

A DIFFERENTIAL EVOLUTION ALGORITHM DEVELOPED FOR  
A NURSE SCHEDULING PROBLEM

P. Shahnazari-Shahrezaei<sup>1\*</sup>, R. Tavakkoli-Moghaddam<sup>2</sup>, M. Azarkish<sup>3</sup> &  
A. Sadeghnejad-Barkousaraie<sup>4</sup>

<sup>1</sup> Department of Industrial Engineering  
Science and Research Branch, Islamic Azad University, Tehran, Iran  
parisa\_shahnazari@iaufb.ac.ir

<sup>2</sup> Department of Industrial Engineering  
University of Tehran, Tehran, Iran  
tavakoli@ut.ac.ir

<sup>3,4</sup> Department of Industrial Engineering and Center of Excellence for  
Intelligence-Based Experimental Mechanics  
University of Tehran, Tehran, Iran

<sup>3</sup> shadi\_azarkish@yahoo.com; <sup>4</sup> azar\_sadeghnezhad@yahoo.com

ABSTRACT

Nurse scheduling is a type of manpower allocation problem that tries to satisfy hospital managers' objectives and nurses' preferences as much as possible by generating fair shift schedules. This paper presents a nurse scheduling problem based on a real case study, and proposes two meta-heuristics - a differential evolution algorithm (DE) and a greedy randomised adaptive search procedure (GRASP) - to solve it. To investigate the efficiency of the proposed algorithms, two problems are solved. Furthermore, some comparison metrics are applied to examine the reliability of the proposed algorithms. The computational results in this paper show that the proposed DE outperforms the GRASP.

OPSOMMING

Verpleegsterskedulering is 'n mannekragtoedelingsprobleem wat deur regverdigte skofskedules beide hospitaalbestuurders se teikens en verpleegpersoneel se voorkeure bevredig. Dié artikel handel oor 'n verpleegsterskeduleringprobleem wat gebaseer is op 'n werklike gevallestudie en stel twee metaheuristieke voor om die probleem op te los - 'n differensiaal evolusionêre metode en 'n gulsige, ewekansige, aanpasbare soekprosedure. Om die doeltreffendheid van die voorgestelde algoritmes te ondersoek word twee probleme opgelos. Verder word vergelykende maatstawwe gebruik om die betroubaarheid van die voorgestelde algoritmes te ondersoek. Die berekende resultate in die artikel toon dat die differensiaal evolusionêre algoritme beter vaar as die gulsige, ewekansige, aanpasbare soekprosedure.

---

\* Corresponding author.

<sup>1</sup> The author was enrolled for a PhD degree (Industrial Engineering) in the Department of Industrial Engineering, Islamic Azad University - Science and Research Branch.

## 1. INTRODUCTION

Manpower scheduling is a challenging problem in which an extensive field of research has been done since the 1970s, dealing with different aspects in management sciences and operational research. It is a managerial process in which the required manpower level for each craft or skill is selected to complete the work in the most effective manner. Proper scheduling, therefore, leads to increased efficiency, maximum use of the capacity of the resources, less time to finish the work, and increased profitability in the organisation.

Manpower scheduling is classified into three groups: individual scheduling (nurse, physician, surgery, and the like); personnel scheduling; and group scheduling (crew, etc). Providing a schedule that almost satisfies all constraints and objectives is often troublesome, and needs specialised knowledge. Different classifications of manpower are observed in industrial and service environments, including homogeneous and heterogeneous, part-time and full-time, skilled and unskilled workforces, and so on [21,36,49,52]. Manpower scheduling and simultaneous allocation have been much considered by researchers. For instance, Cerulli et al. [17] introduced a mathematical model for scheduling and allocating workforces. Emmons & Fuh [25] scheduled full-time and part-time workforces, taking vacations and weekends into account. Blochliger [13] also presented a tutorial for staff scheduling problems.

Selecting a suitable solution method is another important matter in scheduling. Alfares [6] suggested a two-phase algorithm based on mathematical models to schedule manpower, taking cyclic days-off into consideration. Lagodimos et al. [32] used greedy heuristic algorithms. Akjiratikarl et al. [5] introduced a particle swarm optimisation-based algorithm to schedule home care workers in the UK to minimise the total distance travelled by all workers. Musliu et al. [37] also stated that network flow, a combination of management science and artificial intelligence techniques, and mixed approaches joining constraint satisfaction and local improvement algorithms, are just some of the approaches that have been applied so far to solve manpower scheduling problems. Ernst et al. [26] examined literature reviews, applications, models, and algorithms of manpower scheduling, and finally proposed some points for future research.

As mentioned before, a nurse scheduling problem (NSP) is a kind of manpower allocation problem that deals with demand oscillations on different days and shifts. This problem requires hospitals to use their nurses efficiently. On the other hand, shift scheduling must be done so that it provides the opportunity for all nurses to have a safe social life in order to maintain their morale.

An NSP tries to distribute the workload fairly among nurses, reduce the costs, and satisfy both the nurses' preferences and those of the hospital managers. Sometimes these objectives conflict with each other. Nurses' preferences may not be completely satisfied and hospital managers may be forced to pay some penalties. Since hospital managers are often under pressure to control costs, satisfy the nurses, and deliver high-quality services, developing an appropriate nurse scheduling can balance these pressures and use the nurses more effectively.

Over the years, a wide range of models and solution approaches have been developed to deal with NSP. An extensive overview of NSP can be found in Cheang et al. [18] and Burke et al. [15].

As a general rule, nurse scheduling can be categorised into centralised, self, and cyclical scheduling. Two major advantages of centralised scheduling [24,44,46,54] are: (1) fairness to employees through consistent, objective, and impartial application of policies, and (2) opportunities for cost containment through better use of existing resources. When head nurses or unit managers are responsible for generating the schedules locally, it is called 'unit scheduling' [2,14,23,33,34,45]. In this case, nurses usually get more personalised

attention. Self-scheduling is a manual process [35,43] that takes more time than automatic scheduling. Nurses cooperate and are asked for advice. It is usually performed by the personnel members themselves and coordinated by the head nurse of a ward. In cyclical scheduling, each nurse works a cycle of a number of weeks [11,30]. Nurses know their schedule in advance, and the same shift schedule patterns are used over an extended time. Unfortunately, this type of scheduling cannot incorporate many flexible features or personal preferences unless major modifications take place.

Over the last few decades, various approaches have been applied to solve NSP. The more common methods that are applied are mathematical programming, meta-heuristic, and artificial intelligence approaches.

Mathematical programming models are employed to solve NSP using traditional methods that guarantee to find an optimal solution. Azaiez & Al Sharif [9] developed a 0-1 linear goal programming model for nurse scheduling in Riyadh Al-Kharj hospital (in Saudi-Arabia) that considered hospital objectives and nurses' preferences. Bard & Purnomo [10] used a column generation approach, including a combination of integer programming and heuristics, to solve a multi-objective nurses' preference scheduling model. Glass & Knight [28] solved four nurse rostering benchmark problems using mixed integer linear programming. They reduced the solution space of the problems and obtained optimal solutions for all instances within a practical execution time. Topaloglu & Selim [50] presented a novel multi-objective integer programming model for NSP. They converted the presented model to three fuzzy goal programming models using different fuzzy solution approaches.

In real-size problems, mathematical programming approaches are faced with computational difficulties due to increasing complexity. To overcome this, meta-heuristic methods have been developed by some researchers. Meta-heuristics strive to obtain good solutions within a reasonable time, although they cannot be guaranteed to find optimal solutions. Meta-heuristics have been employed in solving NSP since the 1990s. Bellanti et al. [12] suggested a tabu search procedure completed by a greedy procedure to avoid generating infeasible solutions to the NSP. Aickelin & Dowsland [3] presented a genetic algorithm coupled with a decoding routine for solving an NSP at a major UK hospital. Gutjahr & Rauner [29] used an ant colony optimisation approach for a dynamic regional NSP at the Vienna hospital compound. Chiamonte & Chiamonte [19] recommended a new heuristic using a competitive agent-based negotiation that focuses on nurse preferences. Tsai & Li [51] extended a two-stage mathematical modeling for an NSP using genetic algorithms. Burke et al. [16] combined integer programming and variable neighborhood search to solve a highly-constrained NSP.

Artificial intelligence is another approach to solving NSP. Meyer auf'm Hofe [34] applied hierarchical constraint programming for nurse scheduling where legal regulations and the preferences of nurses were hard and soft constraints respectively. Abdennadher & Schlenker [1] presented an interactive constrained based scheduler that assigned a working shift to each nurse on each day of a planning horizon.

NSP is a complex combinatorial optimisation problem. Factors that make this problem complex are the organisational structure and characteristics, work agreements, nurses' recruitment, different skill levels of nurses, substitution of skill levels, working preferences, patient necessities, circumstances in particular nursing units, and so forth. Since most NSPs in the real world are NP-hard [19,31,39,55], traditional methods (e.g. mathematical programming) are generally used to solve small problem instances. The high level of computational complexity of NSP has been widely recognised in the nurse scheduling literature, and has led some authors to develop meta-heuristic solution methods.

An almost new meta-heuristic algorithm, called differential evolution (DE), is used to solve the given problem in this paper. DE was first introduced by Storn & Price [47,48]. DE is a

mathematical global optimisation method, based on population generation, which tries to minimise the non-linear and non-differentiable continuous space functions as much as possible. In order to examine the performance of the proposed DE algorithm, the problem is also solved by another meta-heuristic - the greedy randomised adaptive search procedure (GRASP) - that was introduced by Feo & Resende [27] and Resende [42]. The results are subsequently compared.

The rest of this paper is organised as follows. Section 2 deals with the formulation of the proposed nurse scheduling model. An effective DE algorithm to solve the problem, and a GRASP algorithm to compare the related results, are presented in Section 3. The computational results and validation of the proposed solution methods are found in Section 4. Finally, Section 5 provides some concluding remarks.

## 2. FORMULATION OF PROPOSED NURSE SCHEDULING MODEL

The proposed nurse scheduling model is based on the rules used in the maternity ward of Sina Hospital in Isfahan City, Iran, to schedule nurses in that ward. Nurses are assigned to work in three shifts: a six-hour morning shift, a six-hour afternoon shift, and a 12-hour night shift. The planning horizon is 35 days (five weeks). Nurses are classified into three skill levels: nurse, auxiliary nurse, and co-auxiliary nurse. The highest and lowest skill levels are 'nurse' and 'co-auxiliary nurse' respectively. Each nurse at a higher skill level can be assigned to work at his/her real skill level, or at any lower skill level, but not more than one skill level in a shift. If a nurse is assigned to work at a lower skill level than his/her real skill, hospital managers are forced to pay a penalty. There are limitations on the daily, Friday, weekly, and monthly work hours for nurses. Deviation from the determined weekly work hours is permitted if needed. The total number of required nurses at any skill level in each shift per day is pre-determined. Each nurse cannot be assigned to work for more than 12 hours consecutively. Therefore, nurses cannot be assigned to work the afternoon and night shifts of the same day, or a night shift of a considered day and the morning shift of the next day successively. Nurses are limited to a maximum number of night shifts during the planning period. Also, each nurse who works three night shifts in a row should be off for at least two days to rest. From an efficiency point of view, nurses should not be assigned to work between the two days off, but deviation from this rule is permitted if needed. Nurses prefer to rest for some days or for some shifts determined by them at the beginning of planning. But sometimes, due to the lack of qualified nurses for a shift, they may be assigned to work the days or shifts when they would have preferred to rest.

