Nurse Scheduling: A Fuzzy Multi-Criteria Simulated Metamorphosis Approach

Michael Mutingi, *Member, IAENG*, and Charles Mbohwa

***Abstract*—The nurse scheduling problem (NSP) is multi-criteria decision problem concerned with allocation of shift schedules to available nurses over a planning horizon of one week to one month. Developing interactive, multi-objective, and fast optimization approaches for solving the NSP is imperative. The NSP has posed continued challenges to decision makers in healthcare organizations. This paper presents a fuzzy simulated metamorphosis algorithm (FSM), inspired by biological metamorphosis evolution. The algorithm mimics the metamorphosis process by going through three phases, namely, initialization, growth, and maturation. Initialization randomly generates a single candidate solution using a guided constructive heuristic. Subsequently, the algorithm goes through growth and maturation loops, till termination criteria are satisfied. Computational results based on benchmark problems in the literature demonstrate that, compared to related metaheuristic algorithms, FSM is more efficient and effective, producing better solutions within reasonable computation times.**

***Index Terms*— Simulated metamorphosis, evolution, multi-criteria decision methods, optimization, algorithm, metaheuristics**

I. Introduction

THE most desired practical objective in nurse scheduling is to produce high quality work schedules, so that (i) individual nurse preferences are satisfied and workload is balanced, (ii) patients are satisfied with the quality of service, and (iii) management goals are satisfied. Since these desires are often conflicting, imprecise, and uncertain in a non-stochastic sense, decision making is difficult. This situation is commonplace in healthcare organizations [1][2].

In a fuzzy environment, addressing conflicting multi-criteria decision problems requires interactive tools that are fast, flexible, and easily adaptable to specific problem situations [3]. Decision makers often desire to use judicious approaches that can find a cautious tradeoff between the many goals, which is a common scenario in real world problems [4][5]. Addressing ambiguity, imprecision, and uncertainties of the desired goals is highly desirable in practice [5]. For instance, in a hospital setting, where nurses are often allowed to express their preferences on shift schedules, the decision maker has to incorporate the imprecision in preferences and management goals and choices. To achieve shift fairness and equity among the nursing staff, it is important to balance workload assignment. Patient preferences and expectations have to be considered as well. Though imprecise and conflicting, these factors have to be considered when constructing work schedules [1][6].

In view of the above highlighted issues for interactive fuzzy multi-objective optimization approaches, this paper presents a novel fuzzy simulated metamorphosis algorithm, inspired by the biological concepts of metamorphosis evolution. The algorithm is motivated by the need for interactive, fuzzy multi-criteria, and fast optimization approaches to solving problems with fuzzy conflicting goals, preferences, and constraints. In this connection, the specific objectives are as follows:

(1) To present the basic concepts of the biological metamorphosis evolution process;
(2) To derive from metamorphosis, an interactive multi-criteria fuzzy evolutionary algorithm; and,
(3) To apply the algorithm to typical nurse scheduling problems, demonstrating its effectiveness.

The rest of the paper is organized as follows. The next section presents the nurse scheduling problem and the basic metamorphosis algorithm. Section III proposes a fuzzy simulated metamorphosis algorithm. Section IV presents a fuzzy simulated metamorphosis for the nurse scheduling problem. Computational analysis is provided in Section V. Section VI concludes the paper.

II. Preliminaries

This section provides an overview of the nurse scheduling problem, and introduces the concepts of biological metamorphosis.

A. The Nurse Scheduling Problem

The NSP is a hard optimization problem that involves assignment of different types of shifts and off days to nurses over a period of up to one month[1][9]. The decision maker considers a number of conflicting objectives, choices, and preferences associated with the healthcare organization and individual nurses [10][11]. In practices, contractual work agreements govern the number of assignable shifts and off

Manuscript received Feb 19, 2015.

M. Mutingi is a doctoral student with the Faculty of Engineering and the Built Environment, University of Johannesburg, Bunting Road Campus, P. O. Box 524, Auckland Park 2006, Johannesburg, South Africa. He is an Industrial Engineering Lecturer at the Namibia University of Science & Technology, Namibia (phone: +27789578693; e-mail: mmutingi@gmail.com).

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

(Advance online publication: 10 July 2015)

days per week [12]. Imprecise personal preferences should be satisfied as much as possible. Typically nurses are entitled to day shift d, night shift e, and late night shift N, with holidays or days-off O [12] [13] [14]. Table I lists and describes common shift types and their time allocations.

TABLE I
SHIFTS DESCRIPTIONS

Shift	Shift Description	Time allocation
1	D: Day shift	0800 - 1600 hrs
2	E: night shift	1600 - 2400 hrs
3	N: late night shift	0000 - 0800 hrs
4	O: off days as nurse preferences	

The primary aim is to search for a schedule that satisfies a given set of hard constraints while minimizing a specific cost function [10][12]. However, in practice, individual nurse preferences, which are often imprecise, should be satisfied to the highest degree possible; the higher the degree of satisfaction, the higher the schedule quality. This ensures not only healthcare service quality, but also satisfactory healthcare work environment (job satisfaction).

TABLE II
TYPICAL CONSTRAINTS TYPES FOR THE NSP

Constraints	Description of the constraint
Sequence	A1: Shift sequences *n-d*, *e-n*, and *e-d* are not permissible
	A2: Minimum rest time between night shift n
	A3: Maximum and minimum number of working hours
Schedule	B1: Fair or equal total workload assignment
	B2: Interval between night shifts should ≥ 1 week
	B3: Fair number of requested days-off or holiday assigned
Roster	C6: Shift coverage requirements to fulfil service quality
	C2: Tutorship, where a trainer has to work with a trainee
	C3: Congeniality, where workmates are not compatible

In this study, we classify constraints into sequence, schedule, and roster constraints as listed in Table II. A sequence constraint pertains to the successive order of shifts in an individual nurse schedule or shift pattern. A schedule constraint relates to the restrictions on the complete nurse schedule covering the planning period, based on criteria such as workload and number of night shifts. On the other hand, a roster constraint controls the combination of nurse schedules based on criteria such as shift coverage and congeniality.

