A Fuzzy-based Particle Swarm Optimization Algorithm for Nurse Scheduling

Michael Mutingi, Charles Mbohwa

***Abstract*—The nurse scheduling problem (NSP) has a great impact on the quality and efficiency of health care operations. Healthcare Operations Analysts have to assign daily shifts to nurses over the planning horizon, so that operations costs are minimized, health care quality is improved, and the nursing staff is satisfied. Due to conflicting objectives and a myriad of restrictions imposed by labor laws, company requirements, and other legislative laws, the NSP is a hard problem. In this paper we present a particle swarm optimization-based algorithm that relies on a heuristic mechanism that incorporates hard constraints to improve the computational efficiency of the algorithm. Further, we incorporate soft constraints into objective function evaluation to guide the algorithm. Results from illustrative examples show that the algorithm is effective and efficient, even over large scale problems.**

***Index Terms*—Nurse scheduling problem, nurse rostering, personnel scheduling, metaheuristics, particle swarm optimization**

I. Introduction

THE nurse scheduling problem (NSP) is concerned with the construction and assignment of shift schedules to available nurses on a daily, weekly or monthly basis [1] [2]. The primary goal is to meet the healthcare service needs at an acceptable cost. The effectiveness of the shift schedules impacts the worker morale, healthcare service quality, the recruitment process, the healthcare operations budgets, and the survivability of the healthcare organization in the long run. However, the NSP is a hard combinatorial optimization problem common in healthcare organizations [1] [2] [3].

The main complicating features in nurse scheduling are as follows: (i) health care organizations operate round the clock over seven days a week, (ii) nurse preferences have to be accommodated satisfactorily, (iii) legislative restrictions imposed on staff rosters have to be observed, and (iv) organizational requirements need to be satisfied [1] [4]. Consequently, the nurse scheduling process becomes even more time consuming, especially in the case of a myriad of nurse preferences and other constraints. As such, the NSP has attracted considerable attention among researchers and practicing decision makers in healthcare organizations.

Manuscript received on 06 March, 2014, revised on 10 April, 2014.

Michael Mutingi is a doctoral student with the Faculty of Engineering and the Built Environment, University of Johannesburg, Johannesburg, Bunting Road Campus, P. O. Box 524, Auckland Park 2006, South Africa (phone: +27789578693; e-mail: mmutingi@gmail.com).

Charles Mbohwa is a Professor with the Department of Quality and Operations Management, University of Johannesburg, Johannesburg, Bunting Road Campus, P. O. Box 524, Auckland Park 2006, South Africa (e-mail: cmbohwa@uj.ac.za).

Remarkable empirical and hypothetical studies exist in literature [1] [2] [5] [6]. Cheang et al. [2] presents a noteworthy bibliography on the NSP problem. A comprehensive annotated bibliography on personnel scheduling and rostering is presented in [6]. Various authors have recommended the use of heuristic methods, metaheuristics, constructive heuristics, and hybrid approaches [1-3] [5-8]. The most popular metaheuristic approaches include evolutionary algorithms [9] [10], genetic algorithm [8] [11-13], and particle swarm optimization [14] [15]. To enhance their effectiveness, these approaches are normally combined with problem specific heuristics [8] [9]. Nevertheless, the healthcare environment is characterized by imprecise and conflicting management goals and nurse preferences that are difficult to quantify and evaluate. Incorporating fuzzy management goals, nurse preferences and wishes, and schedule fairness is a non-trivial challenge. Furthermore, it is difficult to incorporate the decision maker's qualitative choices. Conventional nurse scheduling approaches are weak in that they prescribe a solution rather than provide alternative solutions from which the decision maker can interactively select the most practical and appropriate solution. Moreover conventional approaches can be trapped in local optima, resulting in excessive computation times.

The current research develops an interactive approach based on particle swarm optimization [15], enhanced by a unique constraint centered coding mechanism to improve the computational efficiency of the algorithm. The objectives of this research are as follows:

1. To describe the nurse scheduling problem and its constraints;
2. To propose an interactive particle swarm-based procedure, that uses fuzzy evaluation;
3. To present illustrative examples, showing the utility of the algorithm.

The remainder of the paper is structured as follows: The next section describes the NSP problem. Section III provides preliminaries to fuzzy concepts and particle swarm optimization (PSO). Section IV presents the proposed interactive fuzzy PSO and its unique constraint centered coding scheme. Section V provides computational experiments and results. Section VI concludes the paper.

