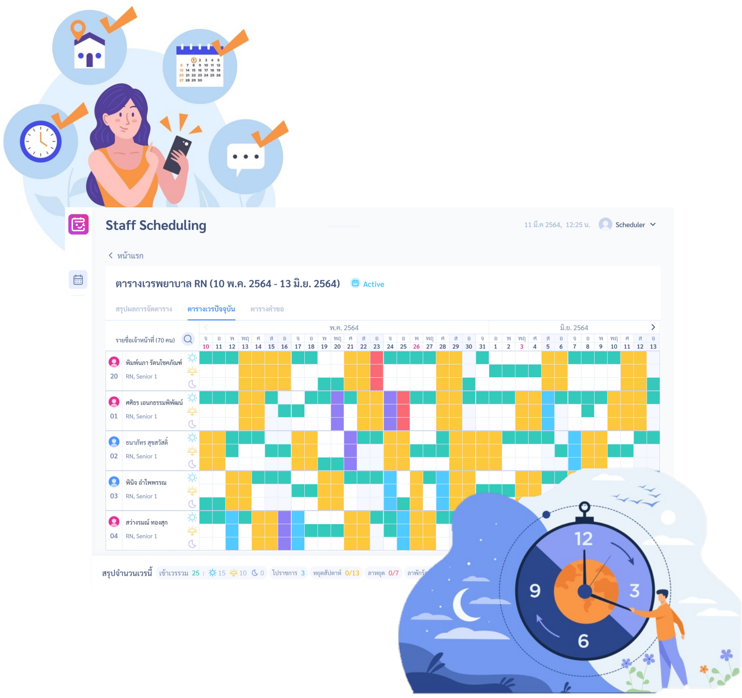

Healthcare Staff Scheduling at an Emergency Department in Thailand

Master's thesis in Electrical Engineering

CHATTARIN WANGWITTAYA

DEPARTMENT OF ELECTRICAL ENGINEERING

CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2022
www.chalmers.se

Master's thesis 2022

Healthcare Staff Scheduling at an Emergency Department in Thailand

CHATTARIN WANGWITTAYA

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Chalmers University of Technology
Gothenburg, Sweden 2022

Healthcare Staff Scheduling at an
Emergency Department in Thailand
CHATTARIN WANGWITTAYA

Supervisor: Kristofer Bengtsson, Associate Professor, Department of Electrical Engineering
Supervisor: Sabino Francesco Roselli, PhD Student, Department of Electrical Engineering
Supervisor: Kamonwat Suksumek, H LAB Co.,Ltd
Supervisor: Pawat Siriwattanayotin, H LAB Co.,Ltd
Examiner: Petter Falkman, Associate Professor, Department of Electrical Engineering

Master's Thesis 2022
Department of Electrical Engineering
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: A user interface design of the healthcare staff scheduling software embedded with the optimization model.

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2022

The Optimization of Healthcare Staff Scheduling in the Emergency Department
CHATTARIN WANGWITTAYA
Department of Electrical Engineering
Chalmers University of Technology