B. Metamorphosis Evolution

A significant number of heuristic optimization algorithms are nature inspired [16]. Metamorphosis is an evolutionary process common in insects such as butterflies [16] [17]. As illustrated in Fig. 1, the process begins with an egg that

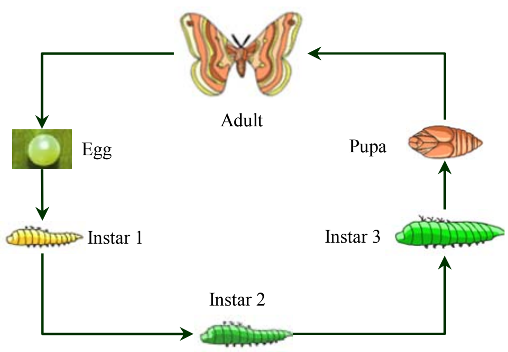

Fig. 1. Metamorphosis evolution cycle.

hatches into an instar larva (instar). Subsequently, the first instar transforms into several instar larvae, then into a pupa, and finally into the adult insect [13]. The process is uniquely characterized with radical evolution and hormone controlled growth and maturation.

When an insect grows and develops, it must periodically shed its rigid exoskeleton in a process called molting. The insect grows a new loose exoskeleton that provides the insect with room for more growth [17]. The insect transforms in body structure as it molts from a juvenile to an adult form, a process called metamorphosis.

The concept of metamorphosis refers to the change of physical form, structure, or substance; a marked and more or less abrupt developmental change in the form or structure of an animal (such as a butterfly or a frog) occurring subsequent to hatching or birth [13]. A species changes body shape and structure at a particular point in its life cycle, such as when a tadpole turns into a frog. Sometimes, in locusts for example, the juvenile form is quite similar to the adult one. In others, they are radically different, and unrecognizable as the same species. The different forms may even entail a completely new lifestyle or habitat, such as when a ground-bound, leaf-eating caterpillar turns into a long distance flying, nectar-eating butterfly.

Insect molting and development is controlled by several hormones [13]. The hormones trigger the insect to shed its exoskeleton and, at the same time, grow from smaller juvenile forms (e.g., a young caterpillar) to larger adult forms (e.g., a winged moth). The hormone that causes an insect to molt is called ecdysone. The hormone, in combination with another, called juvenile hormone, also determines whether the insect will undergo metamorphosis.

III. FUZZY SIMULATED METAMORPHOSIS

Fuzzy Simulated Metamorphosis (FSM) is a development from the basic simulated metamorphosis (SM) evolutionary algorithm originally developed in [18], based on natural biological metamorphosis. FSM is motivated by several fuzzy multi-criteria decision problems in the operations research and operations management community, such as vehicle routing problems [7], nurse scheduling [3][2][6], and task assignment [8]. Such fuzzy decision problems are associated with conflicting imprecise goals, and the need for interactive decision support approaches that can incorporate the choices, intuitions and expert judgments of the decision maker [1]. As a fuzzy multi-criteria heuristic approach, FSM seeks to bridge this gap.

There are three basic phases in the simulated metamorphosis algorithm: initialization, growth, and maturation. Each of these phases has specific operators. Fig. 2 shows an outline of the simulated metamorphosis algorithm. The components of each phase are categorized as follows:

Initialization:	Step 1
Growth Phase:	Step 2, 3, 4, 5
Maturation Phase:	Steps 6, 7, 8

A. Initialization

In the initialization stage, an initial solution is created as a seed for the evolutionary algorithm. In our approach, we use a problem specific heuristic that is guided by hard constraints of the problem. This ensures generation of a feasible initial solution. Alternatively, a decision maker can enter a user-generated solution as a seed. The initial candidate solution s_t (t = 1,…,T) consists of constituent elements e_i (i = 1,…,I), where I is the constituent number of elements in the candidate solution.

Following the creation phase, the algorithm goes into a loop for a maximum of T iterations or generations.

B. Growth

The growth phase comprises the evaluation, transformation, and the regeneration operators.

1) Evaluation

The choice of the evaluation function is very crucial to the success of evaluation operator and the overall algorithm. First, the evaluation function should ensure that it measures the relevant quality of the candidate solution. Second, the function should capture the actual problem characteristics, particularly the imprecise, conflicting and multi-objective nature of the goals and constraints. Third, the fitness function should be easy to evaluate and compute.

The evaluation function F_t, at iteration t, should be a normalized function obtained from its constituent normalized functions denoted by μ_h (h = 1,…,n), where n is the number of constituent objective functions.

In this approach, we use fuzzy multi-factor evaluation method, that is,

$$F_t(s_t) = \sum_h w_h \mu_h(s_t) \qquad (1)$$

where, s_t is the current solution at iteration t; and w_h denotes the weight of the function μ_h. The use of the max-min operator is avoided so as to prevent possible loss of vital information.

2) Transformation

The growth mechanism is achieved through selection and transformation operators. Selection determines whether a constituent element e_i of the candidate solution s_t should be retained for the next iteration, or selected for transformation operation. The goodness or fitness η_i of element e_i (i = 1,…,I) is compared with probability $p_t \in [0,1]$, generated at each iteration t. That is, if $\eta_i \le p_t$, then e_i is transformed, otherwise, it will survive into the next iteration. Deriving from the biological metamorphosis, the magnitude of p_t should decrease over time to guarantee convergence. From our preliminary empirical computations, p_t should follow a decay function of the form,

$$p_t = p_0 e^{-at/T} \qquad (2)$$

where, $p_0 \in [0,1]$ is a randomly generated number; T is the maximum number of iterations; a is an adjustment factor.

It follows that the higher the goodness, the higher the likelihood of survival in the current solution. Therefore, elements with low goodness are subjected to growth. The magnitude of p_t controls the growth rate, which emulates the inhibition or juvenile hormone.

To avoid loss of performing elements, new elements are compared with the rejected ones, keeping the better ones. A pre-determined number of rejected elements are kept in the reject list Q for future use in the regeneration stage.

The regeneration operator has a repair mechanism that considers the feasibility of the candidate solution. All infeasible elements are repaired using problem domain specific heuristics, developed from problem constraints. Elements in Q are used as food for enhancing the repair mechanism.

After regeneration, the candidate solution is tested for readiness for transition to the maturation phase. This is controlled by the dissatisfaction level (juvenile hormone) m_t at iteration t, represented by the expression,

$$m_t = 1 - \mu_1 \wedge \mu_2 \wedge \ldots \wedge \mu_n \qquad (3)$$

Here, $\mu_1,\ldots,\mu_n$, represent the satisfaction level of the respective objective functions; "$\wedge$" is the min operator. This implies that the growth phase repeats until a pre-defined acceptable dissatisfaction m_0 is reached. However, if there is no significant change in m_t after a pre-defined number of trials, then the algorithm proceeds to the maturation phase.

