Nurse Scheduling Based on Branch- and- Bound Algorithm

Paramacutty Paramathevan, W.B.Daundasekera

[*] Department of Mathematics, Faculty of Science, Eastern University, Sri Lanka
[**] Department of Mathematics, Faculty of Science, University of Peradeniya, Sri Lanka

Abstract- The main objective of this study is to search for a scientific method to prepare a weekly schedule for a group of nursing officers working for a ward of a hospital. To achieve this objective, we propose an optimization method. First we develop an optimization model by formulating the objective and the constraints of the problem mathematically.

In the recent literatures, a wide variety of different methodologies and models have been developed pertaining to the problem instances. These include Mathematical Programming, Meta-Heuristic Methods and Constraint Satisfaction Techniques. Knowledge Based Techniques have been applied for solving nurse scheduling problem. Column generation was also been explored in nurse scheduling using Set Covering–Type models. Particular optimization model that we are interested in is a 0-1 Integer Linear Programming problem. We apply the Branch-and-Bound technique to solve the problem using the optimization software package LINGO. Finally, the solution to the optimization problem is converted to a regular nurse schedule. The methodology is illustrated by preparing a weekly schedule for a private hospital in Batticaloa, Sri Lanka.

***Index Terms*-** Nurse scheduling, Mathematical Programming, Meta-Heuristic Methods, Constraint Satisfaction Techniques, Column generation, Set Covering–Type models, Knowledge Based Techniques, Branch-and-Bound technique.

I. INTRODUCTION

Nurse scheduling is one of the most complex and tedious scheduling problems and is considered as NP-hard combinatorial problems [1] in the field of personal scheduling. Not only in Sri Lanka, but also in most of the developed countries, hospital management put much effort to prepare schedule for their own nursing officers while being perceived to be fair by the officers. This scheduling problem is extensively challenging due to different nurse requirements on different days and shifts, and also due to hospitals being in continuous operation unlike most of the other institutions.

Therefore, there is a need for an efficient approach to schedule nurses in order to save the time and resources in the scheduling process and to conduct hospital operations smoothly.

In the passed, a wide variety of methodologies and models have been developed to deal with different problem instances. Introduction to many of these methods can be found in [2]. A number of survey papers [3,4,5] give an overview of the area. The available techniques can roughly be classified into two main categories: exact algorithms and (meta) heuristics. Mathematical programming is the traditional exact method [6,7,8,9,10], which guarantees to find an optimal solution on nurse scheduling for every instance of a problem. However, computational difficulties exist with this approach due to the enormous size of the search spaces that are generated. To reduce complexity, some researchers have restricted the problem dimensions and developed simplified models. However, this leads to solutions that are not applicable to real hospital situations.

Metaheuristic approaches [11,12] were investigated with some success since the 1990's. Genetic and Memetic algorithms form an important class of metaheuristics that have been extensively applied in nurse scheduling [13,14,15,16,17]. A number of attempts have also been made by using other metaheuristics, such as simulated annealing [18], tabu search [19], variable neighborhood search [20,21,22] and estimation of distribution algorithms [23].

However, the major drawback of these metaheuristics is that they cannot provably produce optimal solutions nor can they provably reduce the search space. Also, they usually do not have well defined stopping criteria. Moreover, most nurse rostering problems are highly constrained problems so that the feasible regions of their solution space are disconnected (i.e. separated by the infeasible area). Metaheuristics generally have difficulty in dealing with this situation.

In the field of nurse rostering, some decomposition techniques have been investigated over recent passed years. Aickelin and Dowsland [24] developed a genetic algorithm with an indirect representation. Different heuristic decoders (i.e. decomposers) were employed to construct the schedule, taking care of coverage and nurses' preferences from different aspects. Ikegami and Niwa [25] grouped the constraints into shift constraints and nurse constraints. Based on this, the problems were decomposed into sub problems and solved repeatedly by tabu search. Brucker et al. [26] implemented decomposition by cyclically assigning predefined blocks of shifts to groups of nurses. The rest of the shifts were then assigned manually and the resulting schedule was improved by local search.

In this paper a mechanism based on 0-1 integer programming is proposed to prepare a roster for a group of nurses attached to a private hospital in Batticaloa, Sri Lanka for a period of one week. This proposed method is less complex compared to the existing methods since Branch-and-Bound is a common technique in the scheduling field. Further, this approach not just provides an idea to prepare a roster but includes some additional features which are not so easy to include when preparing a roster manually. These features are elaborated in the proposed mathematical model.