Abstract

Healthcare staff scheduling has been renowned for its correlation with service quality, care outcome, and staff turnover rate. Nevertheless, the complexity of the process usually impedes the hospital from achieving those goals. Particularly at the emergency department of Siriraj Hospital, the complications in scheduling are expedited by the high number of registered nurses (RNs) and the policy for ensuring adequate care service. To enhance the efficacy of human resource management, this thesis investigates the optimization model's capability in the on-duty scheduling of RNs. The scheduling requirements were collected from the interviews with four stakeholders from the management team and the governed staff. The service blueprint was created to visualize the scheduling process, and the mathematical model was formulated following the collected requirements. There are two optimization models developed in this study, i.e., the mixed integer linear programming (MILP) model and the genetic algorithm (GA) model. Two sets of scheduling data for testing the models were obtained from the past RNs schedules in May-June and July-August 2021. The performance comparison between the MILP and GA model demonstrated the inefficiency of GA in optimizing the highly constrained problem, as it can provide only 3.95% of evaluation metrics with better outcomes than MILP. In comparing manual and MILP-optimized schedules, both approaches provide more than half of the evaluation metrics with unchanged outcomes, thus having comparable performance in optimizing most of the schedule's features. However, MILP can significantly optimize 24% to 25% of the metrics while having only 6.58% to 9.21% of the metrics with deteriorated outcomes compared to the manual approach. As a result, the MILP optimization model possesses more superior performance than the GA model and manual approach in optimizing the scheduling of RNs at the emergency department of Siriraj Hospital. The MILP optimization in reducing work stress, promoting staff satisfaction, providing fairness, conforming to the policy, and cutting scheduling time can lead to excellence in service quality and care outcome while lowering the turnover rate. Consequently, the optimization of healthcare staff scheduling with the MILP model exerts the capability of human resource management to its greater extent.

Keywords: optimization, staff scheduling, mixed integer programming, genetic algorithm, healthcare, human resource management, emergency department, nurses.

Acknowledgements

I would like to express my sincere gratitude to all of my supervisors from Chalmers University and H LAB. I am very grateful to Kristofer Bengtsson for accepting my thesis topic and supervising me from the beginning to the completion of the project. Without you, I may not have had a once-in-a-lifetime opportunity to pursue my passion in healthcare optimization as my master's thesis, particularly under the supervision of an expert in healthcare operation optimization. I would like to thank Sabino Francesco Rosell for providing valuable guidance throughout the project, thus driving my thesis toward continuous progress. I am much obliged by the enormous support received from H LAB. Gratitude is owed to Kamonwat Suksumek for offering this thesis topic and allowing me to conduct it with your company. I would like to express my appreciation to Pawat Siriwattanayotin for advising and overseeing this thesis toward its success in satisfying users needs. My aspiration has always been to contribute solutions to improve the healthcare process. So, I am deeply touched that all of you helping me achieve this vision.

I sincerely thank Petter Falkman for accepting the examiner position. I would also like to thank my opponents: Lingyn Deng and Nuttida Issdisai, for their time and relevant discussion toward this thesis.

I wish to acknowledge the emergency department of Siriraj Hospital for their dedicated time and cooperation in shaping the project's requirements.

Last but not least, huge thanks to my family, friends, working team, and pets, who consistently support and encourage me to complete this thesis. Most importantly, thanks to my dedication and a little voice in my head that keep telling "You can do it", and yes, I am doing it now.

Chattarin Wangwittaya, Gothenburg, September 2021

Contents

List of Figures **xi**

List of Tables **xiii**

1 Introduction **1**
1.1 Background . . . 1
1.2 Purpose . . . 2
1.3 Objectives . . . 2
1.4 Scope and limitations . . . 3
1.5 Outline . . . 3

2 Theory **5**
2.1 Optimization models in staff scheduling applications . . . 5
2.2 Mixed Integer Linear Programming Optimization . . . 7
2.2.1 Branch-and-Bound Algorithm . . . 7
2.2.2 Branch-and-Cut algorithm . . . 10
2.3 Genetic Algorithm . . . 10
2.3.1 Population initialization . . . 11
2.3.2 Decoding . . . 12
2.3.3 Selection . . . 12
2.3.4 Crossover . . . 14
2.3.5 Mutation . . . 16
2.3.6 Elitism . . . 17

3 Methods **19**
3.1 Requirements gathering . . . 19
3.2 Qualitative analysis . . . 19
3.3 Mathematical model formulation . . . 20
3.3.1 Model indices . . . 20
3.3.2 The model's primary parameters . . . 21
3.3.3 The model's secondary parameters . . . 23
3.3.4 The model's decision variables . . . 23
3.3.5 The model's objectives . . . 24
3.4 Acquisition of scheduling data . . . 25
3.5 Mixed integer linear programming development . . . 26
3.5.1 Mathematical model . . . 26
3.5.2 Computer implementation . . . 30