### 2.1 Notations

The following notations are used in the presented model.

#### 2.1.1 Indices

$i$	Index for the nurses, ( $i=1, \dots, I$ )
$k$	Index for the days, ( $k=1, \dots, K$ )
$lk$	Index for the last day of each week during the scheduling period, ( $lk=7, \dots, LK$ ); in this case: ( $lk=7, 14, 21, 28, 35$ ) and $ LK $ is equal to cardinality (size) of set $LK$ (e.g., $ LK =5$ ).
$j$	Index for the shifts, ( $j=1, \dots, J$ ); in this case: ( $j = 1$ : Morning, 2: Afternoon, 3: Night)
$s$	Index for the skill levels, ( $s=1, \dots, S$ ); in this case: ( $s = 1$ : Nurse, 2: Auxiliary nurse, 3: Co-auxiliary nurse)

#### 2.1.2 Sets

$I$	Set of nurses
$K$	Set of days during the scheduling period
$LK$	Set of the last days of all weeks during the scheduling period
$J$	Set of shifts
$S$	Set of skill levels

### 2.1.3 Parameters

$h_{kj}$	Length of shift $j$ on day $k$
$dh_{min}$	Lower bound on the total number of hours worked by each nurse in a day
$dh_{max}$	Upper bound on the total number of hours worked by each nurse in a day
$wh_{min}$	Lower bound on the total number of hours worked by each nurse during a week
$wh_{max}$	Upper bound on the total number of hours worked by each nurse during a week
$fh_{min}$	Lower bound on the total number of hours worked by each nurse on Fridays during the planning period
$fh_{max}$	Upper bound on the total number of hours worked by each nurse on Fridays during the planning period
$mh_{min}$	Lower bound on the total number of hours worked by each nurse during the planning period
$mh_{max}$	Upper bound on the total number of hours worked by each nurse during the planning period
$RN_{kjs}$	Total number of required nurses at skill level $s$ in shift $j$ on day $k$
$RSL^i$	Real skill level of nurse $i$
$Max - Night$	Maximum number of night shifts that each nurse can be assigned to work during the planning period
$Penl$	A penalty coefficient for assignment at lower skill level
$k_i$	Set of days that nurse $i$ is interested in taking a rest in some/all shifts
$J_{k_i}$	Set of shifts of day $k_i$ that nurse $i$ is interested in taking a rest
$ASL_{kjs}^i$	1, if nurse $i$ can be assigned to work at his/her real skill level or at any lower skill level $s$ in shift $j$ on day $k$ ; otherwise 0
$d_{k_2}^{1i}$	Deviation from lower bound on the total number of hours worked by nurse $i$ during a week
$d_{k_2}^{2i}$	Deviation from upper bound on the total number of hours worked by nurse $i$ during a week
$d_k^{3i}$	Deviation from off-on-off for nurse $i$
$d_{kj}^{4i}$	Deviation from days/shifts that nurse $i$ is interested in taking a rest

### 2.1.4 Decision variable

$x_{kjs}^i$	1, if nurse $i$ is assigned to work at skill level $s$ in shift $j$ on day $k$ ; otherwise 0
$q_k^i$	1, if nurse $i$ is assigned to work three successive night shifts on days $k-2$ , $k-1$ , and $k$ ; otherwise 0

## 2.2 Objective functions

- The first objective is to minimise the sum of deviations from lower and upper bounds on the total number of hours worked by nurses during a week:

$$Z_1 = \text{Min} \sum_i \sum_{k_2=1}^{|LK|} (d_{k_2}^{1i} + d_{k_2}^{2i}) \quad (1)$$

- The second objective is to minimise the sum of deviations from off-on-offs for nurses:

$$Z_2 = \text{Min} \sum_i \sum_{k=2}^{7|LK|-1} d_k^{3i} \quad (2)$$

- The third objective is to minimise the sum of deviations from days/shifts that nurses are interested in taking a rest:

$$Z_3 = \text{Min} \sum_i \sum_{k \in k_i} \sum_{j \in J_{k_i}} d_{kj}^{4i} \quad (3)$$

- The fourth objective is to minimise the nurses' assignment at lower skill levels than their real skill:

$$Z_4 = \text{Min} \sum_i \sum_k \sum_j \sum_s [(s - RSL^i) * x_{kjs}^i * \text{Penl}] \quad (4)$$

### 2.3 Constraints

In the presented model, constraints are divided into hard constraints that must be satisfied in all cases, and soft constraints that may be violated so that a feasible schedule can be obtained.

#### 2.3.1 Hard constraints

- Lower and upper bound on the total number of hours worked by each nurse on a day:

$$\sum_j \sum_s h_{kj} x_{kjs}^i \geq dh_{min} \quad ; \forall i, k \quad (5)$$

$$\sum_j \sum_s h_{kj} x_{kjs}^i \leq dh_{max} \quad ; \forall i, k \quad (6)$$

- Lower and upper bound on the total number of hours worked by each nurse during the planning period:

$$\sum_k \sum_j \sum_s h_{kj} x_{kjs}^i \geq mh_{min} \quad ; \forall i \quad (7)$$

$$\sum_k \sum_j \sum_s h_{kj} x_{kjs}^i \leq mh_{max} \quad ; \forall i \quad (8)$$

- Lower and upper bound on the total number of hours worked by each nurse on Fridays during the planning period:

$$\sum_{k \in LK} \sum_j \sum_s h_{kj} x_{kjs}^i \geq fh_{min} \quad ; \forall i \quad (9)$$

$$\sum_{k \in LK} \sum_j \sum_s h_{kj} x_{kjs}^i \leq fh_{max} \quad ; \forall i \quad (10)$$

- Each nurse can be assigned to work at his/her real skill level or any lower skill level in each shift per day:

$$x_{kjs}^i \leq ASL_{kjs}^i \quad ; \forall i, k, j, s \quad (11)$$

- Total number of required nurses at any skill level in each shift per day:

$$\sum_i x_{kjs}^i = RN_{kjs} \quad ; \forall k, j, s \quad (12)$$

- Each nurse cannot be assigned to work for more than 12 hours consecutively:

$$\sum_s \sum_{j \in \text{Afternoon}} x_{kjs}^i + \sum_s \sum_{j \in \text{Night}} x_{kjs}^i \leq 1 \quad ; \forall i, k \quad (13)$$

$$\sum_s \sum_{j \in \text{Night}} x_{kjs}^i + \sum_s \sum_{j \in \text{Morning}} x_{(k+1)js}^i \leq 1 \quad ; \forall i, k \quad (14)$$

$$\sum_j \sum_s x_{kjs}^i \leq 2 \quad ; \forall i, k \quad (15)$$

- Each nurse cannot be assigned to work at more than one skill level in each shift per day:

$$\sum_s x_{kjs}^i = 1 \quad ; \forall i, k, j \quad (16)$$

- Maximum number of night shifts that each nurse can be assigned to work during the planning period:

$$\sum_k \sum_s x_{k(j \in Night)_s}^i \leq Max - Night \quad ; \forall i \quad (17)$$

- Each nurse cannot be assigned to work more than three successive night shifts:

$$\sum_{k=k_1}^{k_1+3} \sum_s x_{k(j \in Night)_s}^i \leq 3 \quad ; \forall i, k_1 \in \{1, \dots, 7|LK| - 3\} \quad (18)$$

- Each nurse should be off for at least two days after working three successive night shifts:

$$q_k^i - \sum_s x_{(k-2)(j \in Night)_s}^i \leq 0 \quad ; \forall i, k \in \{3, \dots, 7|LK|\} \quad (19)$$

$$q_k^i - \sum_s x_{(k-1)(j \in Night)_s}^i \leq 0 \quad ; \forall i, k \in \{3, \dots, 7|LK|\} \quad (20)$$

$$q_k^i - \sum_s x_{k(j \in Night)_s}^i \leq 0 \quad ; \forall i, k \in \{3, \dots, 7|LK|\} \quad (21)$$

$$q_k^i - \sum_s x_{(k-2)(j \in Night)_s}^i - \sum_s x_{(k-1)(j \in Night)_s}^i - \sum_s x_{k(j \in Night)_s}^i \geq -2 \quad ; \forall i, k \in \{3, \dots, 7|LK|\} \quad (22)$$

The constraints (19) to (22) ensure that:

$q_k^i = 1$ , if nurse  $i$  is assigned to work three successive night shifts on days  $k-2$ ,  $k-1$ , and  $k$ ; otherwise 0.

Then, the rule is verified by adding the following constraints:

$$\sum_s x_{(k-2)(j \in Night)_s}^i + \sum_s x_{(k-1)(j \in Night)_s}^i + \sum_s x_{k(j \in Night)_s}^i + \sum_j \sum_s x_{(k+1)js}^i + q_k^i \leq 4 \quad ; \forall i, k \in \{3, \dots, 7|LK| - 1\} \quad (23)$$

$$\sum_s x_{(k-2)(j \in Night)_s}^i + \sum_s x_{(k-1)(j \in Night)_s}^i + \sum_s x_{k(j \in Night)_s}^i + \sum_j \sum_s x_{(k+2)js}^i + q_k^i \leq 4 \quad ; \forall i, k \in \{3, \dots, 7|LK| - 2\} \quad (24)$$

### 2.3.2 Soft constraints

- Lower and upper bound on the total number of hours worked by each nurse during a week:

$$\sum_{k=7k_2-6}^{7k_2} \sum_j \sum_s h_{kj} x_{kjs}^i + d_{k_2}^{1i} \geq wh_{min} \quad ; \forall i, k_2 \in \{1, \dots, |LK|\} \quad (25)$$

$$\sum_{k=7k_2-6}^{7k_2} \sum_j \sum_s h_{kj} x_{kjs}^i - d_{k_2}^{2i} \leq wh_{max} \quad ; \forall i, k_2 \in \{1, \dots, |LK|\} \quad (26)$$

- Each nurse should not be assigned to work between two days off (off-on-off is not allowed):

$$\frac{\sum_j \sum_s x_{(k-1)js}^i}{\max(1, \sum_j \sum_s x_{(k-1)js}^i)} - \frac{\sum_j \sum_s x_{kjs}^i}{\max(1, \sum_j \sum_s x_{kjs}^i)} + \frac{\sum_j \sum_s x_{(k+1)js}^i}{\max(1, \sum_j \sum_s x_{(k+1)js}^i)} + d_k^{3i} \geq 0$$

;  $\forall i, k \in \{2, \dots, 7|LK| - 1\}$  (27)