II. The Nurse Scheduling Problem

The NSP is a hard optimization problem that involves assignment of shifts and off days to nurses over the planning

TABLE I
TYPICAL SHIFT TYPES IN COUNTRY B

Shift, *w*	Shift Description	Time allocation
1	*d*: day shift	0800 - 1600 hrs
2	n: night shift	1600 - 2400 hrs
3	*l*: late night shift	0000 - 0800 hrs
4	*o*: off days as nurse preferences	

horizon of up to about one month. Oftentimes, the decision maker should consider a myriad of conflicting objectives and preferences associated with the healthcare organization and individual nurses [1]. Nurses have specific skills and contractual agreements limiting the number of shifts in a week, number of off days, and number of nurses for each shift, among other restrictions. Furthermore, personal preferences, though they may be imprecise in practice, should be taken into account in order to maximize on job satisfaction [3] [8]. For instance, nurses may desire specific days off, certain specific shifts, or number of working days per period. From our studies of hospitals in country B (name withheld for anonymity), each nurse is entitled to day shift *d*, night shift *n*, and late night shift *l*, with some holidays or off days *o*, as listed in Table I.

A. Problem Description

The NSP is described thus. Assume that m and n represent the number of nursing staff and days, respectively. Then, the problem is an $m \times n$ matrix such that each X_{ijk} element in the matrix expresses that nurse i works shift k on day j. In general, the objective is to search for a schedule that satisfies a given set of hard constraints. However, in practice, the wishes or preferences of individual nurses must be satisfied as much as possible. As a result, two categories of shift constraints are involved: (a) hard constrains, which must always be satisfied, and (b) soft constraints, which are often imprecise, but must be satisfied to the highest degree possible. While violation of hard constraints constitutes an infeasible schedule, violation of soft constraints is permissible to some extent, but at the expense of the quality of the schedule. Soft constraints are, therefore, added to improve the quality of the schedule.

B. Constraints

A study of the NSP problem in country B yielded the list of constraints shown in Table II. These constraints are categorized into hard constraints (C1 to C5) and soft nurse preferences (P1 to P3). The set of hard constraints consists of daily restrictions arising from legislative laws. On the other hand, the set of soft constraints arise from nurse preferences. The main challenge is to incorporate these constraints into the scheduling procedure, so as to improve the schedule quality. Our PSO-based algorithm seeks to incorporate these constraints in its coding structure.

TABLE II
TYPICAL CONSTRAINTS FOR THE NSP

Constraints		Description of the constraint
Daily Restrictions	C1	Assign each nurse at most one shift per day.
	C2	The assigned *d*, *n* or *l* shifts ≥ required *d*, *n* or *l* shifts, respectively.
	C3	A (*n*-*d*), (*n*-*l*), or (*l*-*d*) shift combination (sequence) is not permissible.
	C4	Assigned legal holidays = number of legal holidays.
	C5	Interval between night shifts should be at least 1 week.
Nurse Preferences	P1	Preferred or desired day off or holidays.
	P2	Fairness or equality of shifts for each nursing staff
	P3	Congeniality - Compatible or preferable shift assignments among work mates

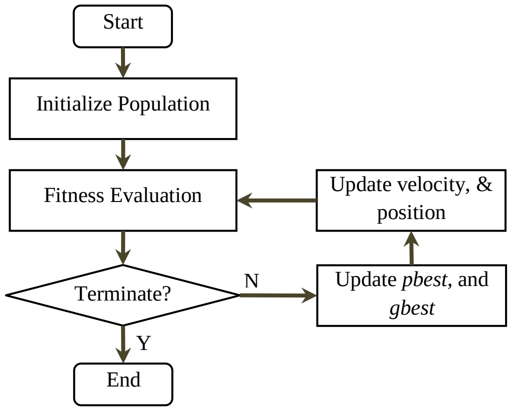

Fig. 1 Flowchart for the basic PSO

III. PRELIMINARIES

A. Basic Particle Swarm Optimization

Particle swarm optimization (PSO) is a stochastic optimization technique motivated by the social behavior of fish schooling and bird flocking [15] [16]. In PSO, the swarm of particles flies through the search space. The PSO mechanism uses a velocity vector to update the current position of each particle in the swarm. While flying, each particle in the swarm adjusts its position based on its own experience and that of the most successful particle. The velocity v_i and the position x_i of each particle i are updated, respectively, as follows:

$$v_i(t+1) = v_i(t) + c_1 \cdot \eta_1 \cdot (pbest_i(t) - x_i(t)) + c_2 \cdot \eta_2 \cdot (gbest(t) - x_i(t)) \quad (1)$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \quad (2)$$

where, $v_i(t)$ and $x_i(t)$ are, respectively, the velocity component and the location component of particle i at iteration t; $v_i(t+1)$ and $x_i(t+1)$ are, respectively, the velocity component and the location component of particle i at iteration t + 1; $pbest_i$ is the best location of particle i, and $gbest_i$ is the global best location of the whole swarm; c_1 and c_2 are, respectively, the cognitive and social parameters, and η_1 and η_2 are uniform random numbers in the range [0, 1].

Fig. 1 presents the basic PSO iterative procedure. The algorithm begins by randomly creating an initial swarm (population) of candidate solutions which are then evaluated according to a problem specific fitness function. After evaluation, the algorithm tests for termination condition which is usually set in terms of a pre-set number of iterations. Subsequently, the velocities and positions of particles are updated according to expressions (1) and (2). However, to incorporate the imprecise soft constraints arising from nurse preferences, fuzzy evaluation techniques have to be employed based on fuzzy theory concepts.

ISBN: 978-988-19253-7-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

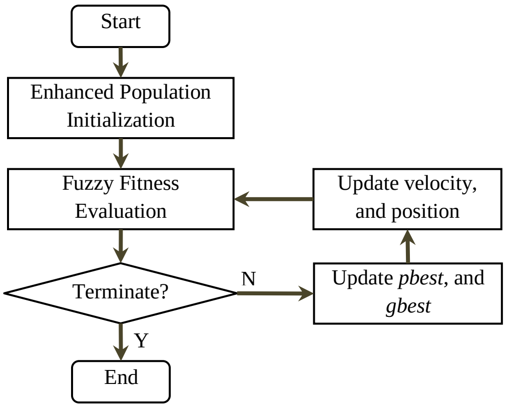

Fig. 2 Proposed PSO-based algorithm structure

B. Fuzzy Set Theory

Fuzzy set theory was initially developed to model imprecision and uncertainty in a non-stochastic sense [17] [18]. A fuzzy number represents imprecise quantities, such as “preferrably 35 working hours per week,” “fair workload,” and “fair schedule.” In this vein, a fuzzy set is a class of objects with no sharp boundary between the objects that belong to that class and those that do not. Fuzzy set theory, unlike Boolean logic, deals with degrees of membership, rather than membership or non-membership [7]. To clarify the concept of fuzzy theory, fuzzy sets can be distinguished from crisp sets: A crisp set is defined thus: Let X be the universe of objects having elements x, and A denote a proper subset of the universe X; $A \subseteq X$. Then, the membership of x in a classical crisp set A is defined by a characteristic function μ_A from X to $\{0,1\}$, such that,

$$\mu_A(x) = \begin{cases} 1 & \text{If } x \in A \\ 0 & \text{If } x \notin A \end{cases} \tag{3}$$

Unlike the crisp set, a fuzzy set is defined thus: Let X be the universe of discourse whose elements are denoted by x. Then, the grade of membership of x in a fuzzy set A is be defined as $\mu_A(x) \in [0,1]$, where $\mu_A(x)$ is the membership function of x in A, which maps each element of X to a membership value in [0,1]. The closer the value of $\mu_A(x)$ to 1.0, the more x belongs to A, and vice versa. The elements of a fuzzy set indicate the value of each element in the set and its grade of membership. Therefore, the fuzzy set A in X is a

		Days									
Nurse	Skill	1	2	3	4	5	6	7	*d*	*n*	*l*
s_1	1	*l*	*n*	*l*	*n*	*d*	*l*	*n*	1	3	3
s_2	1	*o*	*d*	*n*	*l*	*d*	*d*	*l*	3	1	2
s_3	1	*d*	*d*	*d*	*d*	*o*	*n*	*d*	5	1	0
s_4	2	*n*	*l*	*l*	*o*	*d*	*d*	*d*	3	1	2
s_5	2	*d*	*d*	*h*	*n*	*n*	*l*	*n*	2	1	1
s_6	2	*d*	*o*	*d*	*d*	*l*	*n*	*d*	4	1	1
s_7	2	*l*	*n*	*d*	*d*	*l*	*d*	*o*	3	1	2
s_8	2	*n*	*l*	*n*	*l*	*n*	*o*	*l*	0	3	3
	d	3	3	3	3	3	3	3			
	n	2	2	2	2	2	2	2			
	l	2	2	2	2	2	2	2			