C. Maturation

The maturation phase is a loop consisting of intensification and post-processing operators. The aim is to bring to maturity the candidate solution, so as to obtain the best solution.

1) Intensification

The aim of the intensification operator is to ensure sufficient search of an improved solution in the neighborhood of the current solution. This helps to improve the current solution further. Howbeit, at this stage, the juvenile hormone has ceased to control or balance the growth of the solution according to the constituent fitness functions.

2) Post-processing

The post-processing operator is user-guided; it allows the user to interactively make expert changes to the candidate solution, and to re-run the intensification operator. As such, the termination of the maturation phase is user determined. This also ensures that expert knowledge and intuition are incorporated into the solution procedure. This enhances the interactive search power of the algorithm.

IV. Fuzzy Simulated Metamorphosis for Nurse Scheduling

In this section, we present an application of simulated metamorphosis for nurse scheduling in a fuzzy environment with multiple objectives.

A. FSM Encoding Scheme

To enhance the FSM performance, a unique coding scheme is proposed. As an illustration, Fig. 3 presents a typical scheduling problem with 8 nurses to be scheduled into 3 types of shifts, namely day (D), evening (E), and night (N), including the day-off shift (O). The coding scheme covers a planning period of 7 days. In this coding scheme, the nurses are allocated one of the four shifts in each day, subject to specific shift sequence constraints,

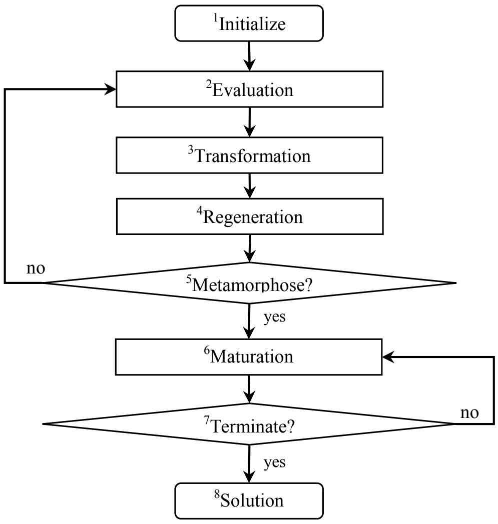

Fig. 2. Fuzzy Simulated Metamorphosis Procedure

schedule constraints and roster constraints, as outlined in the previous section.

B. Initialization

The initialization algorithm is designed to generate a good initial solution, ensuring that all sequence constraints are not violated. Fig. 4 presents the enhanced initialization algorithm. The algorithm generates an initial shift s_1 at random. Successively, the algorithm generates shift s_{k+1}, and tests whether or not the sequence formed by the adding a shift s_{k+1} is not a subset of the predetermined set of forbidden shifts F. An example of a forbidden set is F = {N-D, N-E, E-D}. In addition, the workload of the current sequence $[s_1s_2…s_{k+1}]$ should not exceed the maximum allowable workload w_{max}.

	Days									
Nurse	Mon	Tue	Wed	Thu	Fri	Sat	Sun	*d*	*e*	*n*
Nurse 1	D	D	D	D	D	O	O	5	0	0
Nurse 2	O	D	D	D	D	D	D	6	0	0
Nurse 3	D	O	O	N	N	N	N	1	4	0
Nurse 4	O	O	O	O	E	E	E	0	3	0
Nurse 5	E	E	E	E	O	O	O	0	4	0
Nurse 6	N	N	N	N	N	O	O	0	0	5
Nurse 7	O	O	E	E	E	E	E	0	5	0
Nurse 8	E	E	O	O	E	E	E	0	5	0
Nurse 9	N	N	N	O	O	D	D	2	0	3
D	2	2	2	2	2	2	2			
E	2	2	2	2	2	2	2			
N	2	2	2	2	2	2	2			

Fig. 3. The FMS coding scheme

Algorithm 1. *FSM Initialization Procedure*

Initialize, counter $i = 1$;
Repeat
Initialize $k = 1$
Randomly generate an initial shift s_1
Repeat
  Select shift s_{k+1} = rand (d, e, n, o) with a probability
  If sequence $s = [s_k s_{k+1}] \in$ Forbidden set F, **Then**
    Add shift s_{k+1} to shift pattern P_i with probability p_s
  Else Add shift s_{k+1} to shift pattern P_i
  End If
  If workload w_i of sequence $[s_1s_2…s_k+1] \geq w_{max}$ **Then**
    $s_{k+1} = o$
  End If
  Increment counter $k = k+1$
Until (Shift Pattern P_i is complete)
Increment counter $i = i + 1$
Until (Required schedules, I, are generated)
Return solution

Fig. 4. Enhanced FSM initialization algorithm

C. Growth Phase

1) Evaluation

The goodness, fitness, or quality of a solution is a function of how much it satisfies soft constraints. As such, fitness is a function of the weighted sum of the satisfaction of soft constraints. Thus, each soft constraint is represented as a normalized fuzzy membership function in [0,1]. In this study, two types of membership functions are used: (a) triangular functions, and (b) interval-valued functions, as shown in Fig. 5.

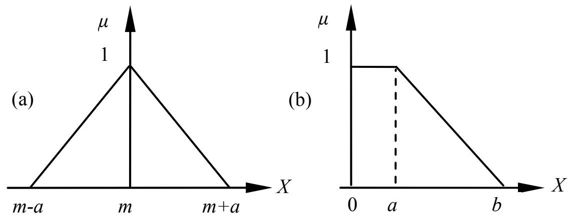

Fig. 5. Linear membership functions

In (a), the satisfaction level is represented by a fuzzy number A<m,a>, where m denotes the center of the fuzzy parameter with width a. The corresponding membership function is,

$$\mu_A(x) = \begin{cases} 1 - \dfrac{|m-x|}{a} & \text{If } m-a \leq x \leq m+a \\ 0 & \text{If otherwise} \end{cases} \tag{4}$$

In (b), the satisfaction level is represented by a decreasing linear function where [0,a] is the most desirable range, and b is the maximum acceptable. Therefore, the corresponding function is,

$$\mu_B(x) = \begin{cases} 1 & \text{If } x \leq a \\ (b-x)/(b-a) & \text{If } a \leq x \leq b \\ 0 & \text{If otherwise} \end{cases} \tag{5}$$

Membership Function 1 - Workload Variation:

For fair workload assignment, workload h_i for each nurse i should be as close as possible to the mean workload w. Therefore, the workload variation $x_i = h_i - w$ should be minimized. Assuming a symmetrical triangular membership function, we obtain,

$$\mu_1 = \mu_A(x_i) \tag{6}$$

where, x_i is the workload variation for nurse i from mean m of the fuzzy parameter, with width a.