- Each nurse is interested in taking a rest in pre-determined days/shifts:

$$\sum_s x_{kjs}^i - d_{kj}^{4i} = 0$$

;  $\forall i, k \in k_i, j \in j_{k_i}$  (28)

$$d_{k_2}^{1i} \geq 0$$

;  $\forall i, k_2 \in \{1, \dots, |LK|\}$  (29)

$$d_{k_2}^{2i} \geq 0$$

;  $\forall i, k_2 \in \{1, \dots, |LK|\}$  (30)

$$d_k^{3i} \geq 0$$

;  $\forall i, k \in \{2, \dots, 7|LK| - 1\}$  (31)

$$d_{kj}^{4i} \geq 0$$

;  $\forall i, k \in k_i, j \in j_{k_i}$  (32)

## 2.4 Simplification of the presented nurse scheduling model

Objective functions and soft constraints of the presented model can be rewritten as follows:

$$Z_1 = \text{Min} \sum_i \sum_{k_2=1}^{|LK|} [\max\{0, (wh_{\min} - \sum_{k=7k_2-6}^{7k_2} \sum_j \sum_s h_{kj} x_{kjs}^i)\} + \max\{0, (\sum_{k=7k_2-6}^{7k_2} \sum_j \sum_s h_{kj} x_{kjs}^i - wh_{\max})\}]$$

(33)

$$Z_2 = \text{Min} \sum_i \sum_{k=2}^{7|LK|-1} [\max\{0, (-\frac{\sum_j \sum_s x_{(k-1)js}^i}{\max(1, \sum_j \sum_s x_{(k-1)js}^i)} + \frac{\sum_j \sum_s x_{kjs}^i}{\max(1, \sum_j \sum_s x_{kjs}^i)} - \frac{\sum_j \sum_s x_{(k+1)js}^i}{\max(1, \sum_j \sum_s x_{(k+1)js}^i)}\}]]$$

(34)

$$Z_3 = \text{Min} \sum_i \sum_{k \in k_i} \sum_{j \in j_{k_i}} \sum_s x_{kjs}^i$$

(35)

$$Z_4 = \text{Min} \sum_i \sum_k \sum_j \sum_s [(s - RSL^i) * x_{kjs}^i * Penl]$$

(36)

which are subject to the constraints (5) to (24).

## 3. META-HEURISTIC SOLUTION METHODS

The DE technique, which belongs to the class of evolutionary algorithms, is established on the principles of mutation, crossover, and selection [47,48]. It is based on a special way of creating new individuals. Mutation and crossover operators are used to generate new individuals, and the selection operator chooses the best generated individual for the next generation. This algorithm, which has recently become increasingly popular, is very efficient. For instance, Pan et al. [40] presented a discrete DE algorithm to solve the permutation flow-shop scheduling problem with total flow-time criterion. Damak et al. [22] recommended a DE algorithm to solve the resource-constrained project scheduling problem with multiple execution modes for each activity and minimisation of the makespan. Wang et al. [53] presented a new hybrid discrete DE algorithm to solve the blocking flow-shop scheduling problems in order to minimise the maximum completion time. Pan et al. [41] also developed an effective hybrid discrete DE algorithm to minimise the maximum



completion time for a flow-shop scheduling problem, with intermediate buffers located between two consecutive machines.

In order to investigate the quality of generated solutions by the proposed DE in this paper, a GRASP algorithm is also applied to solve the given problem. GRASP is a meta-heuristic that is characterised by multiple initialisations [27,42]. In this algorithm, a feasible solution is initially obtained, and then is improved by a local search technique. This process is repeated in an iterative manner to reach the best solution. Some applications of the GRASP algorithm are observed in [4,7,8,38]. The proposed structures of both algorithms are thoroughly explained.

### 3.1 The proposed structure of the DE algorithm

The proposed structure of DE in pseudo-code format is as follows:

---

```

{
Procedure DE
Initialisation:
  Initialise initial population.
  Initialise Pareto Archive set so that it is empty.
  For counter=1 to max-iteration (e.g., 100)
    For each solution in the population
      Create a mutant matrix for each solution in the current population.
      Apply crossover operator and select elements of a solution.
      Apply repair method to construct a feasible solution.
      Apply feasibility check procedure for generated solutions in the previous step.
    End for
    Apply improvement procedure on the generated new solutions.
    Apply feasibility check procedure on the outputs of improvement procedure.
    Update Pareto Archive set.
  End for
Return Pareto Archive set.
End DE
}

```

---

The initialisation method, mutant operator, crossover operator, repair procedure, solution feasibility check procedure, solution improvement procedure, and Pareto Archive updating procedure propounded in the above pseudo-code are described in subsections 3.1.1 to 3.1.7 below.

#### 3.1.1 Initialisation method

The initial solutions generation method that can produce high quality and feasible initial population has been designed as follows:

---

```

{
Repeat the following procedure equal to the required number of initial solutions (e.g., 100 times):
For each Morning shift of planning period
  Assign nurses randomly from among the nurses who are able to work (from the feasibility viewpoint) and also prefer to work on the considered day until the demand for all skill levels is met.
End for
For each Afternoon and Night shift of planning period
  Assign nurses randomly from among the nurses who are able and prefer to work on the considered day, and also have been assigned to fewer shifts so far, until the demand for all skill levels is met.
  If no qualified nurse is found to assign
    Assign nurses randomly from among the nurses who are able to work but do not prefer to work on the considered day, and also have been assigned to fewer shifts so far, until the demand for all skill levels is met.
  End if
End for
End
}

```

---

### 3.1.2 Mutant operator

In a similar way to the genetic algorithm (GA), the parent and offspring theory is used to search in the solution space. In this paper, a mutant matrix is produced for each solution as follows [53]:

---

```
{
For each solution  $s_i$  in the population
  Generate two indices  $j$  and  $k$  that are non-equal to  $i$  randomly.
  Consider two solutions  $s_j$  and  $s_k$  and obtain a new solution (a mutant matrix) by using solutions  $s_j$  and  $s_k$  according to Equation (37):
    
$$M_i = s_i + A * rand * (s_j - s_k) \tag{37}$$

  End For
}
```

---

where  $A$ , which is used to control the population's evolution rate, is greater than 1 (e.g., 2) and  $rand$  is a random number in the interval  $[0,1]$ . Note that each generated mutant matrix  $M_i$  corresponds to a solution  $s_i$ .

### 3.1.3 Crossover operator

Once the mutant matrix is produced in the DE algorithm, the crossover operator must be specified for each solution structure too. In this paper, the crossover operator is defined by Equation (38):

$$T_i = \begin{cases} M_i & r_i < cr \\ s_i & otherwise \end{cases} \tag{38}$$

where  $cr$  is a random number in the interval  $[0,1]$ , and  $r_i$  is a matrix whose elements are generated randomly in the interval  $[0,1]$  for each  $s_i$  and  $M_i$ . Obviously  $r_i$  has the same size as  $s_i$ . It should be noted that Equation (38) is applied to each solution  $s_i$  and to its corresponding mutant matrix  $M_i$  in order to obtain a matrix  $T_i$ .

### 3.1.4 Repair procedure

There is a matrix  $T_i$  for each solution  $s_i$  and its corresponding  $M_i$  that is used to convert solution  $s_i$  to a feasible one by using a repair procedure. Each element of  $T_i$  includes a number that has been generated by applying mutant and crossover operators. These elements are used to prioritise the nurses who will be assigned to shifts. The repair procedure is applied to all solutions to produce new feasible ones where possible.

---

```
{
For each solution  $s_i$  in the population
  For each required skill level in each shift of planning period.
    Assign a nurse who is able to work in the considered shift and whose corresponding element in matrix  $T_i$  is smaller than the other qualified nurses.
    If no qualified nurse is found to assign to the considered skill level
      Do not assign any nurse.
    End if
  End for
  If some required skill levels for some shifts of solution  $s_i$  have not been met
    Consider  $s_i$  as an infeasible solution and apply feasibility check procedure on it in order to convert it to a feasible one if possible.
  End if
End for
}
```

---

### 3.1.5 Solution feasibility check procedure

Since new solutions are generated while running the proposed DE algorithm, a procedure has been designed to check the satisfaction of all constraints for the generated solution, which tries to convert an infeasible solution to a feasible one if possible. This procedure is expressed as the following:

---

```
{
For each generated solution  $s_i$ 
  If a nurse has been assigned to work for more than 12 hours consecutively
    Cancel some of his/her assignments so that the violated constraint is satisfied, and then assign
    a qualified nurse instead.
  End if
  If a nurse has been assigned to work after working the maximum number of successive night shifts
    Cancel some of his/her assignments so that the violated constraint is satisfied, and then assign
    a qualified nurse instead.
  End if
  If a nurse has been assigned to work at a higher skill level than his/her real skill in a shift
    If the number of assignments at the considered skill level in this shift is more than the required
    number
      Cancel his/her assignment.
    End if
    If the number of assignments at the considered skill level in this shift is less than or equal to
    the required number
      Regarding the satisfaction of all constraints, cancel his/her assignment and assign a nurse
      from among the nurses who are able to work at the considered skill level and who have not
      been assigned to this shift so far, or displace his/her assignment with a nurse who is able to
      work at the considered skill level but has been assigned to work at a lower skill level than
      the considered skill in the same shift if this displacement is possible.
      If both these changes are impossible
        Cancel his/her assignment, and change the assignment of a nurse who can work at the
        considered skill level but has been scheduled to work in the previous or next shifts if
        possible, in order to assign at the considered skill level.
      End if
    End if
  End if
  If the demand for a skill level in a shift has not been regarded
    If the number of assignments is more than the required number
      Cancel extra assignments.
    End if
    If the number of assignments is less than the required number
      Regarding the satisfaction of all constraints, assign some qualified nurses.
      If no qualified nurse is found to assign
        Change the schedule of previous or next shifts/days if possible until the required
        number is met.
      End if
    End if
  End if
End for
}
```

---

For the rest constraints, nurses are assigned in this manner, and the schedules change as much as possible until all constraints are satisfied. If assigning a nurse while taking all conditions into consideration is not possible, this solution cannot be converted into a feasible one, and the procedure will end.

### 3.1.6 Solution improvement procedure

This procedure improves all generated feasible solutions, as far as possible. Three neighborhood search operators (NSO1, NSO2, and NSO3) are used in parallel to improve the feasible solutions as below:

---

```

{
For each solution  $s_i$  in the population
  Counter = 0
  While Counter  $\leq$  max-Iteration (say, 20 in this paper) do
    Apply NSO1 on the solution  $s_i$ .
    Apply NSO2 on the solution  $s_i$ .
    Apply NSO3 on the solution  $s_i$ .
    Select the best solution from among the current solution  $s_i$  and the three solutions generated
    by NSO1, NSO2, and NSO3 by using non-dominated relations, and consider it as the current
    solution for the next iteration.
    Add one unit to Counter
  End while
End For
}

```

---

The applied NSOs are defined as follows:

- NSO1: A nurse is selected randomly. If he/she has been assigned to work some shifts/days that he/she prefers to rest, his/her assignment will be cancelled for that shift/day and another nurse who is able to work this shift/day will replace him/her.
- NSO2: A nurse is selected randomly and his/her assignments are checked for all shifts. If this nurse has been assigned to work at a lower skill level than his/her real skill in a shift, his/her assignment will be cancelled for this shift and a qualified nurse in the same shift will replace him/her if possible; or his/her assignment will be displaced with a nurse who has been assigned to work in the same shift at a skill level closer to the considered skill level.
- NSO3: The assignment of nurses who have been assigned to work between two off-days is cancelled, and is replaced by some qualified nurses, if possible.