The rest of the paper is organized as follows. Following section describes research elaborations and the subsequent sections present the results and appendices. Finally, we conclude the paper with our conclusion.

II. RESEARCH ELABORATIONS

Prior to the optimization model, we introduce the notations and meanings of technical terms that are being used. Next we present the optimization model to solve the rostering problem.

1. Let $N = \{1,2,.......,\hat{n}\}$ be the set of nurses. For any $n \in N, y_n$ and z_n be the maximum number of working days and maximum number of night shifts to be assigned for the nurse n within the scheduling period respectively.
2. Let $T = \{1(early), 2(day), 3(late), 4(night)\}$ be the set of shift types of equivalent length of 8 hours each day.
3. Let $D = \{1,2,.......,\hat{d}\}$ be the days in the scheduling time horizon.
4. Let R_{d_t} be demand of number of nurses on day $d \in D$ of shift type $t \in T$.
5. Let k_1 be the maximum number of consecutive working days within the scheduling period.
6. Let k_2 be the maximum number of consecutive night shifts within the scheduling period.
7. Let $p_{n\_d\_t}$ be a penalty occurring if the nurse $n \in N$ is scheduled to work at shift type $t \in T$ on day $d \in D$.

Our task is to assign nurses for different shift types according to the demand on each day in the planning time horizon. It should satisfy the following standard requirements which are common to most of the nurse scheduling problems.

I. Demand at a shift type on a particular day should be satisfied.
II. A nurse should be assigned for at most one shift a day.
III. At most maximum number of working days should be assigned for a nurse in this time horizon.
IV. At most maximum number of night shifts should be assigned for a nurse in this time horizon.

In addition to these standard requirements following special requirements are request by the nursing officers:

V. No night shifts between two non-night shifts.
VI. Shift 1 or Shift 2 should not be assigned after night shift.
VII. Maximum number of consecutive working days is 3.
VIII. Maximum number of consecutive night shifts is 3.

Next we introduce the decision variables. Let

$$x_{n\_d\_t} = \begin{cases} 1, \text{if nurse } n \text{ is assigned for the shif type } t \text{ on day } d \\ 0, \text{otherwise.} \end{cases}$$

Introducing these decision variables, the optimization model which incorporates both common and special requirements can be presented as:

$$\text{Minimize} \sum_{n \in N} \sum_{d \in D} \sum_{t \in T} p_{n\_d\_t} \, x_{n\_d\_t}$$

Subject to

$$\sum_{n \in N} x_{n\_d\_t} = r_{d\_t}, \ \forall d \in D, t \in T$$

$$\sum_{t \in T} x_{n\_d\_t} \le 1, \quad \forall n \in N, d \in D$$

$$\sum_{d \in D} \sum_{t \in T} x_{n\_d\_t} \le y_n, \quad \forall n \in N$$

$$\sum_{d \in D} x_{n\_d\_4} \le z_n, \quad \forall n \in N$$

$$x_{n\_d-1\_4} - x_{n\_d\_4} + x_{n\_d+1\_4} \ge 0, \ \forall n \in N, d \in \{2,3,4,5,6\}$$

$$x_{n\_d-1\_4} + x_{n\_d\_1} + x_{n\_d\_2} \le 1, \ \forall n \in N, d \in D$$

$$\sum_{d=l}^{k_1+l} x_{n\_d\_t} \le k_1, \quad \forall n \in N, l \in \{1,2,3,4\}$$

$$\sum_{d=l}^{k_2+l} x_{n\_d\_4} \le k_2, \quad \forall n \in N, l \in \{1,2,3,4\}$$

The mathematical model described above is for preparing a weekly schedule for wards up to 20 nurses at a private hospital at Batticaloa, Sri Lanka. These schedules must respect working contracts and meet the demand for a given number of nurses on each shift type, while being perceived to be fair by the nurses themselves.

The objective function of our optimization model is considered as a cost function, where cost is interpreted as penalty and penalty is defined based on the desirability of a nurse to work at a shift type on a day. Therefore, our attempt is to minimize the penalty subject to the given constraints. These penalties appeared as coefficients in the objective function.

Hospital administration requires 3 nurses at each shift so the demand R_{d_t} is uniform and is equal to 3 for all shifts throughout the time horizon. Also, maximum number of working days and maximum number of night shifts for a nurse are uniform and are assumed to be 5 and 4 respectively. That is, $y_n = 5$ and $z_n = 4$ for each nurse. Further, both the maximum number of consecutive working days, k_1 and maximum number of consecutive night shifts, k_2 within the scheduling period are required to be 3.