Membership Function 2 - Allocated Days Off:

This membership function measures the variation of the allocated days off from the mean. We assume a symmetrical triangular membership function as follows;

$$\mu_2 = \mu_A(x_i) \tag{7}$$

where, x_i is the actual variation of days off for nurse i from the mean m of the fuzzy parameter with width a.

Membership Function 3 - Variation of Night Shifts:

For shift fairness the variation x_i of the number of night shifts (shifts e and n) allocated to each nurse i should be as close as possible to the mean allocation m. Assuming a symmetrical triangular membership, we obtain,

$$\mu_3 = \mu_A(x_i) \tag{8}$$

where, x_i is the variation of number of nights shifts allocated to nurse i from mean m of the fuzzy parameter, with width a.

Membership Function 4 – Congeniality:

This membership function measures the compatibility (congeniality) of staff allocated similar shifts; the higher the congenialities, the higher the schedule quality. In practice, a decision maker sets limits to acceptable number of uncongenial shifts x_i for each nurse i to reflect satisfaction level. Assuming interval-valued functions in Fig. 5 (b), the corresponding membership function is,

$$\mu_4 = \mu_B(x_i) \tag{9}$$

where, x_i is the actual number of uncongenial allocations; a is the upper limit to the preferred uncongenial shifts; b is the maximum uncongenial shifts.

Membership Function 5 - Forbidden Shift Sequences:

The number of shifts in the forbidden set affects the quality of the schedule for each nurse. Let the number of forbidden sequences for each nurse i be x_i. The desirable goal is to reduce the forbidden shifts as much as possible. Therefore, this can be represented by a linear interval-valued membership function as follows:

$$\mu_5 = \mu_B(x_i) \tag{10}$$

where, x_i is the actual number of forbidden shift sequence; a and b are the fuzzy parameters of the function.

Membership Function 6: Shift Variation:

For each nurse i, a schedule with a continuous sequence or block of similar shifts is often more desirable than schedules with different types of shifts. For instance, shift sequence [*D D D O O*] with shift variation $x_i = 1$ is more desirable than shift [*D O D O D*] with a variation $x_i = 4$. Therefore, the situation can be represented by a linear interval-valued membership functions,

$$\mu_6 = \mu_B(x_i) \tag{11}$$

where, x_i is the actual number of shift variation; and a and b are the fuzzy parameters of the function.

Membership Function 7 – Understaffing:

High quality schedule minimize as much as possible the understaffing for each shift k. In practice, the level of understaffing $x_j = \sum u_k$ in each day j should be within acceptable limits. This can be represented by a linear interval-valued membership function;

$$\mu_7 = \mu_B(x_j) \tag{12}$$

where, x_j is the actual level of understaffing in day j, and a and b are the fuzzy parameters of the function.

Membership Function 8 – Overstaffing:

For high quality schedule, overstaffing o_k for each shift k should be minimized as much as possible. In a practical setting, the level of overstaffing $x_j = \sum o_k$ for all shifts in each day j should be within acceptable limits, which can be represented by a linear interval-valued membership as follows;

$$\mu_8 = \mu_B(x_j) \tag{13}$$

where, x_j is the actual level of overstaffing in day j, and a and b are the fuzzy parameters of the function.

The Overall Fitness Function:

For each nurse i, schedule fitness is obtained from the weighted sum of the first four membership functions. As such, the fitness for each shift pattern (or element) i is obtained according to the following expression;

$$\eta_i = \sum_{z=1}^{5} w_z \mu_z(x_i) \qquad \forall i \tag{14}$$

where, w_z is the weight of each function μ_z, such that condition $\sum w_z = 1.0$ is satisfied.

Similarly, the fitness according to shift requirement and congeniality in each day j is given by,

$$\lambda_j = \sum_{z=6}^{7} w_z \mu_z(x_j) \qquad \forall j \tag{15}$$

where, w_z is the weight of each function μ_z, with $\sum w_z = 1.0$.

The overall fitness of the candidate solution is given by the expression,

$$f = \left(\frac{\eta}{\omega_1} \wedge 1\right) \wedge \left(\frac{\lambda}{\omega_2} \wedge 1\right) \tag{16}$$

where, $\eta = \eta_1 \wedge \eta_2 \wedge \ldots \wedge \eta_I$; $\lambda = \lambda_1 \wedge \lambda_2 \wedge \ldots \wedge \lambda_J$; ω_1 and ω_2 are the weights associated with η and λ, respectively; and "∧" is the min operator.

The weights w_z, ω_1 and ω_2 offer the decision maker an opportunity to incorporate his/her choices reflecting expert opinion and preferences of the management and the nurses. This feature gives the SM algorithm an added advantage over other methods.

2) Transformation

In NSP, elements are two-fold: one that represents horizontal shift patterns, denoted by e_i, and another representing the vertical shift allocations for each day, denoted by e_j. Fitness η_i and λ_j of each element are probabilistically tested for transformation by comparing with a random number $p_t \in [0,1]$, generated at each iteration t. A dynamically decaying transformation probability limit $p_t = p_0 e^{-t/T}$ is used. Two transformation heuristics, column-wise heuristic and row-wise heuristic are used for improving the shift patterns.

As outlined in Fig. 6, the column-wise heuristic searches for improved shift sequences and schedules in the neighborhood of the current schedule for each nurse. Again, the dynamic transformation probability p_t controls the transformation process.

Fig. 7 presents an outline of the row-wise transformation heuristic. The heuristic searches for improved roster structure in the neighborhood of the current schedule for each nurse.