It should be mentioned that the improvement procedure is even applied to the infeasible solutions obtained by the NSOs in order to convert them to feasible ones. If this conversion is impossible, the solution  $s_i$  will be considered as the output of the improvement procedure. The population acquired by the improvement procedure is regarded as the population of the next iteration, after checking the feasibility.

### 3.1.7 Pareto Archive updating procedure

In any iteration, non-dominated solutions are stored in a set called the Pareto Archive. The best solution obtained in each iteration is compared with the solutions existing in the Pareto Archive set by using non-dominated relations. The selected non-dominated solutions are then considered as a new Pareto Archive.

## 3.2 The proposed structure of the GRASP algorithm

The general structure of the GRASP algorithm is as follows:

---

```

{
Procedure GRASP (Max-Iteration, Seed)
Read Input();
For k = 1 to Max-Iteration
  Solution  $\leftarrow$  Greedy Randomised Construction.
  If Solution is not feasible then
    Solution  $\leftarrow$  Repair(Solution).
  End if
  Solution  $\leftarrow$  Local Search(Solution).
  Update Solution(Solution,Best solution).
End for
Return Best Solution.
End GRASP
}

```

---

In the proposed GRASP algorithm, non-dominated solutions in any iteration are stored in a set, and eventually a set of solutions is considered as the output of the GRASP algorithm. The proposed structure of GRASP in pseudo-code format is as follows:

---

```

{
Procedure GRASP (Max-Iteration)
Read Input();
For k = 1 to Max-Iteration (e.g., 500)
    Solution ← Greedy Randomised Construction.
    If Solution is not feasible then
        Solution ← Repair(Solution).
    End if
    Solution ← Local Search(Solution).
    Update Solution(Solution,Best solution).
    Update Pareto Archive.
End for
Return Pareto Archive.
End GRASP
}

```

---

### 3.2.1 Greedy randomised construction procedure

In this phase, one solution is initially constructed. Solution construction starts from the first shift of the first day and continues to the last shift of the last day of the planning horizon.

---

```

{
For each required skill level in each shift of planning period
    Assign some qualified nurses randomly until the required number is met.
    If the required number is not met
        Apply repair procedure to meet the required number.
    End if
End For
}

```

---

### 3.2.2 Repair procedure

The solution feasibility check procedure proposed in Section 3.1.5 is used to repair the solutions.

### 3.2.3 Local search procedure

The three neighborhood search operators (NSO1, NSO2, and NSO3) proposed in Section 3.1.6 are used in parallel as a local search procedure.

### 3.2.4 Solution updating procedure

In this procedure, the best solution is selected from among the current solution and the three solutions obtained by the NSOs in the local search procedure using non-dominated relations. If more than one solution is found to be the best, one will be randomly selected.

### 3.2.5 Pareto Archive updating procedure

The updating procedure proposed in Section 3.1.7 is used to update the Pareto Archive.

## 4. COMPUTATIONAL RESULTS

To examine the efficiency of the model presented in Section 2 (formulated according to a real case study carried out in the maternity ward of Sina Hospital in Isfahan), two numerical examples are solved by the proposed DE and GRASP algorithms on an Intel(R) Core(TM) i3 with 4 GB of memory and 2.27 GHZ of Dual Core CPU, and MATLAB R2009a software. The parameters of the given model are presented in Table 1. The planning horizon is considered to be 35 days (5 weeks).

Table 1: Common features of problem instances

$dh_{min}$	$dh_{max}$	$wh_{min}$	$wh_{max}$	$fh_{min}$	$fh_{max}$
0	18	24	90	0	90
$mh_{min}$	$mh_{max}$	<i>Max – Night</i>		<i>Penl(\$)</i>	
128	252	20		10	

Table 2 lists the real skill level of all nurses for both problem instances. For example, the real skill level of nurses with ID=1 to 8 in Problem instance 1 is ‘nurse’, and they are able to work at ‘auxiliary nurse’ and ‘co-auxiliary nurse’ skill levels as well as their real skill; and the real skill level of nurses with ID=21 to 35 in Problem instance 2 is ‘auxiliary nurse’, and they are able to work at ‘co-auxiliary nurse’ skill level as well as their real skill. As mentioned before, all nurses with a higher real skill level are capable of working at lower skill levels. Nurses with ‘co-auxiliary nurse’ real skill level can only work at their real skill level.

Table 2: Real skill level of nurses

<i>Real skill level</i>	<i>Nurses’ ID (Problem instance 1)</i>	<i>Nurses’ ID (Problem instance 2)</i>
Nurse	1-8	1-20
Auxiliary nurse	9-14	21-35
Co-auxiliary nurse	15-20	36-50

Table 3 represents the number of nurses required in each shift of the planning period. For example, two nurses at ‘auxiliary nurse’ skill level are needed in ‘Morning’, ‘Afternoon’, and ‘Night’ shifts of all days in Problem instance 1 (remembering that all nurses with ‘Nurse’ and ‘Auxiliary nurse’ real skill levels can work at this skill level).

Table 3: Number of nurses required in each shift of the planning period

<i>Number of required nurses</i>	<i>Nurse</i>	<i>Auxiliary nurse</i>	<i>Co-auxiliary nurse</i>
<i>Problem instance 1</i>	Morning: 2	Morning: 2	Morning: 2
	Afternoon: 2	Afternoon: 2	Afternoon: 2
	Night: 2	Night: 2	Night: 0
<i>Problem instance 2</i>	Morning: 6	Morning: 4	Morning: 4
	Afternoon: 6	Afternoon: 4	Afternoon: 4
	Night: 4	Night: 4	Night: 4

Tables 4 and 5 summarise the rest days that nurses prefer in both problem instances. These days are determined by nurses at the beginning of planning.

Table 4: Preferred rest days (Problem instance 1)

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

Table 5: Preferred rest days (Problem instance 2)

<i>Nurses' ID</i>	<i>Preferred days</i>	<i>Nurses' ID</i>	<i>Preferred days</i>
1	17,18,24,25	26	5,6,12,13
2	1,2,4,23	27	14,17,18,19
3	5,6,19,28,29,30	28	26,27,28
4	3,21,22,23	29	4,5,32,33
5	1,2,14,17,25,27,28	30	23,24,27,28
6	9,10,11,34,35	31	1,7,8
7	12,13,14,15	32	2,3,6,7,8
8	25,26	33	17,18,19
9	8,12,13,14	34	19,21,22
10	18,19,20	35	10,11,13,14
11	7,8,28,29,30	36	3,4,34,35
12	9,13,15,19,20	37	10,25,26
13	1,3,31,32,33	38	17,18,19
14	5,6,21,22	39	2,3,4,5
15	4,5,6,11,12	40	1,8,9,10
16	18,26	41	21,22,23
17	16,17,18	42	26,27,28
18	25,26	43	25,26,27
19	23,27,28	44	9,10
20	2,10,23,24,25	45	15,16,17
21	23,24,25,26	46	4,5,21,22
22	7,22,23	47	1,2,3,4,35
23	1,2,3,15,16,17,18	48	23,28,29
24	4,20	49	13,14,17,18
25	9,10,11,12	50	3,4,5,34

Tables 6 and 7 illustrate the objective values of non-dominated solutions existing in Pareto Archives of the proposed DE and GRASP algorithms for Problem instances 1 and 2 respectively. Note that non-repetitive non-dominated solutions (non-identical non-dominated solutions) existing in a Pareto Archive set may have equal objective values. For example, non-repetitive non-dominated solutions 2 and 14 obtained by the proposed DE for Problem instance 1 have equal objective values, as shown in Table 6.

Table 6: Objective values of Pareto Archived set by the proposed DE and GRASP (Problem instance 1)

DE													
<i>Solution number</i>													
	1	2	3	4	5	6	7	8	9	10	11	12	13
$Z_1$	0	12	12	12	12	12	12	12	12	12	12	12	12
$Z_2$	40	18	13	7	7	13	7	13	15	7	13	13	12
$Z_3$	46	48	49	51	51	51	51	51	51	52	49	49	50
$Z_4$	350	400	360	380	380	330	380	330	310	340	360	360	380
<i>Solution number</i>													
	14	15	16	17	18	19	20	21	22	23	24	25	
$Z_1$	12	12	12	12	12	12	12	12	12	12	12	12	
$Z_2$	18	12	5	12	8	12	8	12	8	12	5	24	
$Z_3$	48	50	51	50	49	50	49	50	49	50	50	50	
$Z_4$	400	380	420	380	440	380	440	380	440	380	450	350	
GRASP													
<i>Solution number</i>													
	1	2	3	4	5								
$Z_1$	12	12	12	0	12								
$Z_2$	41	42	43	41	42								
$Z_3$	48	47	48	49	47								
$Z_4$	510	510	500	380	510								

Table 7: Objective values of Pareto Archived set by the proposed DE and GRASP  
(Problem instance 2)

DE												
Solution number												
	1	2	3	4	5	6	7	8	9	10	11	12
Z <sub>1</sub>	78	60	12	12	66	60	90	84	0	24	24	60
Z <sub>2</sub>	13	27	118	116	21	29	5	13	124	117	114	29
Z <sub>3</sub>	76	75	73	58	65	68	77	75	73	56	58	68
Z <sub>4</sub>	630	630	250	280	590	590	690	690	140	480	480	590
Solution number												
	13	14	15	16	17	18	19	20	21	22	23	
Z <sub>1</sub>	216	204	228	216	222	210	24	60	204	96	12	
Z <sub>2</sub>	27	28	20	21	25	28	117	29	28	9	116	
Z <sub>3</sub>	59	60	61	61	60	62	56	68	60	80	58	
Z <sub>4</sub>	1050	1030	1000	990	990	970	480	590	1030	670	280	
GRASP												
Solution number												
	1	2	3	4	5	6	7	8	9	10	11	12
Z <sub>1</sub>	0	0	0	24	12	30	0	6	0	12	0	6
Z <sub>2</sub>	127	126	127	121	123	121	126	122	122	123	127	123
Z <sub>3</sub>	73	75	75	76	74	75	79	77	76	74	74	74
Z <sub>4</sub>	340	350	130	270	340	330	310	310	330	340	330	390

Table 8 lists all possible assignments that may be used in the shift schedule tables. Due to insufficient space in this paper, all assignments are displayed as an abbreviation.