3.6 Genetic algorithm programming development . . . 30
3.6.1 Population initialization . . . 30
3.6.2 Decoding . . . 31
3.6.3 Tournament selection . . . 31
3.6.4 Crossover . . . 32
3.6.5 Mutation . . . 32
3.6.6 Elitism . . . 32
3.6.7 Parameter tuning . . . 32
3.7 Performance comparison . . . 35
3.8 Statistical analysis . . . 37
3.9 Development platform . . . 37

4 Results 39
4.1 Service blueprint mapping . . . 39
4.2 In-depth insights extraction . . . 42
4.2.1 Considered elements . . . 42
4.2.1.1 Context of scheduling . . . 42
4.2.1.2 Scheduling policy . . . 43
4.2.2 As-is pains . . . 44
4.2.3 As-is satisfaction . . . 44
4.2.4 To-be preference . . . 45
4.3 Performance comparison . . . 45
4.3.1 The collation of MILP-optimized and GA-optimized schedule . 45
4.3.2 The collation of manual and MILP-optimized schedule . . . 46
4.3.2.1 May-June schedules . . . 46
4.3.2.2 July-August schedules . . . 49

5 Discussion 53
5.1 Service blueprint mapping . . . 53
5.2 In-depth insights extraction . . . 53
5.3 GA's parameter tuning . . . 54
5.4 The collation of MILP-optimized and GA-optimized schedule . . . 56
5.5 The collation of manual and MILP-optimized schedule . . . 56

6 Conclusion 59

Bibliography 61

A Appendix I
A.1 Acquisition of model's secondary parameters . . . I
A.2 Staffing criteria . . . III
A.3 Evaluation algorithm . . . IV
A.4 Results of statistical tests . . . VI

List of Figures

2.1 Example of search tree in branch-and-bound algorithm. 8
2.2 Genetic algorithm entities. 11
2.3 Genetic algorithm flowchart. 11
2.4 Roulette-wheel selection process. 13
2.5 Tournament selection process. 14
2.6 Single-point crossover process. 15
2.7 Multi-point crossover process with consistent switching. 16
2.8 Multi-point crossover process with switching probability. 16
2.9 Uniform crossover process. 17

3.1 Translation process. 31

4.1 Service blueprint of RN's scheduling process. 41
4.2 Heat map representing the significant changes (in percentage) of the MILP-optimized schedule in relation to the GA-optimized schedule. . 46
4.3 Heat map representing the significant changes (in percentage) of the MILP-optimized schedule in relation to the manual schedule of May-June period. 48
4.4 Policy violations of manual schedule in May-June period. 49
4.5 Request approvals percentage among all types of schedule in May-June period. 49
4.6 Heat map representing the significant changes (in percentage) of the MILP-optimized schedule in relation to the manual schedule of July-August period. 51
4.7 Policy violations of manual schedule in July-August period. 51
4.8 Request approvals percentage among all types of schedule in July-August period. 52

List of Tables

3.1 Model sets, indices, and domains. 20
3.2 Definition of index s. 20
3.3 Definition of index l. 21
3.4 Definition of index o. 21
3.5 The model's primary parameters. 22
3.6 The model's secondary parameters. 23
3.7 Model's decision variables. 24
3.8 Model's objective-related variable. 24
3.9 Characteristics of scheduling inputs in May-June period and July-August period. 25
3.10 Parameter tuning scheme 33
3.11 Statistical analysis of Test 1. 34
3.12 F-tests and t-tests between Test 1's parameters and the default setting. 34
3.13 Statistical analysis of Test 2. 34
3.14 F-tests and t-tests between Test 2's parameters and the default setting. 35
3.15 Statistical analysis of Test 3. 35
3.16 F-tests and t-tests between Test 3's parameters and the default setting. 35
3.17 Statistical analysis of Test 4. 35
3.18 F-tests and t-tests between Test 4's parameters and the default setting. 35
3.19 Evaluation standard for performance comparison 36

