Research Paper

Artificial intelligence and end user tools to develop a nurse duty roster scheduling system

Franklin Leung [a, *], Yee-Chun Lau [b], Martin Law [c], Shih-Kien Djeng [c]

[a] Science and Technology Section, School of Continuing and Professional Education, City University of Hong Kong, HKSAR, China
[b] Department of Rehabilitation, Kowloon Hospital, HKSAR, China
[c] Proton Therapy Pte Ltd, 1 Biopolis Drive, Singapore

ABSTRACT

Objectives: A nurse duty roster is usually prepared monthly in a hospital ward. It is common for nurses to make duty shift requests prior to scheduling. A ward manager normally spends more than a working day to manually prepare and subsequently to optimally adjust the schedule upon staff requests and hospital policies. This study aimed to develop an automatic nurse roster scheduling system with the use of open-source operational research tools by taking into account the hospital standards and the constraints from nurses.
Methods: Artificial intelligence and end user tools operational research tools were used to develop the code for the nurse duty roster scheduling system. To compare with previous research on various heuristics in employee scheduling, the current system was developed on open architecture and adopted with real shift duty requirements in a hospital ward.
Results: The schedule can be generated within 1 min under both hard and soft constraint optimization. All hard constraints are fulfilled and most nurse soft constraints could be met. Compared with those schedules prepared manually, the computer-generated schedules were more optimally adjusted as real time interaction among nurses and management personnel. The generated schedules were flexible to cope with daily and hourly duty changes by redeploying ward staff in order to maintain safe staffing levels.
Conclusions: An economical but yet highly efficient and user friendly solution to nurse roster scheduling system has been developed and adopted using open-source operational research methodology. The open-source platform is found to perform satisfactorily in scheduling application. The system can be implemented to different wards in hospitals and be regularly updated with new hospital polices and nurse manpower by hospital information personnel with training in artificial intelligence.

What is known?

- Nurse scheduling is well known to be modeled as a constraint optimization problem. Previous studies focused on applying appropriate algorithms and developing systems using available software libraries to address hypothetical scenarios.
- Deployment of such systems to real hospital environment has been attempted but their clinical applications are often limited due to their inherent inflexibility and the high cost involved with commercial package maintenance.

What is new?

- We have developed a prototype system using open-source with libraries as well as economical software platforms and then tested the system with real scheduling data. The output quality and the system performance are found to be satisfactory, therefore saving nurses' time to direct to clinical works.
- It demonstrates that open-source development on open-source platforms is a feasible direction as it is cost effective

* Corresponding author.
E-mail addresses: leungfranklin33@gmail.com (F. Leung), lyc692@ha.org.hk (Y.-C. Lau), ynbugy_law@yahoo.com (M. Law), skjeng@gmail.com (S.-K. Djeng).
Peer review under responsibility of Chinese Nursing Association.

and efficient. It is an open platform so that different users of diverse backgrounds can collaborate together to create a generic and powerful solution.

1. Introduction

Nurse scheduling planning (NSP) is the assignment of a number of nurses to a number of shifts in order to satisfy patient service demands from the hospital. The objectives of NSP are the minimization of the overall hospital resources, and the maximization of nurses' preferences while taking into consideration the governmental policies and hospital standards [1]. Usually, nursing manager spends a substantial amount of time developing rosters for many staff requests, and for ad hoc changes to current duty rosters. Because of the tedious and time consuming manual scheduling, the NSP has attracted much research attention [1,2]. Although there are many commercially available nurse roster systems, these systems use proprietary algorithms or platforms leading to expensive license fee and extra maintenance cost. Knowledge and experience sharing among different commercial systems are also limited [2]. Ongoing research studies are mostly based on heuristic algorithms and conventional linear programming optimization techniques which are mainly presented for homogeneous personnel shift scheduling [2,3]. Indeed, NSP can be performed by using the recently developed open-source artificial intelligence (AI) constraint optimization modules, resulting in more flexibility and cost effectiveness in maintaining the system to generate roster schedule [4]. As it is open sourced, source codes developed for different applications are usually contributed for knowledge and experience sharing, in that various software experts can collaborate on this open-source platform to develop a system of common interest in this NSP. The current work reports a solution based on such approach to a practicing nursing roster planning scenario in Hong Kong, as currently there is no automated tool to run the schedule systematically during roster planning and its optimization.