Table 8: Abbreviations of all possible assignments used in the shift schedule tables

<i>All possible assignments in terms of Day:Shift:Skill</i>	
<i>k:1:1</i>	Nurse with ID= <i>i</i> has been assigned to work the morning shift of day <i>k</i> at 'Nurse' skill level.
<i>k:1:2</i>	Nurse with ID= <i>i</i> has been assigned to work the morning shift of day <i>k</i> at 'Auxiliary nurse' skill level.
<i>k:1:3</i>	Nurse with ID= <i>i</i> has been assigned to work the morning shift of day <i>k</i> at 'Co-auxiliary nurse' skill level.
<i>k:2:1</i>	Nurse with ID= <i>i</i> has been assigned to work the afternoon shift of day <i>k</i> at 'Nurse' skill level.
<i>k:2:2</i>	Nurse with ID= <i>i</i> has been assigned to work the afternoon shift of day <i>k</i> at 'Auxiliary nurse' skill level.
<i>k:2:3</i>	Nurse with ID= <i>i</i> has been assigned to work the afternoon shift of day <i>k</i> at 'Co-auxiliary nurse' skill level.
<i>k:3:1</i>	Nurse with ID= <i>i</i> has been assigned to work the night shift of day <i>k</i> at 'Nurse' skill level.
<i>k:3:2</i>	Nurse with ID= <i>i</i> has been assigned to work the night shift of day <i>k</i> at 'Auxiliary nurse' skill level.
<i>k:3:3</i>	Nurse with ID= <i>i</i> has been assigned to work the night shift of day <i>k</i> at 'Co-auxiliary nurse' skill level.

The validation of the presented model is proved by solving a small-sized problem (Problem instance 1) and comparing the results with a shift schedule provided manually by the maternity ward supervisor. Tables 9 to 11 represent a comparison between a manual shift schedule and two sample shift schedules obtained by the proposed algorithms (e.g., Solution 1 obtained by the DE, and Solution 4 obtained by the GRASP in Problem instance 1). This comparison confirms the performance of the constraints considered in the model.



Table 9: Shift schedule provided manually and two sample shift schedules obtained by the proposed DE and GRASP (Problem instance 1: Solution no. 1 of DE and Solution no. 4 of GRASP) for Nurses (ID=1 to 8)

ID	Type	Day:Shift:Skill
1	Manual	1:1:3-1:2:1-2:1:2-2:2:3-3:1:1-3:3:2-6:3:1-7:2:2-8:2:1-9:1:2-10:1:1-10:3:2-14:2:2-15:2:1-16:1:2-17:1:1-17:3:2-21:2:2-22:2:1-23:1:2-24:1:1-24:3:2-28:2:2-29:2:1-31:1:1-31:3:2-33:3:1-34:3:2-35:3:1
	DE	1:1:3-1:2:1-2:1:2-2:2:1-3:1:1-3:3:2-5:3:1-7:2:2-8:2:1-9:1:2-10:1:1-10:3:2-12:3:1-14:2:2-15:2:1-16:1:2-17:1:1-17:3:2-19:3:1-21:2:2-22:2:1-23:1:2-24:1:1-24:3:2-26:3:1-28:2:2-29:2:1-30:1:2-31:1:1-32:1:1-32:3:1-34:3:1
	GRASP	1:2:1-2:2:1-3:1:2-4:1:1-4:3:2-6:3:1-8:2:2-9:2:1-10:1:2-11:1:1-11:3:2-13:3:1-15:2:2-16:2:1-17:1:2-18:1:1-18:3:2-20:3:1-22:2:2-23:2:1-24:1:2-25:1:1-25:3:2-27:3:1-29:2:2-30:2:1-31:1:2-32:1:1-32:3:2-34:3:1
2	Manual	1:1:3-1:2:3-2:1:2-2:2:3-3:1:1-5:3:1-6:3:1-7:2:2-8:2:1-9:1:2-10:1:1-10:3:2-12:3:1-13:3:1-14:2:2-15:2:1-16:1:2-17:1:1-17:3:2-20:3:1-21:2:2-22:2:1-23:1:2-24:1:1-24:3:2-27:3:1-28:2:2-29:2:1-31:3:2-34:3:1-35:2:2
	DE	1:1:2-1:2:1-2:1:2-3:1:1-3:3:2-5:3:1-8:2:1-9:1:2-10:1:1-10:3:2-12:3:1-15:2:1-16:1:2-17:1:1-17:3:2-19:3:1-22:2:1-23:1:2-24:1:1-24:3:2-26:3:1-29:2:1-30:1:2-31:1:1-31:3:2-33:3:1-35:2:2
	GRASP	1:1:3-2:1:2-3:1:1-3:3:2-5:3:1-7:2:2-8:2:2-9:2:1-10:1:2-11:1:1-11:3:2-13:3:1-15:2:2-16:2:1-17:1:2-18:1:1-18:3:2-20:3:1-22:2:2-23:2:1-24:1:2-25:1:1-25:3:2-27:3:1-29:2:2-30:2:1-31:1:2-32:1:1-32:3:2-34:3:1
3	Manual	1:1:2-2:1:1-2:3:1-3:3:1-4:2:1-5:2:1-6:2:1-7:2:1-8:1:1-9:1:1-9:3:1-11:2:1-12:2:1-13:2:1-14:2:1-15:1:1-16:1:1-16:3:1-18:2:1-19:2:1-20:2:1-21:2:1-22:1:1-23:1:1-23:3:1-25:2:1-26:2:1-27:2:1-28:2:1-29:1:1-30:1:1-30:3:1-32:2:1-33:2:1-34:2:1-35:2:1
	DE	1:1:2-1:2:3-2:3:1-4:2:1-5:2:1-6:2:1-7:2:1-8:1:1-9:1:1-9:3:1-11:2:1-12:2:1-13:2:1-14:2:1-15:1:1-16:1:1-16:3:1-18:2:1-19:2:1-20:2:1-21:2:1-22:1:1-23:1:1-23:3:1-25:2:1-26:2:1-27:2:1-28:2:1-29:1:1-30:1:1-30:3:1-32:3:1-34:3:1-35:2:2
	GRASP	1:1:3-2:1:2-3:1:1-3:3:1-5:2:1-6:3:1-8:2:1-9:1:1-10:1:1-10:3:1-12:2:1-13:2:1-14:2:1-15:2:1-16:1:1-17:1:1-17:3:1-19:2:1-20:2:1-21:2:1-22:2:1-23:1:1-24:1:1-24:3:1-26:2:1-27:2:1-28:2:1-29:2:1-30:1:1-31:1:1-31:3:1-33:2:1-34:2:1-35:2:1
4	Manual	1:1:2-2:1:1-2:3:1-3:3:1-4:2:1-5:2:1-6:2:1-7:2:1-8:1:1-9:1:1-9:3:1-11:2:1-12:2:1-13:2:1-14:2:1-15:1:1-16:1:1-16:3:1-18:2:1-19:2:1-20:2:1-21:2:1-22:1:1-23:1:1-23:3:1-25:2:1-26:2:1-27:2:1-28:2:1-29:1:1-30:1:1-30:3:1-32:2:1-33:2:1-34:2:1-35:2:1
	DE	1:1:3-1:2:3-2:1:1-2:3:1-4:2:1-5:2:1-6:2:1-7:2:1-8:1:1-9:1:1-9:3:1-11:2:1-12:2:1-13:2:1-14:2:1-15:1:1-16:1:1-16:3:1-18:2:1-19:2:1-20:2:1-21:2:1-22:1:1-23:1:1-23:3:1-25:2:1-26:2:1-27:2:1-28:2:1-29:1:1-30:1:1-30:3:1-32:2:1-33:3:1
	GRASP	1:1:2-2:1:1-2:3:1-4:3:1-6:2:1-7:2:1-8:2:1-9:1:1-10:1:1-10:3:1-12:2:1-13:2:1-14:2:1-15:2:1-16:1:1-17:1:1-17:3:1-19:2:1-20:2:1-21:2:1-22:2:1-23:1:1-24:1:1-24:3:1-26:2:1-27:2:1-28:2:1-29:2:1-30:1:1-31:1:1-31:3:1-33:2:1-34:2:1-35:2:1
5	Manual	1:3:1-5:1:1-6:1:1-7:1:1-7:3:1-8:3:1-9:2:1-10:3:1-12:1:1-13:1:1-14:1:1-14:3:1-15:3:1-16:2:1-17:3:1-19:1:1-19:3:1-20:1:1-21:1:1-21:3:1-23:2:1-24:3:1-26:1:1-27:1:1-28:1:1-28:3:1-29:3:1-31:3:1-33:1:1-34:1:1-35:1:1
	DE	2:1:1-2:2:1-3:3:1-5:1:1-6:1:1-7:1:1-7:3:1-9:2:1-10:3:1-12:1:1-13:1:1-14:1:1-14:3:1-16:2:1-17:3:1-19:1:1-20:1:1-21:1:1-21:3:1-23:2:1-24:3:1-26:1:1-27:1:1-28:1:1-28:3:1-30:2:1-31:3:1-33:2:1-34:2:1-35:2:1
	GRASP	1:1:2-2:1:1-2:3:1-4:3:1-6:2:1-7:2:1-8:1:1-8:3:1-10:2:1-11:3:1-13:1:1-14:1:1-14:3:1-15:1:1-15:3:1-17:2:1-18:3:1-20:1:1-21:1:1-22:1:1-22:3:1-24:2:1-25:3:1-27:1:1-28:1:1-29:1:1-29:3:1-31:2:1-32:3:1-34:1:1-35:1:1
6	Manual	1:1:1-1:3:1-5:1:1-6:1:1-7:1:1-7:3:1-9:2:1-10:3:1-12:1:1-13:1:1-14:1:1-14:3:1-16:2:1-17:3:1-19:1:1-20:1:1-21:1:1-21:3:1-23:2:1-24:3:1-26:1:1-27:1:1-28:1:1-28:3:1-30:2:1-31:3:1-33:2:1-34:2:1-35:2:1
	DE	3:3:1-5:1:1-6:1:1-7:1:1-7:3:1-9:2:1-10:3:1-12:1:1-13:1:1-14:1:1-14:3:1-16:2:1-17:3:1-19:1:1-20:1:1-21:1:1-21:3:1-23:2:1-24:3:1-26:1:1-27:1:1-28:1:1-28:3:1-30:2:1-31:3:1-33:2:1-34:2:1-35:2:1
	GRASP	1:3:1-3:3:1-4:2:1-5:2:1-6:1:1-7:1:1-8:1:1-8:3:1-10:2:1-11:3:1-13:1:1-14:1:1-14:3:1-15:1:1-15:3:1-17:2:1-18:3:1-20:1:1-21:1:1-22:1:1-22:3:1-24:2:1-25:3:1-27:1:1-28:1:1-29:1:1-29:3:1-31:2:1-32:3:1-34:1:1-35:1:1
7	Manual	1:1:1-1:2:1-2:2:1-3:2:1-4:1:1-4:3:1-8:3:1-10:2:1-11:1:1-11:3:1-15:3:1-17:2:1-18:1:1-18:3:1-19:3:1-22:3:1-24:2:1-25:1:1-25:3:1-26:3:1-29:3:1-31:1:1-31:3:1-32:1:1-32:3:1-34:1:1-35:1:1
	DE	1:1:1-1:3:1-3:2:1-4:1:1-4:3:1-6:3:1-8:3:1-10:2:1-11:1:1-11:3:1-13:3:1-15:3:1-17:2:1-18:1:1-18:3:1-20:3:1-22:3:1-24:2:1-25:1:1-25:3:1-27:3:1-29:3:1-31:2:1-32:2:1-33:1:1-34:1:1-35:1:1-35:3:1
	GRASP	1:1:1-1:3:1-3:2:1-4:2:1-5:1:1-6:1:1-7:1:1-7:3:1-9:3:1-11:2:1-12:1:1-12:3:1-14:3:1-16:3:1-18:2:1-19:1:1-19:3:1-21:3:1-23:3:1-25:2:1-26:1:1-26:3:1-28:3:1-30:3:1-32:2:1-33:1:1-33:3:1-35:3:1
8	Manual	1:1:1-1:2:3-1-4:1:1-4:3:1-5:3:1-6:3:1-7:3:1-9:3:1-11:2:1-12:1:1-12:3:1-13:3:1-15:3:1-17:2:1-18:1:1-18:3:1-19:3:1-20:3:1-24:2:1-25:1:1-25:3:1-26:3:1-27:3:1-30:2:1-31:2:1-32:1:1-32:3:1-33:3:1-34:3:1
	DE	1:1:1-1:3:1-3:2:1-4:1:1-4:3:1-6:3:1-8:3:1-10:2:1-11:1:1-11:3:1-13:3:1-15:3:1-17:2:1-18:1:1-18:3:1-20:3:1-22:3:1-24:2:1-25:1:1-25:3:1-27:3:1-29:3:1-31:2:1-32:1:1-33:1:1-34:1:1-35:1:1-35:3:1
	GRASP	1:1:1-1:2:1-2:2:1-3:2:1-4:1:1-5:1:1-5:3:1-7:3:1-9:3:1-11:2:1-12:1:1-12:3:1-14:3:1-16:3:1-18:2:1-19:1:1-19:3:1-21:3:1-23:3:1-25:2:1-26:1:1-26:3:1-28:3:1-30:3:1-32:2:1-33:1:1-33:3:1-35:3:1