3) Regeneration

Regeneration repairs infeasible elements using a mechanism similar to the initialization algorithm which incorporates hard constraints. Based on the juvenile hormone level m_t at iteration t, the candidate solution is then tested for readiness for maturation,

$$m_t = 1 - \left(\eta_1 \wedge \eta_2 \wedge \ldots \wedge \eta_I\right) \wedge \left(\lambda_1 \wedge \lambda_2 \wedge \ldots \wedge \lambda_J\right) \tag{17}$$

The growth phase repeats until a pre-defined acceptable dissatisfaction m_0 is reached. However, the algorithm proceeds to the maturation phase if there is no significant change ε in m_t, with ε set in the order of 10^{-6}.

D. *Maturation Phase*

Intensification ensures complete search of a near-optimal solution in the neighbourhood of the current solution. In the post-processing stage the user interactively makes expert changes to the candidate solution, and to execute intensification. Expert knowledge and intuition are coded in form of possible adjustments through weights $w_1,\ldots,w_4$ and ω_1, ω_1. Illustrative computations are presented in the next section.

E. *Strengths of the FSM Algorithm*

The proposed SM algorithm has a number of advantages over related metaheuristics. Contrary to Simulated Annealing (SA) which makes purely random choices to decide the next move, SM employs intelligent selection operation to decide which changes to perform. Furthermore, SM takes advantage of multiple transformation operations on weak elements of the current solution, allowing for more distant changes between successive iterations.

The SM algorithm, like Genetic Algorithm (GA), uses the mechanics of evolution as it progresses from one generation to the other. GA necessarily keeps a number of candidate solutions in each generation as parents, generating offspring by a crossover operator. On the contrary, SM simulates metamorphosis, evolving a single solution under hormonal control. In addition, domain specific heuristics are employed to regenerate and repair the emerging candidate solution, developing it into an improved and complete solution. In retrospect, SM reduces the computation time needed to maintain a large population of candidate solutions in GA.

The selection process in the SM is quite different from GA and other related evolutionary algorithms. While GA uses probabilistic selection to retain a set of good solutions from a population of candidate solutions, SM selects and discards inferior elements of a candidate solution, according to the goodness of each element. This enhances the computational speed of the SM procedure.

At the end of the growth phase, the SM algorithm goes through maturation phase where intensive search process is performed to refine the solution, and possibly obtain an improved final solution. The algorithm allows the decision maker to input his/her managerial choices to guide the search process. This interactive facility gives SM an added advantage over other heuristics.

The proposed algorithm uses hormonal control to enhance and guide its global multi-objective optimization process. This significantly eliminates unnecessary search through regions with inferior solutions, hence, improving the search efficiency of the algorithm. In summary, the above mentioned advantages provide the SM algorithm enhanced convergence characteristics that enable the algorithm to perform fewer computations relative to other algorithms.

Algorithm 1: Column-wise transformation heuristic

Initialize iteration $t = 1$;
While ($t \le t_{\max}$) **do**
  While (termination condition) **do**
    With probability $p_c = \min[1 - \lambda, p_t]$;
    Randomly select c_1 = cell with conflict;
    Randomly select c_2 = cell with conflict, but in same column;
    Swap (c_1, c_2);
   Select the best from neighbourhood;
  End While
  $t = t + 1$;
End While

Fig. 6. Pseudo-code for column-wise transformation heuristic

Algorithm 2: Row-wise transformation heuristic

Initialize iteration $t = 1$;
While ($t \le t_{\max}$) **do**
  While (termination condition) **do**
    With probability $p_r = \min[1 - \lambda, p_t]$;
    Randomly select r_1 = cell with conflict;
    Randomly select r_2 = cell with conflict, but in same row;
    Swap (r_1, r_2);
   Select the best from neighbourhood;
  End While
  $t = t + 1$;
End While

Fig. 7. Pseudo-code for row-wise transformation heuristic

V. Computational Analysis

The proposed FSM algorithm was coded in JAVA and implemented on a 3.06 GHz speed processor, with a 4GB RAM. Computational experiments are presented.

A. *Computational Experiments*

To illustrate the effectiveness of the proposed FSM algorithm, three sets of problem cases were used for the experiments: (i) experiment 1, a preliminary experiment adapted from [2], (ii) experiment 2, an extension of problem case in experiment 1, (iii) experiment 3 comprising a set of

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Fitness η_i
Nurse 1	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	0.31
Nurse 2	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.33
Nurse 3	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.33
Nurse 4	E	E	E	E	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.54
Nurse 5	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	E	E	E	E	E	E	E	E	E	0.66
Nurse 6	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	0.49
Nurse 7	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.33
Nurse 8	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.50
Nurse 9	N	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.49
Nurse 10	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.50
Nurse 11	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	0.49
Nurse 12	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.50
Nurse 13	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.50
Nurse 14	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	E	E	E	E	0.57
Nurse 15	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	E	0.31
Fitness λ_j	0.3	0.7	0.7	0.7	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.7	0.7	0.7	0.7	0.7	0.3	0.3	0.3	0.3	0.33

Fig. 8. Initial nurse schedule for experiment 1

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Fitness η_i
Nurse 1	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	1.000
Nurse 2	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	1.000
Nurse 3	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	1.000
Nurse 4	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	1.000
Nurse 5	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	1.000
Nurse 6	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	1.000
Nurse 7	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	1.000
Nurse 8	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	1.000
Nurse 9	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	1.000
Nurse 10	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	1.000
Nurse 11	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	1.000
Nurse 12	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	1.000
Nurse 13	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	1.000
Nurse 14	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	1.000
Nurse 15	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	N	E	1.000
Fitness λ_j	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.000

Fig. 9. Final nurse schedule for experiment 1

20 benchmark problem cases in the literature [14], and (iv) experiment 4 consisting of extensions from the benchmark problems in (iii).

Three sets of problem cases were used for the experiments: (i) experiment 1, a preliminary experiment adapted from [2], (ii) experiment 2, an extension of problem case in experiment 1, (iii) experiment 3 comprising a set of 20 benchmark problem cases in the literature [14], and (iv) experiment 4 consisting of extensions from the benchmark problems in (iii). Problem cases in experiment 3 were obtained from real life situations in healthcare organizations, as reported in [14]. Each experiment includes constraints on shift sequences, length of shift sequences, and length of work and days-off. The number of employees (or groups) for the problems ranges from 7 to 163, to be scheduled over 3 standard shifts; day, evening and night shifts.