III. RESULTS AND DISCUSSION

This proposed model is applied to a private hospital with 20 nurses. By using the information obtained from the hospital administration as well as nursing officers. A 0-1 integer linear programming model is formulated, which consists of 560, 0-1 binary decision variables and 409 constraints.

Computations are carried on a COMPACT laptop, Intel Celeron1.66GHz processor and 1 GB RAM.

The result obtained by solving the 0-1 Integer Linear Programming problem is presented in the following table. Here, numbers represent the shift type for corresponding days:

N.No D.No	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**
1	1	2	2		4	4	1	4					3	3	2	3				1
2	4			3	2	1		2		4			1	2	1			3	4	2
3	1	2		1		4		3	3					2	4		4	1	2	3
4		4	3	3	2	2	3		2						1	4	1	4	1	
5	1		1	1	4	4	2		4			3			3	2	3		2	
6		4	3		2		2		1			4	4	3		1	2	3	1	
7	3		1				1	2	2			1	3	4		4	4	2		3

First row, N.No represents nurse number and the first column, D.No represents day number.

From the above table it can be observed that this hospital runs with excess nurses. So we decided to reduce the number of nurses one by one and solve the model repeatedly for 19, 18,17 and 16 nurses. Finally, we come to a conclusion that 17 is the optimal number of nurses to run this hospital subject to the above rules or constraints with the given penalty values. Further, this can be 16, if the nursing officers agree to work up to 6 days in one week. Solution for the model with number of nurses 17 and 16 are given in the table-I and table-II in Appendix-I.

IV. CONCLUSION

We have developed an optimization model for nurse scheduling. We have also proposed a method of solution to solve the model. This method will guide to prepare a nurse schedule more efficiently, accurately and quickly for any ward of a hospital which satisfies the nurses' preferences and regulations provided by the hospital administrators.

In our model, all nurses are considered as same grade. But there are hospitals nurses are in different grades. According to their qualifications/experience demand for different grades may vary. One of our future works in this study will include the incorporation of grades in the formulation. This will produce more realistic results. Next aspect will be the time horizon. It is always advisable to have a long time horizon because nurses can plan their personal work in advance if they know their roster. Also preparing a weekly schedule is a time consuming activity. So in the future we will consider monthly roster with adequate number of nurses.

APPENDICES

APPENDIX-I

Table-I shows the solution for the model for 17 nurses. The Integer Linear Programming problem contains 476 binary variables and 505 constraints.

Table-I

N.No D.No	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**
1	2	1	2		4		1	4	1	4			3	3	2	3	
2		2	1	2	3	1				4		4	2	3	2	1	4
3	1	2		1		3	4	1	3			4	3	2			4
4	3		3	3	2	1	4	1		2					1	2	4
5	2	4	1		1		3	3	4			3		1	2	2	
6		4	3	1	2	3			4	3		1	4	2	1		2
7	3			4		2	1	2	3	2		1	4			3	4

Here we can see that nurse number 11 is off in this week since he/she gave large penalty values compared with other nurses.

Table-II shows the solution for the problem with 16 nurses. This problem contains 448 variables and 477 constraints.

Table-II

N.No D.No	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**1**	**15**	**16**
1	1	2	2		4		1	4	1	4			3	3	2	3
2			3	2	3	1			4	4	1	4	2	3	2	1
3	1	2		1		3	4	3	4		2	4	3	2		1
4	3	2	3	3	4	1	4	1		2	2	4			1	
5	1	3					3	3	4	2			4	1	2	2
6			1	1	4	3			4	3	2	1	4	2	2	1
7	2	3	4	4		2	1	2	3		1	1	3	4		4