Fig. 3 PSO coding scheme – a candidate solution

set of ordered pairs;

$$A = \{x, \mu_A(x) \mid x \in X\} \tag{4}$$

IV. Enhanced Particle Swarm Optimization

Fig. 2 presents a flow chart summarizing the logical of the enhanced PSO. The algorithm consists of initialization, particle coding scheme, fitness evaluation, and velocity update.

A. PSO-based Coding Scheme

To enhance the performance of PSO, we develop a unique coding scheme as shown in Fig. 3. The scheme covers a planning period of 7 days. In this coding scheme, nursing staff s_1, s_2,…,s_8, are allocated one of the four shifts in each day, including the off shift, *o*.

B. Enhanced Initialization Algorithm

The PSO-based algorithm begins by randomly initializing a flock, where each bird is called a particle. Particles fly at a certain velocity, to find a global best position after a number of iterations. Iteratively, each particle adjusts its velocity according to its momentum, its best position (*pbest*) and that of its neighbors (*gbest*), which then determines its new position. Given a search space D, total number of particles N, the position of the i^{th} particle is expressed thus: $x_i = [x_{i1}, x_{i2},...,x_{iD}]$, the best position of the i^{th} particle is given by $pbest_i = [pbest_{i1}, pbest_{i2},...,pbest_{iD}]$, and the velocity of the i^{th} particle is $v_i = [v_{i1}, v_{i2},...,v_{iD}]$.

Algorithm 1. *Initialization Procedure*

```
1.  Assign holidays;
2.  Repeat
3.      Randomly generate an initial shift k1
4.      Repeat
5.          Randomly generate shift kn = rand (d, e, n, o)
6.          If sequence (kn-1, kn) ∉ Forbidden set F Then
7.              Add shift kn to shift pattern Pi
8.              n = n+1
9.          End If
10.     Until (Shift Pattern Pi is complete)
11. Until (Required Shift Patterns, I, are generated)
```

Fig. 4 Enhanced FPSO initialization algorithm

We propose an enhanced initialization algorithm that seeks to satisfy all hard constraints of the NSP problem. Fig. 4 presents the pseudo-code for the proposed algorithm. The procedure begins by randomly allocating holidays to all the nurses. This is followed by random assignment of the *d*, *e*, *n*, *o* shifts, subject to whether or not the subsequent assignment belongs to the forbidden set F that comprises illegal shift sequences.

C. Fitness Evaluation

The fitness or goodness of a solution is evaluated as a function of how much it satisfies the soft constraints. As such, fitness is formulated as a function of the weighted sum of the satisfaction of each of the soft constraints. We assume that the weights are normalized. Furthermore, we represent each soft constraint as a normalized fuzzy membership function whose values fall in the range [0,1].

ISBN: 978-988-19253-7-4
ISSN: 2078-0958 (Print); ISSN: 2078-0966 (Online)

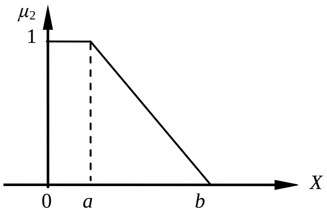

Fig. 5 Linear membership function for congeniality

1) Membership Functions

In order to model nurse preferences into our scheduling algorithm, we use three fuzzy membership functions to model the measure of satisfaction of specific preference functions.

a) Membership Function 1: Congeniality

This membership function measures the quality of shift allocation in terms of nursing staff compatibility (congeniality). It follows that the lower the number of uncongenial shift allocations in a schedule, the higher the quality of that the schedule, and vice versa. The normal practice would be to set a range of acceptable number of uncongenial allocations within which the acceptability of the schedule is 100%, for instance, range [0,*c*], where *c* is the maximum. Fig. 5 shows this phenomenon in form of a interval-valued membership function. Therefore, the membership function is represented by expression;

$$\mu_1 = \begin{cases} 1 & x \le a \\ (b-x)/(b-a) & a \le x \le b \\ 0 & x \ge b \end{cases} \tag{5}$$

where, b is the maximum limit to the number of uncongenial shift allocations; a is the upper limit to the most preferred number of uncongenial shift allocations; x is the actual number of uncongenial allocations.

b) Membership Function 2: Workload Assignment

In order to construct shift patterns with fair workloads, measured in terms of total number of hours allocated to each nurse i, the variation of each nurse workload h_i from the average workload a should be minimized. Thus, this is equivalent to minimizing a function $f = \sum|h_i - a|$.