It takes more than four days for the ward supervisor to provide a shift schedule manually in the small-sized problem (Problem instance 1). When the number of nurses is greater than 25, and the nurses' preferences increase, the ward supervisor is not capable of providing a shift schedule manually. Since Sina hospital intends to develop the maternity ward and employ more nurses, the hospital managers have decided to automate the nurses' shift scheduling.

Table 10: Shift schedule provided manually and two sample shift schedules obtained by the proposed DE and GRASP (Problem instance 1: Solution no. 1 of DE and Solution no. 4 of GRASP) for Auxiliary nurses (ID=9 to 14)

ID	Type	Day:Shift:Skill
9	Manual	1:3:2-2:3:2-3:2-4:3:2-6:2-7:1:2-8:1:2-8:3:2-10:2-11:3:2-13:2-14:1:2-15:1:2-15:3:2-17:2-2-18:3:2-20:2-2-21:1:2-22:1:2-22:3:2-24:2:2-25:3:2-27:2:2-28:1:2-29:1:2-29:3:2-31:2-2-32:3:2-1:3:2-3:2-4:3:2-6:2-7:1:2-7:2-8:1:2-8:3:2-10:2-11:3:2-13:2-14:1:2-14:2:2-15:1:2-15:3:2-17:2-2-18:3:2-20:2-2-21:1:2-21:2-2-22:1:2-22:3:2-24:2:2-25:3:2-27:2:2-28:1:2-29:1:2-29:3:2-31:2-2-32:3:2-34:2:2-35:1:2
	DE	1:3:2-3:2-5:3:2-7:2-8:1:2-9:1:2-9:3:2-11:2-12:2-12:3:2-14:2-15:1:2-16:1:2-16:3:2-18:2-2-19:3:2-21:2-2-22:1:2-23:1:2-23:3:2-25:2:2-26:3:2-28:2:2-29:1:2-30:1:2-30:3:2-32:2:2-33:3:2-35:2:2
	GRASP	1:3:2-3:2-4:3:2-6:2-7:1:2-8:1:2-8:3:2-10:2-11:3:2-13:2-14:1:2-15:1:2-15:3:2-17:2-2-18:3:2-20:2-2-21:1:2-22:1:2-22:3:2-24:2:2-25:3:2-27:2:2-28:1:2-29:1:2-29:3:2-31:2-2-32:3:2-33:3:2-34:2:2-35:2:2
10	Manual	1:3:2-2:3:2-4:3:2-6:2-7:1:2-8:1:2-8:3:2-10:2-11:3:2-13:2-14:1:2-15:1:2-15:3:2-17:2-2-18:3:2-20:2-2-21:1:2-22:1:2-22:3:2-24:2:2-25:3:2-27:2:2-28:1:2-29:1:2-29:3:2-31:2-2-32:3:2-33:3:2-34:2:2-35:2:2
	DE	1:3:2-2:3:2-4:3:2-6:2-7:1:2-8:1:2-8:3:2-10:2-11:3:2-13:2-14:1:2-15:1:2-15:3:2-17:2-2-18:3:2-20:2-2-21:1:2-22:1:2-22:3:2-24:2:2-25:3:2-27:2:2-28:1:2-28:2:2-29:1:2-29:3:2-31:2-2-32:3:2-34:2:2-35:1:2
	GRASP	1:3:2-3:2-4:3:2-6:2-7:1:2-8:1:2-9:1:2-9:3:2-11:2-12:2-12:3:2-14:2-15:1:2-16:1:2-16:3:2-18:2-2-19:3:2-21:2-2-22:1:2-23:1:2-23:3:2-25:2:2-26:3:2-28:2:2-29:1:2-30:1:2-30:3:2-32:2:2-33:3:2-35:2:2
11	Manual	2:3:2-3:2-4:2-5:2-6:1:2-6:3:2-7:3:2-8:2-9:3:2-11:2-12:2-12:3:2-13:1:2-13:3:2-15:2-16:3:2-18:2-2-19:2-2-20:1:2-20:3:2-22:2-2-23:3:2-25:2:2-26:2:2-27:1:2-27:3:2-29:2:2-30:3:2-32:2:2-33:2:2-34:2:2-35:1:2
	DE	3:2-2-4:2-5:2-6:1:2-6:3:2-8:2-9:3:2-11:2-12:2-12:3:2-13:1:2-13:3:2-15:2-16:3:2-18:2-2-19:2-2-20:1:2-20:3:2-22:2-2-23:3:2-25:2:2-26:2:2-27:1:2-27:3:2-29:2:2-30:3:2-31:3:2-32:2:2-33:2-2-34:1:2-34:3:2
	GRASP	1:2:3-2:3:2-4:2-5:2-6:2-7:1:2-7:3:2-9:2-10:3:2-12:2-12:3:2-14:1:2-14:3:2-16:2-2-17:3:2-19:2-2-20:2-2-21:1:2-21:3:2-23:2-2-24:3:2-26:2:2-27:2:2-28:1:2-28:3:2-30:2-2-31:3:2-33:2:2-34:2:2-35:1:2-35:3:2
12	Manual	4:2-5:2-6:1:2-6:3:2-7:3:2-8:2-9:3:2-11:2-12:2-12:3:2-13:1:2-13:3:2-14:3:2-15:2-16:3:2-18:2-2-19:2-2-20:1:2-20:3:2-22:2-2-23:3:2-25:2:2-26:2:2-27:1:2-27:3:2-29:2:2-30:3:2-32:2:2-33:2:2-34:1:2-35:1:2-35:3:2
	DE	2:3-2-4:2-5:2-6:1:2-6:3:2-8:2-9:3:2-11:2-12:2-12:3:2-13:1:2-13:3:2-15:2-16:3:2-18:2-2-19:2-2-20:1:2-20:3:2-22:2-2-23:3:2-25:2:2-26:2:2-27:1:2-27:3:2-29:2:2-30:3:2-32:2:2-33:2:2-34:1:2-34:3:2
	GRASP	1:2:3-2:3:2-4:2-5:2-6:1:2-6:3:2-8:3:2-10:3:2-12:2-12:3:2-14:1:2-14:3:2-16:2-2-17:3:2-19:2-2-20:2-2-21:1:2-21:3:2-23:2-2-24:3:2-26:2:2-27:2:2-28:1:2-28:3:2-30:2-2-31:3:2-33:2:2-34:2:2-35:1:2-35:3:2
13	Manual	1:2-2-2:2-3:1:2-4:1:2-5:1:2-5:3:2-9:2-10:1:2-11:1:2-12:1:2-12:3:2-14:3:2-16:2-2-17:1:2-18:1:2-19:1:2-19:3:2-21:3:2-23:2-2-24:1:2-25:1:2-26:1:2-26:3:2-28:3:2-30:1:2-30:2:2-31:1:2-32:1:2-33:1:2-33:3:2-35:3:2
	DE	1:2:2-2:2-3:1:2-4:1:2-5:1:2-5:3:2-9:2-10:1:2-11:1:2-12:1:2-12:3:2-14:3:2-16:2-2-17:1:2-18:1:2-19:1:2-19:3:2-21:3:2-23:2-2-24:1:2-25:1:2-26:1:2-26:3:2-28:3:2-30:2-2-31:1:2-32:1:2-33:1:2-33:3:2-35:3:2
	GRASP	1:2:2-2:2-3:2-4:1:2-5:1:2-6:1:2-6:3:2-8:3:2-10:2-11:1:2-12:1:2-13:1:2-13:3:2-15:3:2-17:2-2-18:1:2-19:1:2-20:1:2-20:3:2-22:3:2-24:2:2-25:1:2-26:1:2-27:1:2-27:3:2-29:3:2-31:2:2-32:1:2-33:1:2-34:1:2-34:3:2
14	Manual	1:2-2-2:2-3:1:2-4:1:2-5:1:2-5:3:2-9:2-10:1:2-11:1:2-12:1:2-12:3:2-16:2-2-17:1:2-18:1:2-19:1:2-19:3:2-21:3:2-23:2-2-24:1:2-25:1:2-26:1:2-26:3:2-28:3:2-30:1:2-30:2:2-31:1:2-32:1:2-33:1:2-33:3:2-35:3:2
	DE	1:2-2-2:2-3:1:2-4:1:2-5:1:2-5:3:2-9:2-10:1:2-11:1:2-12:1:2-12:3:2-14:3:2-16:2-2-17:1:2-18:1:2-19:1:2-19:3:2-21:3:2-23:2-2-24:1:2-25:1:2-26:1:2-26:3:2-28:3:2-30:2-2-31:1:2-32:1:2-33:1:2-33:3:2-35:3:2
	GRASP	1:2:2-2:2-3:1:2-4:1:2-5:1:2-5:3:2-9:2-10:2-11:1:2-12:1:2-13:1:2-13:3:2-15:3:2-17:2-2-18:1:2-19:1:2-20:1:2-20:3:2-22:3:2-24:2:2-25:1:2-26:1:2-27:1:2-27:3:2-29:3:2-31:2:2-32:1:2-33:1:2-34:1:2-34:3:2

Table 11: Shift schedule provided manually and two sample shift schedules obtained by the proposed DE and GRASP (Problem instance 1: Solution no. 1 of DE and Solution no. 4 of GRASP) for Co-auxiliary nurses (ID=15 to 20)