The termination criteria are controlled by two conditions: (i) the maximum number of iterations, set at $T_m = 300$, and (ii) the maximum number of iterations with no improvement, set at $T_I = 30$. This implies that the algorithm terminates when either of the conditions is met. Generally, each experiment was executed 50 independent times. Computational results and discussions are presented in the next section.

B. Results and Discussion

1) Experiment 1

The first experimental problem was adapted from [2]. In this problem, there are 15 nurses to be scheduled over a planning horizon of 30 days. Shift sequences N-D, E-N, and E-D are unsatisfactory. The daily requirements for shift *D*, *E*, and *N* are 11, 2 and 2, respectively. The day-off *o* and congeniality preferences were not considered. The initial schedule with this setup is shown in Fig. 8. The fitness values for individual nurses are very low; therefore, the schedule quality is unsatisfactory as can be seen from the low overall fitness.

Fig. 9 shows the final optimal schedule obtained in the preliminary experiments. The overall fitness for the best solution is 1.00. This demonstrates that all the desires and preferences are satisfied and the solution is desirable to patients, staff and the management, according to their expectations.

Table III compares the performance of FSM against basic

TABLE III
A COMPARISON OF FMS WITH OTHER ALGORITHMS

Approach	References	Best Fitness	Success Rate (%)	CPU Time(s)	Iterations
Basic CGA	Jan et al. (2000)	1.00	8.33	**	**
CGA	Jan et al. (2000)	1.00	100	49.00	100
FSM		1.00	100	32.40	40

** value not provided

Cooperative Genetic Algorithm (basic CGA) and improved CGA algorithms reported in Jan et al. (2000). Out of 50 independent runs, the success rate of FSM was 100%, which is comparable to 100% for CGA with 12 independent runs. In each successful run, the FSM algorithm was able to obtain the optimal solution in less than 40 iterations, compared to 100 iterations for CGA. The average computational time was 32.40 seconds. This indicates the superior computational efficiency of FSM compared to CGA.

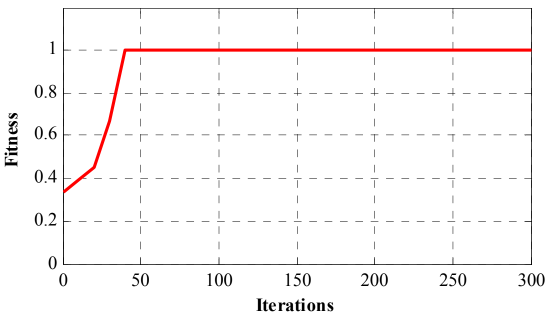

Fig. 10. A transcription of computational results for problem case 1

To further demonstrate the performance of the FMS algorithm, a plot of the intermediate solutions arrived at during the algorithm execution is presented in Fig. 10. The overall fitness value f is plotted against number of iterations t. The fitness value increased from 0.33 at the initialization stage to 1.00 at the 40th iteration, which implies that the algorithm obtained the optimum solution at the 40th iteration, though the user intended the computation to run up to 300 iterations.

2) Experiment 2

This experimental problem is an extension of experiment 1. Here, fuzzy multi-criteria evaluation, including day off and congeniality preferences, is fully utilized to determine the fitness of the candidate solution. The computational experiment consists of 15 nurses that are to be scheduled over a horizon of 30 days.

Fig. 11 presents the initial schedule created using the enhanced initialization constructor. The daily shift requirements for shifts D, E, and N are 10, 2, and 2, respectively. Assume that, due to congeniality issues, nurse combinations (2,4) and (7,10) in any working shift are to be avoided as much as possible. The fitness values for each shift pattern are obtained using expression (11). Similarly, the fitness values for each day are obtained from (12). The maximum number of iterations T_m = 300. The overall fitness at the initialization stage is 0.27, which is very low.

Fig. 12 shows the final nurse schedule obtained by the FSM algorithm. The solution shows a marked improvement in the fitness values of individual shift patterns. Also, there is a 100% satisfaction of the shift requirements in each day, which is a marked improvement from the initial solution. Consequently, the overall fitness value of the final schedule is 0.8197, which is a significant improvement from the initial schedule.

3) Experiment 3

In this experiment, computational results for 20 benchmark problems are reported. For comparative analysis,

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Fitness η_i
Nurse 1	D	D	D	D	O	O	N	N	E	E	D	D	D	D	D	D	D	D	D	N	N	E	E	O	D	D	D	D	D	D	0.45
Nurse 2	N	N	N	N	N	N	N	O	O	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	0.46
Nurse 3	N	E	E	D	D	D	D	O	D	D	D	D	O	O	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	N	0.62
Nurse 4	N	N	N	N	O	O	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	0.43
Nurse 5	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	O	O	D	D	D	D	D	D	D	D	D	D	D	0.71
Nurse 6	D	D	D	D	D	D	D	E	D	D	D	D	D	D	O	D	D	D	D	D	D	D	D	O	O	D	D	D	D	D	0.61
Nurse 7	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	D	D	D	O	D	O	N	N	E	E	D	0.50
Nurse 8	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	O	D	D	D	D	D	D	D	0.65
Nurse 9	D	D	D	D	D	D	D	D	O	O	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.68
Nurse 10	D	D	D	D	D	D	D	D	D	D	D	D	D	O	D	D	D	D	D	O	D	D	D	D	D	D	D	D	D	D	0.54
Nurse 11	D	D	D	D	D	D	D	D	O	O	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.71
Nurse 12	D	D	D	D	D	D	D	D	D	D	D	D	D	D	O	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.65
Nurse 13	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	O	O	D	D	D	D	D	D	D	D	D	D	D	0.71
Nurse 14	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	0.57
Nurse 15	E	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	E	D	0.37
Fitness λ_j	0.7	0.7	0.7	0.3	0.6	0.6	0.7	0.6	0.4	0.6	0.3	0.3	0.5	0.6	0.3	0.3	0.7	0.6	0.6	0.5	0.3	0.7	0.6	0.6	0.6	0.3	0.3	0.7	1.0	0.3	0.27