Since workload assignment should be as fair as possible, the workload variation should be close to zero as much as possible. Therefore, the following membership function holds;

$$\mu_2 = \begin{cases} 1 & x \le c \\ (c-x)/(c-d) & c \le x \le d \\ 0 & x \ge d \end{cases} \tag{6}$$

where, d is the maximum acceptable workload variation; c is the upper limit to the most preferred workload variation; and x is the actual workload variation from the mean workload.

c) Membership Function 3: Days off Allocation

This membership function measures the quality of shift allocation in terms of the variation of the allocated days off or holidays from the mean number of allocated off days or holidays.

$$\mu_3 = \begin{cases} 1 & x \le p \\ (q-x)/(q-p) & p \le x \le q \\ 0 & x \ge q \end{cases} \tag{7}$$

where, q is the limit to the number of days off shift allocations; p is the upper limit to the preferred variation of the number of days off shift allocations; x is the actual variation of days off shift allocations from the mean.

To further improve the quality of the schedule, other membership functions can be added in the same manner. The final fitness function is formulated in terms of its constituent normalized membership functions.

2) The Overall Fitness Function

Since fitness is obtained from the weighted sum of the satisfaction of each of the soft constraints. As such, the final objective function is a function of the normalized functions (membership functions) as follows;

$$z = \sum_f w_f \mu_f \tag{8}$$

where, w_f is the weight of each objective function f, such that $\sum w_f$ = 1.0. The weight w_f offers the decision maker an opportunity to incorporate his/her choices reflecting the preferences of the management and the nurses. This gives PSO an advantage over other metaheuristic approaches.

D. Velocity and Position Update

Iteratively, the position and velocity at iteration (t+1) are updated according to the following;

$$\begin{aligned} v_i(t+1) = w \cdot v_i(t) + c_1 \cdot \eta_1 \cdot (pbest_i(t) - x_i(t)) \\ + c_2 \cdot \eta_2 \cdot (gbest(t) - x_i(t)) \end{aligned} \tag{9}$$

Algorithm 1. *PSO-based Algorithm Procedure*

```
Input w, η1 , η2, c1, c2, N;
Enhanced Initialization
For i = 1 to N:
 Initialize particle position xi(0) and velocity vi(0);
 Initialize pbesti(0);
End For
Initialize gbest(0);
For i = 1 to N:
 Compute fuzzy fitness f(x), x = (x1, x2,...,xN);
Repeat
    For i = 1 to N:
        Compute fitness fi;
        If (fi > current pbest) then
            Set current value as new pbest;
            If (fi > current gbest) then
                gbest = i;
        End If
    End For;
    For i = 1 to N:
        Find neighborhood best;
        Compute particle velocity vi(t+1);
        Update particle position xi(t+1);
    End For;
Until (maximum iterations is reached);
```

Fig. 6 A pseudo-code for the overall PSO-based algorithm

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{10}$$

where, c_1 and c_2 are constants, η_1 and η_2 are uniformly distributed random variables in [0,1], and w is an inertia weight showing the effect of previous velocity on the new velocity vector.

E. Overall PSO-based Algorithm

Fig. 6 shows the overall procedure of the fuzzy based PSO algorithm. The approach has a number of advantages. First, its procedure is intuitively easy to follow and can be easily implemented in a number of problem situations. In addition, the algorithm is computationally efficient, being able to obtain good solutions within reasonable computation times. Notably, fuzzy evaluation allows the optimization process to accept inferior intermediate solutions which eventually yield to improved solutions. This ensures that instances of infeasible solutions are avoided during algorithm execution. Figure 6 provides a summary of proposed PSO-based algorithm in terms of its pseudo code. In the next section, we present illustrative examples, computational results, together with the relevant discussions.

V. Computational Results and Discussions

The proposed PSO-based algorithm was coded implemented in JAVA on a 3.06 GHz speed processor with 4GB RAM. For the purpose of illustration, we present a typical experiment, together with computational results and pertinent discussions.