4.1 Statistical features of the GA-optimized schedule of May-June period. 46
4.2 Statistical features of the manual schedule of May-June period. 48
4.3 Statistical features of the MILP-optimized schedule of May-June period. 48
4.4 The values of decision variables computed from the manual schedule, MILP optimization, and GA optimization in May-June schedule. . . . 48
4.5 Statistical features of the manual schedule of July-August period. . . 50
4.6 Statistical features of the MILP-optimized schedule of July-August period. 50
4.7 The values of decision variables computed from the manual schedule, MILP optimization, and GA optimization in July-August schedule. . 51

A.1 Staffing criteria for weekday III
A.2 Staffing criteria for weekend or holiday III
A.3 F-tests and t-tests of the average shifts between MILP-optimized and GA-optimized schedules (May-June). VI

A.4 F-tests and t-tests of the average working days between MILP-optimized and GA-optimized schedule (May-June). VI
A.5 F-tests and t-tests of the average morning shifts between MILP-optimized and GA-optimized schedule (May-June). VII
A.6 F-tests and t-tests of the average afternoon shifts between MILP-optimized and GA-optimized schedule (May-June). VII
A.7 F-tests and t-tests of the average night shifts between MILP-optimized and GA-optimized schedule (May-June). VII
A.8 F-tests and t-tests of the average double-shift days between MILP-optimized and GA-optimized schedule (May-June). VII
A.9 F-tests and t-tests of the average weekends off between MILP-optimized and GA-optimized schedule (May-June). VIII
A.10 F-tests and t-tests of the average off-bank quota transactions between MILP-optimized and GA-optimized schedule (May-June). VIII
A.11 F-tests and t-tests of the average days off after night shifts between MILP-optimized and GA-optimized schedule (May-June). VIII
A.12 F-tests and t-tests of the average shifts between manual and MILP-optimized schedule (May-June). VIII
A.13 F-tests and t-tests of the average working days between manual and MILP-optimized schedule (May-June). IX
A.14 F-tests and t-tests of the average morning shifts between manual and MILP-optimized schedule (May-June). IX
A.15 F-tests and t-tests of the average afternoon shifts between manual and MILP-optimized schedule (May-June). IX
A.16 F-tests and t-tests of the average night shifts between manual and MILP-optimized schedule (May-June). IX
A.17 F-tests and t-tests of the average double-shifts days between manual and MILP-optimized schedule (May-June). X
A.18 F-tests and t-tests of the average weekends off between manual and MILP-optimized schedule (May-June). X
A.19 F-tests and t-tests of the average off-bank quota transactions between manual and MILP-optimized schedule (May-June). X
A.20 F-tests and t-tests of the average days off after night shifts between manual and MILP-optimized schedule (May-June). X
A.21 F-tests and t-tests of the average shifts between manual and MILP-optimized schedule (July-August). XI
A.22 F-tests and t-tests of the average working days between manual and MILP-optimized schedule (July-August). XI
A.23 F-tests and t-tests of the average morning shifts between manual and MILP-optimized schedule (July-August). XI
A.24 F-tests and t-tests of the average afternoon shifts between manual and MILP-optimized schedule (July-August). XI
A.25 F-tests and t-tests of the average night shifts between manual and MILP-optimized schedule (July-August). XII
A.26 F-tests and t-tests of the average double-shifts days between manual and MILP-optimized schedule (July-August). XII

A.27 F-tests and t-tests of the average weekends off between manual and MILP-optimized schedule (July-August). XII
A.28 F-tests and t-tests of the average off-bank quota transactions between manual and MILP-optimized schedule (July-August). XII
A.29 F-tests and t-tests of the average days off after night shifts between manual and MILP-optimized schedule (July-August). XIII

1

Introduction

1.1 Background

A healthcare staff scheduling is about the shift assignments of care providers under a particular period to ensure adequate staff in each shift [1, 2, 3, 4]. In addition to scheduling, staffing quantifies the number of working personnel required in each shift following the expected service demands or patients [2, 5]. Altogether, these two modules establish the balance between staffing requisitions and workloads, thus enhancing care quality and patient safety [5, 6, 7]. Nevertheless, scheduling is more challenging to achieve since it considers a variety of staff attributes.