ID	Type	(Day:Shift:Skill)
15	Manual	3:1:3-4:2:3-6:1:3-7:2:3-9:1:3-10:2:3-12:1:3-13:2:3-15:1:3-16:2:3-18:1:3-19:2:3-21:1:3-22:2:3-24:1:3-25:2:3-27:1:3-28:2:3-30:1:3-31:2:3-33:1:3-34:2:3
	DE	3:1:3-4:2:3-6:1:3-7:2:3-9:1:3-10:2:3-12:1:3-13:2:3-15:1:3-16:2:3-18:1:3-19:2:3-21:1:3-22:2:3-24:1:3-25:2:3-27:1:3-28:2:3-30:1:3-31:2:3-33:1:3-34:2:3
	GRASP	3:1:3-4:2:3-6:1:3-7:2:3-9:1:3-10:2:3-12:1:3-13:2:3-15:1:3-16:2:3-18:1:3-19:2:3-21:1:3-22:2:3-24:1:3-25:2:3-27:1:3-28:2:3-30:1:3-31:2:3-33:1:3-34:2:3
16	Manual	3:1:3-4:2:3-6:1:3-7:2:3-9:1:3-10:2:3-12:1:3-13:2:3-15:1:3-16:2:3-18:1:3-19:2:3-21:1:3-22:2:3-24:1:3-25:2:3-27:1:3-28:2:3-30:1:3-31:2:3-33:1:3-34:2:3
	DE	3:1:3-4:2:3-6:1:3-7:2:3-9:1:3-10:2:3-12:1:3-13:2:3-15:1:3-16:2:3-18:1:3-19:2:3-21:1:3-22:2:3-24:1:3-25:2:3-27:1:3-28:2:3-30:1:3-31:2:3-33:1:3-34:2:3
	GRASP	3:1:3-4:2:3-6:1:3-7:2:3-9:1:3-10:2:3-12:1:3-13:2:3-15:1:3-16:2:3-18:1:3-19:2:3-21:1:3-22:2:3-24:1:3-25:2:3-27:1:3-28:2:3-30:1:3-31:2:3-33:1:3-34:2:3
17	Manual	4:1:3-5:2:3-7:1:3-8:2:3-10:1:3-11:2:3-13:1:3-14:2:3-16:1:3-17:2:3-19:1:3-20:2:3-22:1:3-23:2:3-25:1:3-26:2:3-28:1:3-29:2:3-31:1:3-32:2:3-34:1:3-35:2:3
	DE	2:2:3-4:1:3-5:2:3-7:1:3-8:2:3-10:1:3-11:2:3-13:1:3-14:2:3-16:1:3-17:2:3-19:1:3-20:2:3-22:1:3-23:2:3-25:1:3-26:2:3-28:1:3-29:2:3-31:1:3-32:2:3-34:1:3-35:2:3
	GRASP	2:2:3-4:1:3-5:2:3-7:1:3-8:2:3-10:1:3-11:2:3-13:1:3-14:2:3-16:1:3-17:2:3-19:1:3-20:2:3-22:1:3-23:2:3-25:1:3-26:2:3-28:1:3-29:2:3-31:1:3-32:2:3-34:1:3-35:2:3
18	Manual	4:1:3-5:2:3-7:1:3-8:2:3-10:1:3-11:2:3-13:1:3-14:2:3-16:1:3-17:2:3-19:1:3-20:2:3-22:1:3-23:2:3-25:1:3-26:2:3-28:1:3-29:2:3-31:1:3-32:2:3-34:1:3-35:2:3
	DE	2:2:3-4:1:3-5:2:3-7:1:3-8:2:3-10:1:3-11:2:3-13:1:3-14:2:3-16:1:3-17:2:3-19:1:3-20:2:3-22:1:3-23:2:3-25:1:3-26:2:3-28:1:3-29:2:3-31:1:3-32:2:3-34:1:3-35:2:3
	GRASP	2:2:3-4:1:3-5:2:3-7:1:3-8:2:3-10:1:3-11:2:3-13:1:3-14:2:3-16:1:3-17:2:3-19:1:3-20:2:3-22:1:3-23:2:3-25:1:3-26:2:3-28:1:3-29:2:3-31:1:3-32:2:3-34:1:3-35:2:3
19	Manual	2:1:3-3:2:3-5:1:3-6:2:3-8:1:3-9:2:3-11:1:3-12:2:3-14:1:3-15:2:3-17:1:3-18:2:3-20:1:3-21:2:3-23:1:3-24:2:3-26:1:3-27:2:3-29:1:3-30:2:3-32:1:3-33:2:3-35:1:3
	DE	2:1:3-3:2:3-5:1:3-6:2:3-8:1:3-9:2:3-11:1:3-12:2:3-14:1:3-15:2:3-17:1:3-18:2:3-20:1:3-21:2:3-23:1:3-24:2:3-26:1:3-27:2:3-29:1:3-30:2:3-32:1:3-33:2:3-35:1:3
	GRASP	2:1:3-3:2:3-5:1:3-6:2:3-8:1:3-9:2:3-11:1:3-12:2:3-14:1:3-15:2:3-17:1:3-18:2:3-20:1:3-21:2:3-23:1:3-24:2:3-26:1:3-27:2:3-29:1:3-30:2:3-32:1:3-33:2:3-35:1:3
20	Manual	2:1:3-3:2:3-5:1:3-6:2:3-8:1:3-9:2:3-11:1:3-12:2:3-14:1:3-15:2:3-17:1:3-18:2:3-20:1:3-21:2:3-23:1:3-24:2:3-26:1:3-27:2:3-29:1:3-30:2:3-32:1:3-33:2:3-35:1:3
	DE	2:1:3-3:2:3-5:1:3-6:2:3-8:1:3-9:2:3-11:1:3-12:2:3-14:1:3-15:2:3-17:1:3-18:2:3-20:1:3-21:2:3-23:1:3-24:2:3-26:1:3-27:2:3-29:1:3-30:2:3-32:1:3-33:2:3-35:1:3
	GRASP	2:1:3-3:2:3-5:1:3-6:2:3-8:1:3-9:2:3-11:1:3-12:2:3-14:1:3-15:2:3-17:1:3-18:2:3-20:1:3-21:2:3-23:1:3-24:2:3-26:1:3-27:2:3-29:1:3-30:2:3-32:1:3-33:2:3-35:1:3

To validate the reliability of the proposed algorithms, the following comparison metrics are taken into account:

- Quality metrics: This metric puts together the non-dominated solutions found by both algorithms, and measures the ratios between them. In fact, this metric considers the non-dominated solutions found by both algorithms as a set of solutions. It then acquires non-dominated solutions from this set by using non-dominated relations, and measures what percentage of these non-dominated solutions belongs to each algorithm.

- Spacing metric (SM) [20]: This metric is defined by the following formula:

$$SM = \frac{\sum_{i=1}^{N-1} |d_{mean} - d_i|}{(N-1) * d_{mean}} \quad (39)$$

where  $d_i$  is the Euclidean distance between successive solutions in the obtained non-dominated set of solutions, and  $d_{mean}$  is the average of these distances. Spacing metric allows measuring the uniformity of the spread of the solution set points.

- Diversification metric (DM) [20]: This metric measures the spread of the solution set by the following formula:

$$DM = \sqrt{\sum_{i=1}^n \max(\|x_t^i - y_t^i\|)} \quad (40)$$

where  $\|x_t^i - y_t^i\|$  is the Euclidean distance between the non-dominated solutions  $x_t^i$  and  $y_t^i$ .

- Hyper area ratio metric (HR) [20]: This metric is defined by the following formula:

$$HR = \frac{H_{es}}{H_{sc}} \quad (41)$$

where  $H_{es}$  and  $H_{sc}$  are the areas occupied by the solution set of practical PF and the solution set of theoretical PF respectively. (The set of all globally optimal solutions is called the Pareto optimal set, and the set of all non-dominated objective vectors is called the Pareto front (PF). Obtaining an approximation to either the Pareto optimal set or the PF is referred to as Pareto optimisation). Since obtaining theoretical PF is practically impossible, it is produced by all non-dominated solutions provided from both algorithms in this paper. If the value of HR is smaller than 1, the points of practical tradeoff surface are not spread over the whole tradeoff surface, and if the value of HR is greater than 1, the practical tradeoff surface is distant from the tradeoff surface.

- The relative metric  $C(F', F'')$  [20]: The following function transforms two given sets  $F'$  and  $F''$  to a real value in the interval  $[0, 1]$ :

$$C(F', F'') = \frac{\text{card}(\{x'' \in F''; \exists x' \in F' | x' < x''\})}{\text{card}(F'')} \quad (42)$$

where  $\text{card}(F'')$  corresponds to the number of elements inside the set  $F''$ , and  $x' < x''$  means that the vector  $x'$  dominates the vector  $x''$ . Therefore, this metric allows for computing the portion of the surface  $F''$  that is dominated by the tradeoff surface  $F'$ . If points in  $F'$  dominate to all points in  $F''$ , then by definition  $C(F', F'') = 1$ .  $C(F', F'') = 0$  implies the opposite.

Table 12 lists the computational time of both algorithms and the values of these comparison metrics.

Table 12: Computational results

		<i>Computational time (second)</i>	<i>QM</i>	<i>SM</i>	<i>DM</i>	<i>HR</i>
Problem instance 1	DE	47.62	%100	0.4419	49.25	1
	GRASP	89.41	%0	1.7343	25.36	0.2
Problem instance 2	DE	60.76	%85.18	0.6460	127.05	0.8518
	GRASP	161.23	%14.82	1.0058	49.76	0.4444
		<i>C(DE, GRASP)</i>	<i>C(GRASP, DE)</i>			
Problem instance 1	1		0			
		<i>C(DE, GRASP)</i>	<i>C(GRASP, DE)</i>			
Problem instance 2	0.67		0			

The proposed DE takes less time than the proposed GRASP. Meanwhile, the values of the comparison metrics show that the proposed DE is superior to the proposed GRASP in both problem instances. As a result:

- The proposed DE consumes less time than the proposed GRASP.
- The proposed DE can obtain a greater number of Pareto-optimal solutions with higher qualities than the proposed GRASP.
- The proposed DE acquires non-dominated solutions that have lower less values of spacing metric. This shows that the non-dominated solutions obtained by the proposed DE are distributed more uniformly in comparison with the proposed GRASP.
- The values of the diversification metric in the proposed DE are sufficiently greater than the proposed GRASP.

- The values of hyper area ratio metric show that the proposed DE has a higher convergence towards the tradeoff surface, i.e. the closer the value of this metric is to 1, the better is the tradeoff surface.
- Based on the relative metric, it is obvious that the proposed DE provides a better performance than the proposed GRASP.

## 5. CONCLUDING REMARKS

This paper has presented a nurse scheduling problem (NSP) based on a real case study in the maternity ward of Sina Hospital in Isfahan that takes both hospital managers' objectives and nurses' preferences into consideration. Because of the NP-hard class of an NSP, two meta-heuristic algorithms - differential evolution (DE) and greedy randomised adaptive search procedure (GRASP) - have been proposed to solve the given problem. Two problem instances have been designed to evaluate the performance of the constraints considered in the model and the proposed algorithms. The results obtained from the small-sized problem (Problem instance 1) have been compared with a shift schedule that has been drawn up manually by the maternity ward supervisor. The related results have confirmed the performance of the model. Furthermore, some useful comparison metrics - quality metric, spacing metric, diversification metric, hyper area ratio metric, and the relative metric - have been applied to validate the efficiency of the proposed algorithms. The results in this paper show that the proposed DE outperforms the proposed GRASP, and also show that it is capable of improving the quality of acquired solutions. The model presented is applicable to other wards and hospitals with some modifications. Future research will be developed around the fuzzy nature of hospital managers' objectives and nurses' preferences, and new solution methods.