In Hong Kong, China, all public hospitals are managed by the Hong Kong Hospital Authority [5]. Ward nurses and supporting staff are managed by the ward manager. Each ward may comprise 30 to 80 nurses, depending upon the nature of the clinical services provided. There are morning, afternoon and night shifts. Each ward designates the required number of nurses and a senior nurse as the supervisor on duty at each shift. The ward manager has to devise the roster in advance for the following month. It happens frequently that nurses may make different requests for a particular shift on some designated days. Although some of their requests need not to be satisfied, the ward manager usually tries the best to accommodate such requests in order to maintain work-life balance for staff members and to cope with the workload in hospitals. In addition, some constraints have to be satisfied such as there should be at least one day off within 10 working days. Fairness in rostering has also to be taken into consideration. As nurses do not receive additional compensation for duty working on public holidays, many nurses may apply for day off on public holidays. Therefore, fairness in rostering is important, or otherwise, the same nurse may receive disproportional share of a day off during public holidays. In general, ward managers in healthcare sectors have to spend significant effort and time in the order of days to devise the NSP schedule for staff members manually and then optimize the schedule repetitively. It is also desirable to apply interactive rostering systems to cope with unpredictable daily and hourly duty changes by redeploying ward staff in order to maintain staffing levels and to save nurses' time to direct to clinical work.

2. Methods

2.1. Modeling and programming for NSP

NSP can be modeled as a constraint satisfaction problem. The key input involves the variables, their domains, and the constraints that restrict how these variables may be assigned values from their domains. In our rostering engine, each variable represents the unknown shift that a nurse should work on a particular day. The domain consists of all the shifts defined for the group of staff that the roster is generated for.

The NSP system is based on economical and open-source operational research tools made available by Google® OR-tools [6], which is suitable for operation research optimization and can be employed to tackle problems in vehicle routing, flows, and constraint programming such as scheduling. Software engineers can model their problems in the programming language of his/her own choice (for example of C, Python and Java) and to solve the problems with commercial solvers such as Gurobi or CPLEX [7,8], or open-source solvers such as Google's GLOP and CP-SAT [6]. The GLOP is a linear optimizer to obtain the optimal value of a linear objective function expressed as a set of linear inequalities as constraints. The CP-SAT solver is mainly used for constraint satisfaction and hence it is selected for addressing the NSP problem here (Fig. 1). Python is adopted as the programming language for this project [9].

2.2. Assessing the performance of the system in a ward

In order to assess the performance of the system, initial data collection was performed within the year 2019 in the ward managed by one of the authors (YCL). Since the data collection did not involve patients, no approval from the institute was needed. The system ran on a desktop computer. It input an Excel spreadsheet detailing all nurse shift requests and outputs the results in an Excel spreadsheet containing the generated roster schedule (Fig. 1). As the NSP is modeled as a constraint optimization problem, our rostering constraints are classified either as hard or soft. Hard constraints are those constraints that must be satisfied or otherwise the entire work schedule will be deemed invalid. As an example of a hard constraint, each nurse could only work on one shift each day. Soft constraints were shift requests made by nurses, of which nurses might request for day off or for some preferred schedule. It is desirable that these soft constraints are fulfilled. However, failure to satisfy such soft constraints did not make the schedule invalid. The overview of the key constraints (Fig. 1) is described as follows.

2.2.1. Hard constraints

2.2.1.1. Manpower demand. This constraint defines manpower requirement for a specific ward. For an example as in our illustrated case, there were nine nurses working for day shift and two nurses working for night shift. As the ward was divided into areas for male and female patients, at least three nurses should work in each area. A nurse of higher seniority or of higher rank must be present in the morning and afternoon shift.