Appendix-II

This appendix gives the C++ code to generate 0-1 integer Programming Problem.

```
#include<iostream.h>
#include<conio.h>
#include<fstream.h>
int const num_of_nurses=20;  //20 nurse problem
int const demand=3;
int const num_of_shifts=4;
int const time_horizone=7;
int const max_num_of_dayshifts=5;
int const max_num_of_nightshifts=4;
int const max_num_of_consecutive_night_shifts=3;
int const max_num_of_consecutive_working_days=3;
//ofstream testfile("C:\\Param\\objective.txt");
int i,j,k,r;
int main()
{
//* consraint1 start
ofstream testfile3("C:\\Param\\cons1.txt");
for( j=1;j<=time_horizone;j++)
{       for( k=1;k<=num_of_shifts;k++)
        {       for ( i=1;i<=num_of_nurses;i++)
                {       if (i==num_of_nurses)
                                {
        cout<<"x"<<i<<"_"<<j<<"_"<<k;

        testfile3<<"x"<<i<<"_"<<j<<"_"<<k;
                                }
                        else
                        {

        cout<<"x"<<i<<"_"<<j<<"_"<<k<<"+";

        testfile3<<"x"<<i<<"_"<<j<<"_"<<k<<"+";
                        }
                }
                cout<<"="<<demand<<";"<<endl<<endl;
                testfile3<<"="<<demand<<";"<<endl<<endl;
        }
cout<<endl<<endl;
testfile3<<endl<<endl;
}
//end of cons1*/
/*consraint2 start
ofstream testfile3("C:\\Param\\cons2.txt");
for (i=1;i<=num_of_nurses;i++)
{       for( j=1;j<=time_horizone;j++)
        {       for( k=1;k<=num_of_shifts;k++)
                {       if (k==num_of_shifts)
                        {
        cout<<"x"<<i<<"_"<<j<<"_"<<k;

        testfile3<<"x"<<i<<"_"<<j<<"_"<<k;
                        }
                else
                        {

        cout<<"x"<<i<<"_"<<j<<"_"<<k<<"+";

        testfile3<<"x"<<i<<"_"<<j<<"_"<<k<<"+";
                        }
                }
        cout<<"<=1"<<";"<<endl;
        testfile3<<"<=1"<<";"<<endl;
        cout<<endl;
        testfile3<<endl;
        }
cout<<endl<<endl;
testfile3<<endl<<endl;
}
end of consraint2......*/
/*constraint3 start
ofstream testfile3("C:\\Param\\cons3.txt");
for (i=1;i<=num_of_nurses;i++)
{       for( j=1;j<=time_horizone;j++)
        {       for( k=1;k<=num_of_shifts;k++)
                {       if (j==time_horizone && k==4)
                        {
        cout<<"x"<<i<<"_"<<j<<"_"<<k;

        testfile3<<"x"<<i<<"_"<<j<<"_"<<k;
                        }
                else
                        {
```

```
        cout<<"x"<<i<<"_"<<j<<"_"<<k<<"+";

        testfile3<<"x"<<i<<"_"<<j<<"_"<<k<<"+";
                        }

                }
        }
        cout<<"<="<<max_num_of_dayshifts<<";"<<endl<<en
dl;
        testfile3<<"<="<<max_num_of_dayshifts<<";"<<endl<
<endl;
}
end of consraint3*/
/*constraint4 start
ofstream testfile3("C:\\Param\\cons4.txt");
for (i=1;i<=num_of_nurses;i++)
{       for( j=1;j<=time_horizone;j++)
        {       if(j!=time_horizone)
                {       cout<<"x"<<i<<"_"<<j<<"_4"<<"+";

        testfile3<<"x"<<i<<"_"<<j<<"_4"<<"+";
                }
                else {
                        cout<<"x"<<i<<"_"<<j<<"_4";
                        testfile3<<"x"<<i<<"_"<<j<<"_4";
                  }
        }

        cout<<"<="<<max_num_of_nightshifts<<";"<<endl;
        testfile3<<"<="<<max_num_of_nightshifts<<";"<<endl