The significance of scheduling is also contributed by the potent correlation between scheduling results and staff satisfaction [4]. Several factors, such as work stress, fairness, and the staff-to-patient ratio, can affect the service quality and care process outcome [8]. Therefore, one of the keys to exceptional service is to provide staff with satisfaction on their working schedule while obeying the staffing criteria [4, 6, 8, 9].

Still, the complexity impedes the hospital from attaining the scheduling intents. Those complication aspects include staff (e.g., seniority and individual preference), working policy, and limited human resource [5]. As a result, the hospital cannot utilize its resource through efficient scheduling. This scenario causes the decline in staff satisfaction, the possible degradation of service quality, and the failure of workforce retention [5, 8, 9].

World Health Organization (WHO) has addressed nurse shortage in all countries as a solemn issue to resolve within 2030, and one of the proposed solutions is to reduce the nurse's exit rate [10]. In Thailand, the high turnover rate of registered nurses (RNs) has always been a critical issue for the Ministry of Public Health to tackle [11, 12]. The report shows that the annual resignation of Thai nurses is about 4%, and it is anticipated to rise to 15% in 2020 [12]. The past research also discovers that the main factor promoting the exit rate of Thai nurses is job dissatisfaction [11, 12]. More than 40% of Thai nurses experience pressure from working more than 12 hours daily, particularly newly graduated nurses [13]. Furthermore, the excessive number of afternoon and night shifts urges work stress while lowering care quality [11, 12]. Consequently, the managers should satisfy RNs with individual-based working preferences, well-being, and optimal workload [12].

According to all of the problems in healthcare workforce management, optimizing staff scheduling is the key activity to diminish these issues. As mentioned earlier, scheduling is a difficult task, and manually performing it does not exert the optimization to its fullest extent [14]. With this primary need for assistive technology, operation research and computer science have brought modelling solutions to optimize staff scheduling [5]. Their use cases throughout this time have proved that they are efficacious tools for the scheduling task [6]. Additionally, most of the existing solutions contain either a generalized basis of scheduling or a tailor-made model. The first type is applicable for the majority of the hospital's scheduling systems. However, numerous hospitals adopt different characteristics of scheduling that the generalized basis does not include. Therefore, the development of user-centric solutions or tailor-made models is unavoidable in some cases.

Siriraj Hospital has been renowned as the country's oldest and largest medical school in Thailand. In the emergency department (ED), the number of patient visits in 2018 was as high as 97745 cases [15]. Consequently, the 24-hours care service must be procured through staff scheduling. However, the task is complicated and time-consuming for a shift manager to manually execute due to the considerations of legacy policies, several request types, and a vast number of RNs. Besides, the difficulty and human errors usually cause request disapprovals, unmeet staffing requirements, and policy violation in some shifts. As a result, the optimization model should be developed and tailored to fit the context of staff scheduling in the ED of Siriraj Hospital to improve the work-life quality and satisfaction of RNs, while maximizing care services and policy alignments.

1.2 Purpose

To promote the efficacy in human resource management at the ED of Siriraj Hospital by optimizing the healthcare staff schedule.

1.3 Objectives

This thesis focuses on the optimization of the healthcare staff schedule in the ED of Siriraj Hospital by:

1. Collect the requirements for healthcare staff schedules in the emergency department and investigate insights regarding as-is and to-be of the scheduling system.
2. Develop the models for optimizing the healthcare staff schedule in the emergency department and compare the efficiency between different optimization techniques and the manual approach.