2.2.1.2. Shift allocation. This constraint describes the frequency that a staff may be assigned to duties during a particular period. For example, each nurse member should have at least one day off each week; each nurse can only work for one shift (either morning, afternoon or night) each day; nurses were on approved annual leave.

2.2.1.3. Shift sequence pattern. This constraint, based on hospital policy, defines the pattern of shifts assigned on consecutive days to
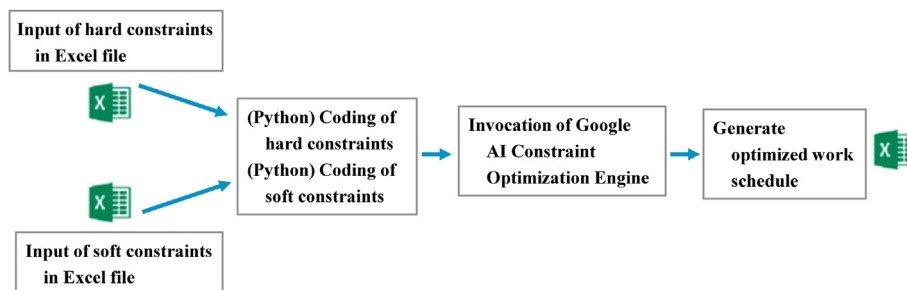
Fig. 1. System architecture used in the current study.

nurses. For example, a nurse member cannot work on a morning shift after a night shift.

2.2.2. Soft constraints

2.2.2.1. Shift sequence pattern. This soft constraint, based on the routine management in nurse shift in the ward, defines the pattern of shifts assigned on consecutive days to individual nurses. There are patterns generally not in preference to nurses but it is allowed to have such patterns as schedule sequence. For example, the following patterns are not preferred: ON, AAA, PPP, POP, POA, POOP (A: morning duty, P: afternoon duty, N: night duty, O: day off). Nurses do not prefer AAA as they have to get up early in the morning for three consecutive days; PPP are not preferred as nurses cannot have supper with their family for three consecutive days; other patterns are not preferred as nurses generally like to have longer rest time between previous duty and next duty when rest day is involved. However, these shift duties, though not preferred, may still be scheduled in consideration of tight manpower resource.

2.2.2.2. Personal request shift. These are shift requests made by nurses, in which nurses may request for day off or for some preferred schedule. It is desirable that these soft constraints are fulfilled.

2.2.3. Input coding to allocate nurses shifts

From the Excel input file, requests are transformed by the code into constraints as input to the optimization solver module (Fig. 1). Soft constraint requests are transformed (staff, date, shift, penalty weight) and hard constraint requests are transformed into (staff, date, assigned shift). Additional constraints are coded and input into the optimization solver module. An example to code the constraints is presented (Appendix A), in which there are a total of nine nurses assumed for each shift duty (represented by *s*) with at least three nurses working in one side of the ward and at least another three nurses working in the other side. Each nurse (represented by *e*) can be either working in one shift each day or off duty.

2.2.4. Output coding to optimally allocate nurses' shifts

All nurse requests, as expressed in an Excel spreadsheet, are inputted into the code. Then a model is created by maximizing the alignment of the generated shift schedule with the nurse requests (Appendix B). The solution is obtained by invoking the optimization engine provided by Google® OR-tools (OR-Tools) [6]. and is output to an Excel spreadsheet.

3. Results

Upon completion of the input Excel spreadsheet tabulating the nurse shift requests the code can then be run and an output schedule is generated within 1 min with GPU (about 2 min without

GPU, GPU: graphics processing unit) with consideration to all hard and soft constraints. Fig. 2 shows that most of the nurses' requirements are fulfilled, except for nurse 6 on 8 May and for nurse 22 on 10 May. Both nurses were assigned day-off by the computer code instead of requesting annual leave while all hard constraints were still fulfilled.