## REFERENCES

- [1] Abdennadher, S. & Schlenker, H. 1999. Nurse scheduling using constraint logic programming, in *Proceedings of the 16th National Conference on Artificial Intelligence*, pp 838-843.
- [2] Aickelin, U. & Dowsland, K.A. 2000. Exploiting problem structure in a genetic algorithm approach to a nurse rostering problem, *Journal of Scheduling*, 3(3) pp 139-153.
- [3] Aickelin, U. & Dowsland, K.A. 2004. An indirect Genetic Algorithm for a nurse-scheduling problem, *Computers & Operations Research*, 31(5), pp 761-778.
- [4] Aiex, R.M., Binato, S. & Resende, M.G.C. 2003. Parallel GRASP with path-relinking for job shop scheduling, *Parallel Computing*, 29(4), pp 393-430.
- [5] Akjiratikar, C., Yenradee, P. & Drake, P.R. 2007. PSO-based algorithm for home care worker scheduling in the UK, *Computers & Industrial Engineering*, 53(4), pp 559-583.
- [6] Alfares, H.K. 1998. An efficient two phase algorithm for cyclic days-off scheduling, *Computers & Operations Research*, 25(11), pp 913-923.
- [7] Alvarez-Valdes, R., Crespo, E., Tamarit, J.M. & Villa, F. 2008. GRASP and path relinking for project scheduling under partially renewable resources, *European Journal of Operational Research*, 189(3), pp 1153-1170.
- [8] Armentano, V.A. & de Franca Filho, M.F. 2007. Minimizing total tardiness in parallel machine scheduling with setup times: An adaptive memory-based GRASP approach, *European Journal of Operational Research*, 183(1), pp 100-114.
- [9] Azaiez, M.N. & Al Sharif, S.S. 2005. A 0-1 goal programming model for nurse scheduling, *Computers & Operations Research*, 32(3), pp 491-507.
- [10] Bard, J. & Purnomo, H.W. 2005. Preference scheduling for nurses using column generation, *European Journal of Operational Research*, 164(2), pp 510-534.
- [11] Bechtold, S.E. & Jacobs, L.W. 1990. Implicit modeling of flexible break assignments in optimal shift scheduling, *Management Science*, 36(11), pp 1339-1351.
- [12] Bellanti, F., Carello, G., Della Croce, F. & Tadei, R. 2004. A greedy-based neighborhood search approach to a nurse rostering problem, *European Journal of Operational Research*, 153(1), pp 28-40.
- [13] Blochliger, I. 2004. Modeling staff scheduling problems: A tutorial, *European Journal of Operational Research*, 158 (3), pp 533-543.
- [14] Bradley, D. & Martin, J. 1990. Continuous personnel scheduling algorithms: A literature review, *Journal of the Society of Health Systems*, 2(2), pp 8-23.
- [15] Burke, E.K., De Causmaecker, P., Van den Berghe, G. & Van Landeghem, H. 2004. The state of the art of nurse rostering, *Journal of Scheduling*, 7(6), pp 441-499.

- [16] Burke, E.K., Li, J. & Qu, R. 2010. A hybrid model of integer programming and variable neighbourhood search for highly-constrained nurse rostering problems, *European Journal of Operational Research*, 203(2), pp 484-493.
- [17] Cerulli, R., Gaudioso, M. & Mautone, R. 1992. A class of manpower scheduling problems, *Methods and Models of Operations Research*, 36(1), pp 93-105.
- [18] Cheang, B., Li, H., Lim, A. & Rodrigues, B. 2003. Nurse rostering problems - a bibliographic survey, *European Journal of Operational Research*, 151(3), pp 447-460.
- [19] Chiamonte, M.V. & Chiamonte, L.M. 2008. An agent-based nurse rostering system under minimal staffing conditions, *International Journal of Production Economics*, 114(2), pp 697- 713.
- [20] Coello Coello, C.A., Lamont, G.B. & Van Veldhuizen, D.A. 2007. *Evolutionary algorithms for solving multi-objective problems*, Springer Science+Business Media, LLC, New York, USA.
- [21] Costa, M.C., Jarray, F. & Picouleau, C. 2006. An acyclic days-off scheduling problem, *4OR: A Quarterly Journal of Operations Research*, 4(1), pp 73-85.
- [22] Damak, N., Jarboui, B., Siarry, P. & Loukil, T. 2009. Differential evolution for solving multimode resource-constrained project scheduling problems, *Computers & Operations Research*, 36(9), pp 2653-2659.
- [23] Dowland, K.A. 1998. Nurse scheduling with tabu search and strategic oscillation, *European Journal of Operational Research*, 106(2-3), pp 393-407.
- [24] Easton, F., Rossin, D. & Borders, W. 1992. Analysis of alternative scheduling policies for hospital nurses, *Production and Operations Management*, 1(2), pp 159-174.
- [25] Emmons, H. & Fuh, D.S. 1997. Sizing and scheduling a full-time and part-time workforce with off-day and off-weekend constraints, *Annals of Operations Research*, 70(1), pp 473-492.
- [26] Ernst, A.T., Jiang, E., Krishnamoorthy, M. & Sier, D. 2004. Staff scheduling and rostering: A review of applications, methods and models, *European Journal of Operational Research*, 153(1), pp 3-27.
- [27] Feo, T. & Resende, M.G.C. 1995. Greedy randomized adaptive search procedures, *Journal of Global Optimization*, 6(2), pp 109-133.
- [28] Glass, C.A. & Knight, R.A. 2010. The nurse rostering problem: A critical appraisal of the problem structure, *European Journal of Operational Research*, 202(2), pp 379-389.
- [29] Gutjahr, W.J. & Rauner, M.S. 2007. An ACO algorithm for a dynamic regional nurse-scheduling problem in Austria, *Computers & Operations Research*, 34(3), pp 642-666.
- [30] Hung, R. 1991. A cyclical schedule of 10-Hour, four-day workweeks, *Nursing Management*, 22(5), pp 30-33.
- [31] Karp, R.M. 1972. Reducibility among combinatorial problems, in R.E. Miller, J.W. Thatcher (eds), *Complexity of Computer Computations*, Plenum Press, New York, pp 85-103.
- [32] Lagodimos, A.G. & Leopoulos, V. 2000. Greedy heuristic algorithms for manpower shift planning, *International Journal of Production Economics*, 68(1), pp 95-106.
- [33] Meisels, A., Gudes, E. & Solotorevski, G. 1995. Employee timetabling, constraint networks and knowledge-based rules: A mixed approach, in E.K. Burke, P. Ross (eds), *Practice and Theory of Automated Timetabling*; First International Conference Edinburgh, Springer, pp 93-105.
- [34] Meyer auf'm Hofe, H. 1997. ConPlan/SIEDAplan: Personnel assignment as a problem of hierarchical constraint satisfaction, in *Proceedings of the 3rd International Conference on Practical Application of Constraint Technology*, pp 257-272.
- [35] Miller, H.E. 1984. Implementing self scheduling, *The Journal of Nursing Administration*, 14(3), pp 33-36.
- [36] Mohan, S. 2008. Scheduling part-time personnel with availability restrictions and preferences to maximize employee satisfaction, *Mathematical and Computer Modeling*, 48(11-12), pp 1806-1813.
- [37] Musliu, N., Gartner, J. & Slany, W. 2002. Efficient generation of rotating workforce schedules, *Discrete Applied Mathematics*, 118(1-2), pp 85-98.
- [38] Nascimento, M.C.V., Resende, M.G.C. & Toledo, F.M.B. 2010. GRASP heuristic with path-relinking for the multi-plant capacitated lot sizing problem, *European Journal of Operational Research*, 200(3), pp 747-754.
- [39] Osogami, T. & Imai, H. 2000. Classification of various neighborhood operations for the nurse scheduling problem, *Lecture Notes in Computer Science*, 1969, pp 72-83.
- [40] Pan, Q.K., Tasgetiren, M.F. & Liang, Y.C. 2008. A discrete differential evolution algorithm for the permutation flow shop scheduling problem, *Computers & Industrial Engineering*, 55(4), pp 795-816.
- [41] Pan, Q.K., Wang, L., Gao, L. & Li, W.D. 2011. An effective hybrid discrete differential evolution algorithm for the flow shop scheduling with intermediate buffers, *Information Sciences*, 181(3), pp 668-685.
- [42] Resende, M.G.C. 2000. *Greedy randomized adaptive search procedures (GRASP)*, Technical Report TR 98.41.1, AT&T Labs-Research.
- [43] Ringl, K.K. & Dotson, L. 1989. Self-scheduling for professional nurses, *Nursing Management*, 20(2), pp 42-44.

- [44] Siferd, S.P. & Benton, W.C. 1992. Workforce staffing and scheduling: Hospital nursing specific models, *European Journal of Operational Research*, 60(3), pp 233-246.
- [45] Sitompul, D. & Randhawa, S. 1990. Nurse scheduling models: A state-of-the-art review, *Journal of the Society of Health Systems*, 2(1), pp 62-72.
- [46] Smith-Daniels, V., Schweikhart, S. & Smith-Daniels, D. 1988. Capacity management in health care services: Review and future research directions, *Decision Sciences*, 19(4), pp 889-919.
- [47] Storn, R. & Price, K. 1995. *Differential evolution: A simple and efficient adaptive scheme for global optimization over continuous spaces*, Technical Report TR-95-012, International Computer Science Institute, Berkeley, California.
- [48] Storn, R. & Price, K. 1997. Differential evolution: A simple and efficient heuristic for global optimization over continuous spaces, *Journal of Global Optimization*, 11(4), pp 341-359.
- [49] Techawiboonwong, A., Yenradee, P. & Das, S.K. 2006. A master scheduling model with skilled and unskilled temporary workers, *International Journal of Production Economics*, 103(2), pp 798-809.
- [50] Topaloglu, S. & Selim, H. 2010. Nurse scheduling using fuzzy modeling approach, *Fuzzy Sets and Systems*, 161(11), pp 1543-1563.
- [51] Tsai, C.C. & Li, S.H.A. 2009. A two-stage modeling with genetic algorithms for the nurse scheduling problem, *Expert Systems with Applications*, 36(5), pp 9506-9512.
- [52] Valls, V., Perez, A. & Quintanilla, S. 2009. Skilled workforce scheduling in service centers, *European Journal of Operational Research*, 193(3), pp 791-804.
- [53] Wang, L., Pan, Q.K., Suganthan, P.N., Wang, W.H. & Wang, Y.M. 2010. A novel hybrid discrete differential evolution algorithm for blocking flow shop scheduling problems, *Computers & Operations Research*, 37(3), pp 509-520.
- [54] Warner, M. 1976. Nurse staffing, scheduling, and reallocation in the hospital, *Hospital & Health Services Administration*, 21(3), pp 77-90.
- [55] Winstanley, G. 2004. Distributed and devolved work allocation planning, *Applied Artificial Intelligence*, 18(2), pp 97-115.