnel scheduling: A literature review, European J. Oper. Res. 226 (2013) 367–385.
[52] B. Cheang, H. Li, A. Lim, B. Rodrigues, Nurse rostering problems - A bibliographic survey, European J. Oper. Res. 151 (2003) 447–460.