1.4 Scope and limitations

This thesis concentrates on the scheduling of healthcare staff in the emergency department of Siriraj Hospital. The staff in this context solely imply registered nurses and does not involve practitioner nurses, ordinary workers, and physicians. Moreover, the project scope does not enclose an OT plan and a tasking table. On-duty scheduling is the only type that this thesis concerns.These exclusions are decided based on the limited time of research. Therefore, only one schedule type with the most complicated scheduling system and one specialization of staff that exists in a large portion are selected in the scope.

The optimization techniques consist of deterministic optimization and stochastic optimization. The development focuses only on a mixed integer linear programming (MILP) for the deterministic type and a genetic algorithm (GA) for the stochastic model. The selections of solution approaches correlate to the limited research time. Additionally, due to future production and cost-saving, MILP only utilizes an open-source solver named CBC (Coin-or branch and cut).

The scheduling data for model inputs and performance comparison are the pairs of past scheduling requests and their corresponding manual schedules. These pairs are retrieved from two periods, i.e., May-June and July-August 2021.

1.5 Outline

This thesis contains six chapters. Following Chapter 1 for introduction section, Chapter 2 describes the past works and project-related theory, including the literature review on optimization models in staff scheduling applications and the principles of MILP and GA. Chapter 3 explains the method used in the requirements gathering, qualitative analysis, mathematics model formulation, acquisition of scheduling data, MILP development, GA development, performance comparison, statistical analysis, and development platform. Chapter 4 exhibits the study results of service blueprint mapping, in-depth insights extraction, GA's parameters tuning, and performance comparison. Then, the discussion on those results is conducted in Chapter 5, which encloses result interpretations and future developments. Finally, the conclusion of this thesis is presented in Chapter 6.

2

Theory

2.1 Optimization models in staff scheduling applications

Optimization models have been renowned for their application in scheduling tasks, especially in the area of operation research and computer science [5]. By considering the degree of randomness in a search direction, the model can be classified into two types, i.e., deterministic and stochastic optimization [16, 17].

The deterministic model solves the optimization problems by utilizing mathematical principles. With the effective implementation of gradient descent or the Hessian matrix, the deterministic algorithm explores the search space with a converging direction toward the optimum [16, 17]. As a result, the deterministic type has been applied to optimize different sorts of constrained problems, including integer programming and mixed integer programming (MIP); both types present with a linear and non-linear basis [17].

M. A. Centeno et al. [18] demonstrated the effectiveness of the integration between simulation model for staffing estimation and the integer linear programming to optimize the working schedule for staff in the ED. M. Isken [19] applied the concept of the MIP to create a flexible and optimal calendar for healthcare staff. His work was also implemented in a tertiary-care hospital as a decision support system for workforce management. S. Topaloglu [20] addressed the scheduling optimization of medical residents through the multi-objective MIP programming model that can provide a much higher schedule quality than the manual approach. J. Brunner and G. Edenharter [21] optimized the MIP with flexible shift lengths for physicians scheduling problems. N. Zinouri [22] demonstrated the powerful combination of patient demand prediction and the MIP for the optimization of healthcare staff scheduling in the surgical ward . T. Chawasemerwa et al. [23] used the MIP along with the non-violation of constraints and the minimization of penalty to generate optimal and fair schedules for doctors. Marchesi et al. [1] addressed the improvement in the care operation process of two hospitals by optimizing scheduling and staffing. In addition to the association in demand uncertainty and staffing, they also translated the scheduling problem into the MIP and used GUROBI solver to identify the optimal working plan. A. Dumrongsiri and P. Chongphaisal [13] applied the MIP optimization to generate the optimal schedule for RNs from one of the hospitals in Thailand. Even though the targeted sites of their research and this thesis are both in Thailand, each hospital adopted inconsistent policies and yearned for different scheduling goals.

The primary perk of using deterministic optimization is the speed of the model's convergence. The deterministic algorithm usually takes less time to approach the problem's optimum when comparing to the stochastic type [17]. Still, the deterministic approach can be slower when solving complicated optimization problems [16]. Additionally, the model often gets stuck at one stationary point and derives the local optimum as the solution instead of the global one [17].