In order to assess the correctness and effectiveness of the generated output schedule, the computer-generated schedule has been compared with the schedule prepared manually by the ward manager. It has been observed that the computer-generated schedule is even more optimally adjusted than the one manually prepared, in addition to the time saving from the day as in the manual planning to minutes as in using the code for the schedule generation. The generated schedule is particularly useful to cope with unpredictable daily and hourly duty changes by redeploying ward staff in order to maintain safe staffing levels as the interactive nature of the code to generate an updated schedule.

4. Discussion

In general, computer-generated rostering systems ensure that ward staff are appropriately allocated to provide high quality and efficient health service [10]. To address the significant human resources allocated by the ward manager in NSP, a staff rostering system was developed by the Hong Kong Hospital Authority using constraint programming techniques at the Year 2000 to automate the nurse rostering task [11]. However, this system has gradually faded out to become an administrative system for roster filing and the system is now mainly used by a number of hospitals just as an administrative system to save roster records into databases rather than to generate roster schedule for medical personnel. The root cause, as probably the same for earlier versions of similar systems in other healthcare sectors, is believed that the nature of work environment in patient wards is diversified among different specialties in hospitals. The modification of the logic for a centralized roster planning system to accommodate different requirements is time consuming and may not be cost effective. The scheduling rules and hospital policies may also evolve along with time for regular revision of the system. This indicates that successful implementing computer rostering system for workforce groups requires constant update to the system in terms of computing technology and of hospital policies. With the advancement of computer hardware and development of user-friendly open source software [6], readily interactive results from revised NSP are advantages from our study. Even unpredictable ward staff manpower relocation can be made by simply updating the input Excel table to the calculating system (Fig. 1), thus saving lots of nurses' time in schedule preparation (from day to minute) and almost interactively solving urgent manpower allocation problems in patient wards [10]. All of these will be value added to enhance the routine patient services.

A:

Staff number	Mon 6-May	Tue 7-May	Wed 8-May	Thur 9-May	Fri 10-May	Sat 11-May	Sunday 12-May	Holiday 13-May	Tue 14-May	Wed 15-May	Thur 16-May	Fri 17-May	Sat 18-May	Sunday 19-May	Mon 20-May	Tue 21-May
1		A	N	O						O					A	N
2						A	N	O								
3			O			A	O		D5	A	N	O				P
4				A	O			A		N	O					
5			N	D5	O	O			O	O				A	N	O
6		N	A	O							A			N	O	
7	A					O	A	N		A		O			A	
8			A	O			A	N					O	O	AL	AL
9				N	O								O	AL	AL	
10	N	O	A				O				A	A				
11											N	A				
12							O	O								
13							N								A	N
14					N		O	AL	AL	AL	AL	AL	DO	O		
15		O									N		O			
16				N		A	O	AL	AL	AL	AL	AL	DO	O		AL
17	AL	AL	AL	AL	AL	DO	O		AL	N	O					
18	N			O		A					N		O			
19	O		A				A	N	O	A					AL	AL
20		A	N				O				A	P	A	N		
21			SD	A	N	O				O	SD					O
22			A	N	A	O			AL			O		A	N	O
23		A		D5	A	N		AL								A
24	AL	AL	AL	AL	AL	DO	O	AL	AL	AL	AL	AL	DO	O		A
25	AL	AL	AL	AL	AL	DO		AL	AL	N				O		A
26			A					A	N					O		
27							AL	AL	AL	AL	AL	AL	DO	O		
28					P	A	N									P
29	A			A	N	P	A	O			P	A	N	O		
30	A	N	O						A	N	O	A				
31						A	N								A	N

B:

Staff number	Mon 6-May	Tue 7-May	Wed 8-May	Thur 9-May	Fri 10-May	Sat 11-May	Sunday 12-May	Holiday 13-May	Tue 14-May	Wed 15-May	Thur 16-May	Fri 17-May	Sat 18-May	Sunday 19-May	Mon 20-May	Tue 21-May
1	P	A	O	O	O	O	P	A	P	O	O	O	A	P	A	O
2	A	P	P	A	A	N	O	P	P	P	O	O	O	O	O	O
3	O	O	O	P	P	A	O	O	D5	A	N	O	P	A	P	P
4	O	P	A	P	A	O	P	A	P	A	N	O	A	A	P	A
5	P	A	N	D5	O	O	O	P	O	O	O	A	P	A	N	O
6	A	A	O	O	A	P	P	A	A	P	A	A	N	O	A	P
7	A	A	P	P	A	O	A	N	P	A	A	O	A	P	A	P
8	P	P	A	O	O	P	P	A	N	P	A	P	O	O	AL	AL
9	A	P	P	A	N	O	P	A	A	P	A	P	O	O	AL	AL
10	N	O	A	A	P	A	O	A	P	P	A	A	N	A	AL	AL
11	O	O	P	P	A	A	P	A	A	N	P	A	P	A	O	P
12	O	P	A	P	A	O	O	A	A	A	P	P	A	A	O	A
13	O	O	P	A	O	A	N	P	P	A	P	A	O	A	A	N
14	A	A	P	P	A	N	O	AL	AL	AL	AL	AL	DO	O	P	A
15	O	A	A	O	A	P	A	O	P	A	A	P	A	A	P	A
16	P	P	A	N	P	A	O	AL	AL	AL	AL	AL	DO	O	P	AL
17	AL	AL	AL	AL	AL	DO	O	P	AL	N	O	P	P	A	P	A
18	N	O	P	A	O	O	A	P	A	P	A	N	O	O	O	P
19	O	O	A	A	P	A	P	A	A	N	O	A	A	P	AL	AL
20	P	A	N	O	P	A	O	P	A	O	P	P	N	O	P	P
21	P	P	O	SD	A	N	O	P	A	O	SD	P	P	A	A	O
22	P	A	A	N	O	A	O	A	AL	A	A	O	P	A	N	O
23	P	A	P	O	D5	A	N	P	AL	O	A	P	P	A	P	A
24	AL	AL	AL	AL	AL	DO	O	AL	AL	AL	AL	AL	DO	O	P	A
25	AL	AL	AL	AL	AL	DO	O	P	AL	N	O	P	O	O	O	A
26	A	O	P	A	A	P	P	A	N	O	O	P	A	O	P	A
27	P	P	A	P	P	A	P	AL	AL	AL	AL	AL	DO	O	O	O
28	O	A	A	O	P	A	P	A	N	P	A	O	A	P	P	A
29	A	P	A	A	N	P	A	O	A	P	A	N	O	A	A	P
30	A	N	O	P	P	A	P	A	P	A	N	O	A	P	P	A
31	A	O	O	O	O	O	A	N	P	P	A	A	O	P	A	N

Fig. 2. Tabulation of requests from nurses and of output to the nurse duty schedule.
Fig. 2-A: An example as a summary detailing each nurse request expressed in an Excel spreadsheet as an input file to the code, showing specific requests on annual leave around 12/13 May (long weekend). Those blank entries were of no specific request from the nurses.
Fig. 2-B: Results of the computer-generated roster schedule, showing that most nurses' requests have been fulfilled including the two boxed entries (request of nurse 6 on 8 May has been changed from A to O, and request of nurse 22 on 10 May has been changed from A to O). In particular, nurse 6 made an unusual request for a morning shift after night shift. However, morning shift is usually not granted after night shift work. The computer system changed the morning shift to off-duty in correspondence with the constraint defined. Nurses usually will not object to off-duty as an alternative arrangement.
A = morning duty. P = afternoon duty. N = night duty. SD = study day. D5 = 9am to 5pm normal office duty. DO = day off immediately after annual leave. O = day off. AL = annual leave.

We compared the technical configuration of the staff rostering system between the commercial package implemented in Hong Kong Hospital Authority in the Year 2000 and our open-source platform [11]. Our system can run on a desktop computer deployed with an end user tool while commercial packages may require more sophisticated client server configuration. The rostering engine of the commercial package was developed using the Microsoft Visual Studio development environment and class libraries from ILOG were used for its constraint programming capability [12]. Our open-source platform system is based on Python and Google freely available OR-tools CP-SAT libraries [6]. We observed that the open-source platform has strong information technology community support and many sample programs with similar functions applicable to NSP can be found in the community support Github [13]. Technical questions can be raised in the community and prompt responses are received. This greatly enhances the efficiency in developing and maintaining our system.