Fig. 11. Initial nurse schedule for experiment 2

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	Fitness η_i
Nurse 1	D	D	N	N	E	E	O	D	D	D	D	D	D	D	D	O	D	D	D	D	N	N	E	E	D	D	D	D	D	D	0.77
Nurse 2	D	D	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	O	N	N	0.72
Nurse 3	E	D	D	D	D	D	D	D	O	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	O	N	N	E	E	0.70
Nurse 4	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	D	O	N	N	E	E	O	0.70
Nurse 5	D	N	N	E	E	D	D	D	D	D	D	O	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	D	D	0.82
Nurse 6	E	E	O	D	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	O	N	N	E	0.77
Nurse 7	D	D	D	D	D	D	D	D	D	N	N	E	E	O	D	D	D	D	D	O	D	D	N	N	E	E	D	D	D	D	0.75
Nurse 8	O	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	O	D	0.82
Nurse 9	D	D	D	D	D	D	D	N	N	E	E	D	D	D	O	D	O	D	D	D	D	N	N	E	E	D	D	D	D	D	0.79
Nurse 10	N	N	E	E	D	D	D	O	D	D	D	D	D	D	D	D	N	N	E	E	O	D	D	D	D	D	D	D	D	D	0.77
Nurse 11	D	O	D	N	N	E	E	D	D	D	O	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	0.79
Nurse 12	D	D	D	D	O	O	N	N	E	E	D	D	D	D	D	D	D	D	D	N	N	E	E	O	D	D	D	D	D	D	0.82
Nurse 13	D	D	D	D	D	D	D	D	D	O	N	N	E	E	D	D	D	O	O	D	D	D	D	D	N	N	E	E	D	D	0.78
Nurse 14	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	O	O	N	N	E	E	D	D	D	0.78
Nurse 15	D	D	D	D	D	D	D	D	N	N	E	E	D	D	D	D	D	D	D	D	D	O	O	N	N	E	E	D	D	D	0.76
Fitness λ_j	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	1.0	0.88

Fig. 12. Final nurse schedule for experiment 2

TABLE IV
COMPARISON BETWEEN FSM AND OTHER ALGORITHMS

Problem	Success Rate (%)				CPU Time (sec)			
	MC	MC-T	FSEA	FSM	MC	MC-T	FSEA	FSM
1	100	100	100	100	4.77	0.07	0.1	0.09
2	100	100	100	100	1.48	0.07	0.1	0.08
3	100	100	100	100	69.36	0.42	0.18	0.14
4	100	100	100	100	0.12	0.11	0.08	0.1
5	100	100	100	100	15.78	0.43	0.31	0.33
6	100	100	100	100	2.89	0.08	0.09	0.07
7	100	100	100	100	62.51	52.79	4.38	3.16
8	100	100	100	100	32.52	0.74	0.88	0.73
9	50	100	100	100	84.17	15.96	4.87	2.14
10	100	100	100	100	11.40	0.60	0.78	0.66
11	10	100	100	100	254.82	13.15	10.3	7.12
12	100	100	100	100	74.26	1.17	5.33	3.27
13	100	100	100	100	68.32	0.87	2.34	1.2
14	100	100	100	100	8.77	0.76	2.85	1.95
15	15	100	80	100	331.11	159.04	46.34	33.12
16	100	100	100	100	14.48	0.54	3.15	2.19
17	100	100	100	100	54.79	2.16	7.59	5.54
18	100	100	100	100	60.58	6.83	8.35	8.13
19	70	100	100	100	577.96	75.83	72.62	62.2
20	100	100	100	100	183.82	71.38	27.78	31.22
Mean	87.25	100.00	99.00	100.00	95.70	20.15	9.92	8.17

the success rate and the computational time (CPU time) are taken into consideration. For each problem, 10 independent runs were executed using the FSM algorithm. The maximum number of iterations for each run was set at T_m = 300.

Table IV provides a summary of the comparative computational results, in terms of search success rate and average CPU time. FSM is compared with min-conflicts heuristic (MC) and MC with tabu search mechanism (MC-T), as well as FSEA. It can be seen that FSM was able to find satisfactory solutions for all the problems, hence 100% mean success rate, even for large scale problems 15, 19 and 20. The success rate of FSM is comparable to MC-T, but is much better than MC and FSEA. In terms of computational efficiency, FSM outperformed all the other algorithms, with a mean time 8.17 sec, compared to 95.70 sec for MC, 20.15 for MC-T and 9.92 for FSEA.

From these comparative analyses, it can be seen that FMS is capable of producing good feasible solutions satisfying patient expectations, healthcare staff preferences, and management choices.

4) Further Experiments

Further comparative experiments were done between FSM and a workforce scheduling commercial software called First Class Scheduler (FCS) [19]. Table V presents the comparative results of the experiments.

It can be seen that the FSM algorithm outperforms FCS almost in all problem instances, even over large problems such as 7 and 18 which have the same shift requirements over all days and shifts. The symbol "1000 (?)" indicates the problem instances for which FCS could not obtain a solution within 1000 sec [14].

FCS is known to be able to solve medium to large scale problems allowing interaction with the user [14][20]. However, the FSM algorithm outperformed FCS on medium to large scale problems. Therefore, FSM is more efficient and effective.

VI. CONCLUSIONS

Motivated by the biological metamorphosis process and the need to solve multi-objective optimization problems with conflicting and fuzzy goals and constraints, this research proposed a simulated metamorphosis algorithm,

TABLE V.
FURTHER COMPARISON BETWEEN FSM AND FIRST CLASS SCHEDULER (FCS)

Problem	Groups	CPU Time (s)	
		FCS	FSM
1	9	0.9	0.09
2	9	0.4	0.08
3	17	1.9	0.14
4	13	1.7	0.1
5	11	3.5	0.33
6	7	2	0.07
7	29	16.1	3.16
8	16	124	0.73
9	47	>1000 (?)	2.14
10	27	9.5	0.66
11	30	367	7.12
12	20	>1000 (?)	3.27
13	24	>1000 (?)	1.2
14	13	0.54	1.95
15	64	>1000 (?)	33.12
16	29	2.44	2.19
17	33	>1000	5.54
18	53	2.57	8.13
19	120	>1000 (?)	62.2
20	163	>1000 (?)	31.22
Mean	-	>40.97	8.17

based on the concepts of biological evolution in insects, including moths, butterflies, and beetles. The algorithm mimics the hormone controlled evolution process going through initialization, and then growth and maturation loops.