Fig. 7 (a) presents a typical candidate solution obtained using the proposed enhanced coding method. The solution

		Days									
Nurse	Skill	1	2	3	4	5	6	7	*d*	*n*	*l*
s_1	1	*o*	*d*	*l*	*l*	*o*	*n*	*n*	1	1	2
s_2	1	*d*	*d*	*d*	*o*	*l*	*l*	*o*	3	0	2
s_3	1	*o*	*d*	*d*	*d*	*l*	*n*	*o*	3	1	1
s_4	2	*d*	*l*	*o*	*l*	*n*	*o*	*d*	2	1	2
s_5	2	*d*	*l*	*l*	*o*	*d*	*d*	*n*	3	1	2
s_6	2	*n*	*n*	*o*	*d*	*d*	*d*	*l*	3	2	1
s_7	2	*n*	*o*	*d*	*d*	*d*	*l*	*l*	3	1	2
s_8	2	*l*	*o*	*n*	*n*	*n*	*o*	*d*	1	3	1
s_9	2	*l*	*n*	*n*	*n*	*o*	*d*	*d*	2	3	1
	d	3	3	3	3	3	3	3			
	n	2	2	2	2	2	2	2			
	l	2	2	2	2	2	2	2			

(a)

		Days									
Nurse	Skill	1	2	3	4	5	6	7	*d*	*n*	*l*
s_1	1	*o*	*l*	*l*	*l*	*o*	*n*	*n*	0	2	3
s_2	1	*d*	*d*	*d*	*o*	*l*	*l*	*o*	3	0	2
s_3	1	*o*	*d*	*d*	*d*	*l*	*n*	*o*	3	1	1
s_4	2	*d*	*d*	*o*	*l*	*n*	*o*	*d*	3	1	1
s_5	2	*l*	*l*	*l*	*o*	*d*	*d*	*n*	2	1	3
s_6	2	*n*	*n*	*o*	*d*	*d*	*l*	*l*	2	2	2
s_7	2	*n*	*o*	*d*	*d*	*d*	*d*	*l*	4	1	1
s_8	2	*d*	*o*	*n*	*n*	*n*	*o*	*d*	2	3	0
s_9	2	*l*	*n*	*n*	*n*	*o*	*d*	*d*	2	3	1
	d	3	3	3	3	3	3	3			
	n	2	2	2	2	2	2	2			
	l	2	2	2	2	2	2	2			

(b)

Fig. 7 Initial and final candidate solutions

satisfies all the hard or absolute constraints. Furthermore, the solution shows a schedule or shift assignment covering a planning horizon of 7 days, where 9 nurses are allocated shift types *d*, *n*, *l*, or *o*. The initial population normally comprises a number of candidates obtained in a similar manner. In this problem, assume that combinations (s_1,s_2), (s_5,s_8) and (s_6,s_9) are known to have a very low congeniality, and we should, as much as possible, avoid assigning them the same shifts. The workload assignment is fair across all the nurses. However, hard constraints are always satisfied. Fig. 6 (b) shows an improved solution obtained after 150 iterations, considering the congeniality preferences.

Further experimentations with large numbers of nurses indicated that the PSO-based algorithm can solve large scale scheduling problems within a reasonable computation time, while respecting all the hard constraints and fulfilling preference constraints as much as possible, in the range of 83%.

VI. Concluding Remarks

The development of efficient and effective decision support tools that can address the nurse scheduling problem in healthcare organizations is imperative. High quality schedules are necessary to improve worker moral, avoiding absenteeism and attrition. In an environment where nurse preferences are ill-defined or fuzzy, the use of fuzzy set theory concepts is a suitable option. In this research, a PSO-based algorithm with a fuzzy goal-based fitness function is proposed to solve the nurse scheduling problem, producing near-optimal solutions. An enhanced solution generation heuristic is developed for better efficiency. Experimental results demonstrated the algorithm is capable of solving large scale nurse scheduling problems. The approach provides useful contributions to academicians as well as practitioners in the health service industry.

A. Contributions to Theory

The PSO-based algorithm proposed in this study is a contribution to the healthcare management science and operations management community. It provides an approach to solving nurse scheduling problems in the presence of imprecise management goals and preferences. The approach emulates the solution process with more realism. Unlike other metaheuristic approaches, As opposed to conventional linear programming methods, the algorithm is capable of handling large-scale problems, while providing good solutions within a reasonable computation time. The algorithm forms a platform for further of decision support system for decision makers in the field. The suggested method is also a useful contribution to the practicing decision maker.