;
}
end of consraint4*/
/*constraint5 start
ofstream testfile3("C:\\Param\\cons5.txt");
for (i=1;i<=num_of_nurses;i++)
{       for( j=2;j<time_horizone;j++)
{       cout<<"x"<<i<<"_"<<j-1<<"_"<<"4"<<"-
"<<"x"<<i<<"_"<<j<<"_"<<"4"<<"+"<<"x"<<i<<"_"<<j+1<<"
_"<<"4"<<">=0;"<<endl;
testfile3<<"x"<<i<<"_"<<j-1<<"_"<<"4"<<"-
"<<"x"<<i<<"_"<<j<<"_"<<"4"<<"+"<<"x"<<i<<"_"<<j+1<<"
_"<<"4"<<">=0;"<<endl;
        }
        cout<<endl;
        testfile3<<endl;
}
end of consraint5*/
/*constraint6 start
ofstream testfile3("C:\\Param\\cons6.txt");
for (i=1;i<=num_of_nurses;i++)
{       for( j=2;j<=time_horizone;j++)
{       cout<<"x"<<i<<"_"<<j-
1<<"_"<<"4"<<"+"<<"x"<<i<<"_"<<j<<"_"<<"1"<<"+"<<"x"<
<i<<"_"<<j<<"_"<<"2"<<"<=1;"<<endl;
testfile3<<"x"<<i<<"_"<<j-
1<<"_"<<"4"<<"+"<<"x"<<i<<"_"<<j<<"_"<<"1"<<"+"<<"x"<
<i<<"_"<<j<<"_"<<"2"<<"<=1;"<<endl;
        }
        cout<<endl;
        testfile3<<endl;
}
end of consraint6*/
//no consercutive night shifts
/*constraint7 start
ofstream testfile3("C:\\Param\\cons7.txt");
for (i=1;i<=num_of_nurses;i++)
{       for             (r=1;r<=(time_horizone-
max_num_of_consecutive_night_shifts);r++)
        {
        for(j=r;j<=(r+max_num_of_consecutive_night_shifts);j
++)
                {       if
(j<r+max_num_of_consecutive_night_shifts)
                {       cout<<"x"<<i<<"_"<<j<<"_4"<<"+";

        testfile3<<"x"<<i<<"_"<<j<<"_4"<<"+";
                }
                else
                        {
        cout<<"x"<<i<<"_"<<j<<"_4";

        testfile3<<"x"<<i<<"_"<<j<<"_4";
                        }
                }

        cout<<"<="<<max_num_of_consecutive_night_shifts<
<";"<<endl;

        testfile3<<"<="<<max_num_of_consecutive_night_shif
ts<<";"<<endl;
        }
}
cout<<endl;
testfile3<<endl;
end of consraint7*/

/*constraint8 start
ofstream testfile3("C:\\Param\\cons8.txt");
for (i=1;i<=num_of_nurses;i++)
{       for             (r=1;r<=(time_horizone-
max_num_of_consecutive_working_days);r++)
        {
        for(j=r;j<=r+max_num_of_consecutive_working_days;j
++)
                for( k=1;k<=num_of_shifts;k++)
        {       if                                      (
j==r+max_num_of_consecutive_working_days               &&
k==num_of_shifts )
{
                        cout<<"x"<<i<<"_"<<j<<"_"<<k;
                        testfile3<<"x"<<i<<"_"<<j<<"_"<<k;
                        }
                else
                {
        cout<<"x"<<i<<"_"<<j<<"_"<<k<<"+";
```

```
        testfile3<<"x"<<i<<"_"<<j<<"_"<<k<<"+";
                }
                }
//testfile<<"x"<<i<<"_"<<j<<"_"<<k<<"+")}
        cout<<"<="<<max_num_of_consecutive_working_days
<<";"<<endl<<endl;
        testfile3<<"<="<<max_num_of_consecutive_working_
days<<";"<<endl<<endl;
        }
}
cout<<endl<<endl;
testfile3<<endl<<endl;
end of constraint8 */

//binormial variables
/*binomial variables start
ofstream testfile3("C:\\Param\\binomial.txt");
for (i=1;i<=num_of_nurses;i++)
{       for( j=1;j<=time_horizone;j++)
        {       for( k=1;k<=num_of_shifts;k++)
                {
        cout<<"@bin(x"<<i<<"_"<<j<<"_"<<k<<");"<<endl;

        testfile3<<"@bin(x"<<i<<"_"<<j<<"_"<<k<<");"<<end
l;
                }
        }
}
cout<<"end";
testfile3<<"end";
end of binomial variables */
getch();
return 0;
}
```