There are some limitations to our algorithm. If a hard constraint is not fulfilled, one cannot identify from the results which hard constraint is not fulfilled. This may not be important in practice as all hard constraints have to be fulfilled to make the roster schedule valid. Further development and enhancement of the code include the consideration of the remaining annual leave for staff members and the adoption of different constraints only specific to certain wards. As an example, a psychiatry ward requires at least one male nurse for night duty. Therefore, the current study has a wide spectrum of clinical applications across various patient wards in hospitals.

5. Conclusions

An economical but yet highly efficient and user-friendly solution has been developed using open-source operational research tools. The developed system can easily be modified to fulfill different requests from nurses and the optimization rules can be easily implemented in the code. Hence it can be applied to wards of different specialties in hospitals. It demonstrates that AI optimization engines available from open-source and end user tools are powerful to effectively assist in regular healthcare workloads such as NSP. The incorporation of such engines into the hospital roster planning system will minimize the cost of system development to regularly enhance the flexibility in fulfilling different requirements of policies and standards in healthcare sectors, in addition to maximizing the fairness and equality of the ward personnel in their duty assignments.

Declaration of competing interest

The authors have declared no conflict of interest.

Data availability statement

The datasets used and analyzed during the current study are available from the corresponding author on reasonable request.

Appendices. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.ijnss.2022.06.013.

References

[1] El Adoly AA, Gheith M, Nashat Fors M. A new formulation and solution for the nurse scheduling problem: a case study in Egypt. Alex Eng J 2018;57(4): 2289–98. https://doi.org/10.1016/j.aej.2017.09.007.

[2] Chen PS, Zeng ZY. Developing two heuristic algorithms with metaheuristic algorithms to improve solutions of optimization problems with soft and hard constraints: an application to nurse rostering problems. Appl Soft Comput 2020;93:106336. https://doi.org/10.1016/j.asoc.2020.106336.

[3] Nobil AH, Sharifnia SME, Cárdenas-Barrón LE. Mixed integer linear programming problem for personnel multi-day shift scheduling: a case study in an Iran hospital. Alex Eng J 2022;61(1):419–26. https://doi.org/10.1016/j.aej.2021.06.030.

[4] Abdennadher S, Schlenker H. Nurse scheduling using constraint logic programming. In: Proceedings of the sixteenth national conference on Artificial intelligence and the eleventh Innovative applications of artificial intelligence conference innovative applications of artificial intelligence. New York: ACM; 1999. p. 838–43.

[5] Hospital Authority of Hong Kong. www.ha.org.hk; 2021 [Accessed 2022-03-12].

[6] OR-Tools. https://developers.google.com/optimization [Accessed 2022-03-12].

[7] Gurobi. https://www.gurobi.com [Accessed 2022-03-12].

[8] CPLEX. https://www.ibm.com/analytics/cplex-optimize [Accessed 2022-03-12].

[9] LeCun Y, Bengio Y, Hinton G. Deep learning. Nature 2015;521(7553):436–44. https://doi.org/10.1038/nature14539.

[10] NHS England and NHS Improvement. E-rostering the clinical workforce: levels of attainment and meaningful use standards. https://www.england.nhs.uk/wp-content/uploads/2020/09/E-rostering_meaningful_use_standards.pdf [Accessed 2022-03-12].

[11] Chun A, Chan S, Lam G, et al. Nurse rostering at the hospital Authority of Hong Kong. In: Conference: proceedings of the seventeenth national conference on artificial intelligence and twelfth conference on innovative applications of artificial intelligence; 2000. July 30 - August 3, Austin, Texas, USA.

[12] ILOG. http://www.ilog.com [Accessed 2022-03-12].

[13] Github. https://github.com/topics/nurse-scheduling-problem [Accessed 2022-03-12].