B. Contributions to Practice

The proposed algorithm offers the user an opportunity to use weights in order to incorporate preferences and choices in an interactive manner. It uses an interactive approach that provides a list of good alternative solutions, rather than a single "optimal solution". This is more acceptable to most practicing decision makers in the field. Therefore, the user can utilize information from nurses and the management to make adjustments to the solution process based on weights. The PSO-based method is an effective and efficient algorithm nurse scheduling problems.

ACKNOWLEDGMENT

The authors would like to appreciate the reviewers for their invaluable review comments on the earlier version of the paper.

REFERENCES

[1] B. Cheang, H. Li, A. Lim, and B. Rodrigues, “Nurse rostering problems–a bibliographic survey,” *European Journal of Operational Research*, vol. 151, pp. 447–460, 2003.
[2] A.T. Ernst, H. Jiang, M. Krishnamoorthy, D. Sier, “Staff scheduling and rostering: A review of applications, methods and models,” *European Journal of Operational Research*, vol 153, pp. 3-27, 2004b.
[3] S. Topaloglu, and S. Selim, “Nurse scheduling using fuzzy modeling approach,” *Fuzzy Sets and Systems*, vol. 161, pp. 1543–1563, 2010.
[4] H. K. Alfares, “Survey, categorization, and comparison of recent tour scheduling literature,” *Annals of Operations Research*, vol. 127, pp. 145–175, 2004.
[5] E. Burke, P. Causmaecker, V. G. Berghe, H. Landeghem, “The state of the art of nurse rostering,” *Journal of Scheduling*, vol. 7, pp.441-499, 2004.
[6] A.T. Ernst, , H. Jiang, M. Krishnamoorthy, B. Owens, D. Sier, “An annotated bibliography of personnel scheduling and rostering,” *Annals of Operations Research*, vol 127, pp. 21-144, 2004a.
[7] S. Shaffer, “A rule-based expert system for automated staff scheduling,” *IEEE International Conference on Systems, Man, and Cybernetics, Decision Aiding for Complex Systems*, vol.3 pp. 1691-1696, 1991.
[8] M. Mutingi, and C. Mbohwa, “Healthcare staff scheduling in a fuzzy environment: A fuzzy genetic algorithm approach,” *Proceedings of the 2014 International Conference on Industrial Engineering and Operations Management*, Bali, Indonesia, January 7 – 9, 2014, pp. 3038-3047.
[9] A. Jan, M., Yamamoto, A. Ohuchi, “Evolutionary algorithms for nurse scheduling problem,” *Proceedings of the 2000 Congress on Evolutionary Computation, La Jolla, California*, vol. 1, pp. 196-202, 16-19, 2000.
[10] T. Inoue, T. Furuhashi, H. Maeda, and M. Takaba, “A proposal of combined method of evolutionary algorithm and heuristics for nurse scheduling support system,” *IEEE Transactions on Industrial Electronics*, vol. 50, no.5, 2003, pp. 833-838.
[11] D. E. Goldberg, *Genetic Algorithms: In Search, Optimization & Machine Learning*, Addison-Wesley, Inc., MA, 1989.
[12] J. H. Holland, *Adaptation in Natural and Artificial System*, University of Michigan Press, Ann Arbor, MI, 1975.
[13] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolutionary Programs*. Springer, 1996.
[14] C. Akjiratikarl, P. Yenradee, P.R. Drake, “PSO-based algorithm for home care worker scheduling in the UK,” *Computers & Industrial Engineering*, vol. 53, pp. 559-583, 2007.
[15] J. Kennedy, R.C. Eberhart, “Particle swarm optimization,” *In IEEE International Conference on Neural Networks*, Seoul, Korea, vol. 4, pp. 1942-1948, 1995.
[16] Y. Shi, and R.C. Eberhart, “A modified particle swarm optimizer,*” In IEEE International Conference on Evolutionary Computation*, pp. 69-73, 1998.
[17] M. Sakawa, *Fuzzy Sets and Interactive Multi-Objective Optimization*, Plenum Press, New York, 1993.
[18] R.E. Bellman, and L.A. Zadeh, “Decision making in a fuzzy environment,” *Management Science*, vol. 17, pp. 141–164, 1970.