REFERENCES

[1] R.M. Karp, Reducibility among combinatorial problems, in: R.E. Miller, J.W. Thatcher (Eds.), Complexity of Computer Computations, Plenum Press, New York, 1972, pp.85–103.

[2] E.K. Burke, G. Kendall (Eds.), Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques, Springer, 2005.

[3] E.K. Burke, P. De Causmaecker, G. Vanden Berghe, H. Landeghem, The state of the art of nurse rostering, Journal of Scheduling 7 (6) (2004) 441–499.

[4] B. Cheang, H. Li, A. Lim, B. Rodrigues, Nurse rostering problems – A bibliographic survey, European Journal of Operational Research 151 (2003) 447–460.

[5] D. Sitompul, S. Randhawa, Nurse scheduling models: A state-of-the-art review, Journal of the Society of Health Systems 2 (1990) 62–72.

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

[7] N. Beaumont, Scheduling staff using mixed integer programming, European Journal of Operational Research 98 (1997) 473–484.

[8] Berrada, J.A. Ferland, P. Michelon, A multi-objective approach to nurse scheduling with both hard and soft constraints, Socio-Economic Planning Science 30 (1996) 183–193.

[9] A. Ikegami, A. Niwa, A subproblem-centric model and approach to the nurse scheduling problem, Mathematical Programming 97 (2003) 517–541.

[10] M. Warner, J. Prawda, A mathematical programming model for scheduling nursing personnel in a hospital, Management Science 19 (1972) 411–422.

[11] L.D. Smith, D. Bird, A. Wiggins, A computerized system to schedule nurses that recognizes staff preferences, Hospital and Health Service Administration (1979) 19–35.

[12] R. Blau, Multishift personnel scheduling with a microcomputer, Personnel Administrator 20 (1985) 43–58.

[13] U. Aickelin, K. Dowsland, Exploiting problem structure in a genetic algorithm approach to a nurse rostering problem, Journal of Scheduling 3 (2000) 139–153.

[14] U. Aickelin, K. Dowsland, An indirect genetic algorithm for a nurse scheduling problem, Computers and Operations Research 31 (2004) 761–778.

[15] E.K. Burke, P. Cowling, P. De Causmaecker, G. Vanden Berghe, A memetic approach to the nurse rostering problem, Applied Intelligence 15 (2001) 199–214.

[16] F.F. Easton, N. Mansour, A distributed genetic algorithm for deterministic and stochastic labor scheduling problems, European Journal of Operational Research 118(1999) 505–523.

[17] H. Kawanaka, K. Yamamoto, T. Yoshikawa, T. Shinigi, S. Tsuruoka, Genetic algorithm with the constraints for nurse scheduling problem, in: Proceedings of Congress on Evolutionary Computation (CEC), 2001, pp. 1123–1130.

[18] .M.J. Brusco, L.W. Jacobs, A simulated annealing approach to the cyclic staff-scheduling problem, Naval Research Logistics 40 (1993) 69–84.

[19] E.K. Burke, P. De Causmaecker, G. Vanden Berghe, A hybrid tabu search algorithm for the nurse rostering problem, Lecture Notes in Artificial Intelligence, 1585, Springer,1999. pp. 187–194.

[20] E.K. Burke, T. Curtis, G. Post, R. Qu, B. Veltman, A hybrid heuristic ordering and variable neighbourhood search for the nurse rostering problem, European Journal of Operational Research 188 (2008) 330–341.

[21] E.K. Burke, P. De Causmaecker, S. Petrovic, G. Vanden Berghe, Variable neighborhood search for nurse rostering problems, in: M.G.C.

Resende, J.P. De Sousa (Eds.), Metaheuristics: Computer Decision-Making (Combinatorial Optimization Book Series), Kluwer, 2004, pp. 153–172 (Chapter 7).

[22] E.K.Burke, Jingpeng Li, Rong Qu, A hybrid model of Integer Programming and Variable Neighbourhood Search for Highly-Constrained Nurse Restering Problems, European Journal of Operational Research 203 (2010) 484–493.

[23] U. Aickelin, J. Li, An estimation of distribution algorithm for nurse scheduling, Annals of Operations Research 155 (2007) 289–309.

[24] U. Aickelin, K. Dowsland, An indirect genetic algorithm for a nurse scheduling problem, Computers and Operations Research 31 (2004) 761–778.

[25] P. Hansen, N. Mladenovic´ , Variable neighbourhood search: Principles and applications, European Journal of Operational Research 130 (1999) 449–467.

[26] P. Brucker, R. Qu, E.K. Burke, G. Post, A decomposition, construction and post-processing approach for a specific nurse rostering problem, in: Proceedings of the Second Multidisciplinary International Scheduling: Theory and Applications Conference (MISTA), 2005, pp. 397–406.

AUTHORS

First Author – P.Paramadevan, B.Sc,M.Sc, Department of Mathematics, Eastern University, Sri Lanka, Vantharumoolai, paramdeve@yahoo.com.au, param@esn.ac.lk

Second Author – W.B.Daundasekera,B.Sc,M.Sc,Ph.D, Associate Professor, Department of Mathematics, University of Peradeniya, Sri Lanka, wbd@pdn.ac.lk

Correspondence Author – W.B.Daundasekera,B.Sc,M.Sc,Ph.D, Associate Professor, Department of Mathematics, University of Peradeniya, Sri Lanka, wbd@pdn.ac.lk