The suggested methods offers a practical approach to optimizing multi-objective problems with fuzzy conflicting goals and constraints such as the nurse scheduling, homecare nurse scheduling, vehicle routing, job shop scheduling, and task assignment. Equipped with the facility to incorporate the user's choices and wishes, the algorithm offers an interactive approach that can accommodate the decision maker's expert intuition and experience, which is otherwise impossible with other optimization algorithms. The algorithm uses novel adaptive parameters that enhance guided intelligent transformation of the solution throughout the search process. By using dynamic hormonal guidance and unique operators, the algorithm dynamically employs two successive iterative loops, working on a single candidate solution, with dynamic and adaptive balance between exploitation and exploration of the solution space. This makes the proposed metaheuristic is efficient and effective.

Fuzzy Simulated Metamorphosis is an invaluable addition to the operations research and management community, specifically to researchers concerned with multi-objective global optimization. Deriving from the computational experimental, the application of the algorithm can be extended to hard problems such as task assignment, vehicle routing, homecare nurse scheduling, and job sequencing.

Using the FSM algorithm, a weekly nurse schedule can be produced easily from a fuzzy multi-criteria view point. However, the practicing decision maker faces two challenges: (i) the allocation of patient visits to nurses during their shifts, in a homecare setting, and (ii) the allocation of daily care tasks to nurses during their shifts, in hospitals.

ACKNOWLEDGMENT

This paper is an extension of the paper: Simulated Metamorphosis: A Novel Optimizer, Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering and Computer Science 2014, WCECS 2014, 22-24 October, 2014, San Francisco, USA, pp 924-929.

REFERENCES

[1] S. Topaloglu, S. Selim, "Nurse scheduling using fuzzy modeling approach," Fuzzy Sets and Systems, vol. 161, pp 1543–1563, 2010.

[2] A. Jan, M. Yamamoto, A. Ohuchi, "Evolutionary algorithms for nurse scheduling problem," IEEE Proceedings of the 2000 Congress on Evolutionary Computation, vol. 1, pp 196 – 203, July 2000.

[3] M. Mutingi, 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, pp 303-312, January 7 – 9, 2014.

[4] P. Vasant, N. Barsoum, , "Hybrid genetic algorithms and line search method for industrial production planning with non-linear fitness function", Engineering Applications of Artificial Intelligence, 22 (4-5), pp 767-777, 2009

[5] M. Mutingi, C. Mbohwa, 2015. A Multi-Criteria Approach for Nurse Scheduling - Fuzzy Simulated Metamorphosis Algorithm Approach. IEEE International Conference on Industrial Engineering and Operations Management, March 2015, Dubai

[6] 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, pp 833-838, 2003.

[7] C.D. Tarantilis, C.T. Kiranoudis, V.S.A. Vassiliadis, "A threshold accepting metaheuristic for the heterogeneous fixed fleet vehicle routing problem," European Journal of Operations Research, vol. 152, pp 148–158, 2004.

[8] M. Cheng, H.I. Ozaku, N. Kuwahara, K. Kogure, J. Ota, "Nursing care scheduling problem: Analysis of staffing levels," IEEE Proceedings of the 2007 International Conference on Robotics and Biomimetics, vol. 1, pp 1715-1719, December 2007.

[9] J. F. Bard, H.W. Purnomo, 'Preference scheduling for nurses using column generation', European Journal of Operational Research 164 (2): 510–534, 2005.

[10] 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.

[11] 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.

[12] 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.

[13] G. Tufte, "Metamorphosis and Artificial Development: An Abstract Approach to Functionality," In G. Kampis, I. Karsai, and E. Szathm´ary (Eds.): ECAL 2009, Part I, LNCS 5777, Springer-Verlag Berlin Heidelberg, pp 83–90, 2011.

[14] N. Musliu, "Heuristic methods for automated rotating workforce scheduling". International Journal of Computational Intelligence Research, vol. 2, no. 4, pp 309-326, 2006.

[15] P. Brucker, E. K. Burke, T. Curtois, "A shift sequence based approach for nurse scheduling and a new benchmark dataset". Journal of Heuristics, 16, pp 559 – 573, 2010.

[16] F. E. B. Otero, A.D. Masegosa, G. Terrazas, "Thematic issue on advances in nature inspired cooperative strategies for optimization", Memetic Computing, 6 (3), pp 147-148, 2014.

[17] J.W. Truman and L.M. Riddiford, "Endocrine insights into the evolution of metamorphosis in insects," The Annual Review of Entomolog, vol. 47, pp 467–500, 2002.

[18] M. Mutingi, C. Mbohwa, Simulated Metamorphosis - A Novel Optimizer, Lecture Notes in Engineering and Computer Science: Proceedings of The World Congress on Engineering and Computer Science 2014, WCECS 2014, 22-24 October, 2014, San Francisco, USA, pp 924-929.

[19] J. Gartner, N. Musliu, W. Slany. "Rota: A research project on algorithms for workforce scheduling and shift design optimization", Artificial Intelligence Communications, 14 (2), pp 83-92, 2001.

[20] M. Morz, N Musliu, "Genetic algorithm for rotating workforce scheduling problem", IEEE International Conference on Computational Cybernetics, pp 121-126, 2004

Michael Mutingi received a MEng in Industrial Engineering from the National University of Science & Technology, Zimbabwe in 2004. He recently completed his DEng in Engineering Management with the University of Johannesburg, South Africa. His research interests are in healthcare operations planning & control, metaheuristic optimization, fuzzy multi-criteria optimization, manufacturing systems, and business system dynamics. He has published more than 75 articles in international journals and conference proceedings, 12 book chapters, and 1 book.

Michael Mutingi has received best paper awards from International Association of Engineers (IAENG), Institute of Industrial Engineering & Operations Management (IIEOM), Industrial Engineering and Engineering Management (IEEM), and Planetary Scientific Research Centre (PSRC).

Charles Mbohwa is a Professor at the University of Johannesburg. He has a DEng from Tokyo Metropolitan Institute of Technology, MSc in operations management and manufacturing systems from the University of Nottingham, and a BSc (honours) in mechanical engineering from the University of Zimbabwe. He has been a senior lecturer in mechanical engineering at the University of Zimbabwe and a mechanical engineer at the National Railways of Zimbabwe. Prof C. Mbohwa has been a British Council Scholar, Japan Foundation Fellow, a Heiwa Nakajima Fellow, a Kubota Foundation Fellow and a Fulbright Fellow. His research interests are in operations management, engineering management, energy systems and sustainability assessment. He has published a book, several book chapters and more than 120 academic papers.