Stochastic optimization is the finding of the optimal solution with randomness as the core of its search direction [17, 24]. The majority of its algorithms are population-based, in which the population is evolving toward the optimum. Besides, the evolution process is where the randomness enters the optimization system [17]. Several stochastic techniques in the optimization problems include particle swarm optimization, simulated annealing, and GA [17, 24].

L. Rosocha et al. [25] applied the stochastic technique of simulated annealing to optimize healthcare staff scheduling, which improved worker's well-being, staff preference, and quality of work. N. Mohd Rasip et al. [26] utilized particle swarm optimization to tackle the nurse scheduling task in one of the Malaysian public hospitals. Their stochastic model also incorporated staff satisfaction, work balance, and service demand matching as the optimization goals. W. Abo-Hamad and A. Arisha [27] developed the combination of the GA and the clonal selection algorithm to optimize ED staff scheduling in one of the Irish public hospitals. Their case study showed enhancements in the shift balance among workers and the continuity of care services. M. Mutingi [28] demonstrated the application of fuzzy GA for optimizing healthcare staff scheduling. Instead of returning a single output, his model can generate a set of scheduling solutions for the shift manager to decide. K. Leksakul and S. Phetsawat [29] optimized nurse scheduling problems using the GA to achieve the following purposes: adequate service provision, minimizing labour costs, and the balance of OT assignments. A. Amindoust et al. [30] demonstrated the effectiveness of GA to optimize nurse scheduling by minimizing work fatigues. Their use case in one of the Iran hospitals contributed more optimal results compared to the manual scheduling.

The pros and cons of using stochastic optimization arise from the randomness in its algorithm. Adding randomness to the search process makes the algorithm less prone to halt in one local optimum, thus having more possibility to approach the global optimum [17, 24]. As a result, this method is more suitable for solving highly complicated problems, such as high dimensional, non-linear, and no stationary points, that deterministic optimization may fail to achieve [24]. Nevertheless, its randomness can provoke the predicament between having robustness to explore search space and having fast convergence to the optimum. Therefore, the stochastic type usually takes much time to converge when compared to the deterministic one [17].

2.2 Mixed Integer Linear Programming Optimization

MILP is the mathematical model developed for the optimization problem where at least one of its decision variables must be an integer [17, 31]. The subsequent explanations of MILP are based on the publications of various authors [17, 31, 32, 33]. The following equations (Eq.2.1 to 2.4) express the MILP formulation and its principle, including two main elements, i.e., optimization objective and constraints. Optimization objective:

$$Minimize(z = \bar{c}^T \bar{x}) \tag{2.1}$$

Subject to:

$$A\bar{x} \leq \bar{b} \tag{2.2}$$

$$\bar{l} \leq \bar{x} \leq \bar{u} \tag{2.3}$$

$$x_i \in \mathbb{Z}, \forall i \in I \tag{2.4}$$

The ultimate goal of MILP is to achieve the optimization objective, which usually presents in minimization purpose as shown in Eq.2.1. The examples of objective include the reduction of working hours and the decrease in cost or material. From Eq.2.1, the objective function (z) consists of $\bar{c}^T$ and $\bar{x}$. $\bar{c}^T$ is the n-vector of objective coefficients, and $\bar{x}$ is the n-vector of decision variables needed to be solved. The remaining elements are problem's constraints, which can be classified into linear constraints, bound constraint, and integer requirements or integrality restrictions.

Eq.2.2 represents the linear constraints of the problem. A is the $m \times n$ matrix for formulating a linear system in the matrix form, and $\bar{b}$ is the m-vector of linear constraints coefficients. The bound constraints in Eq.2.3 define the permissible range of decision variables values. l and u are the n-vector of lower and upper bound, respectively. For the integrality restrictions, Eq.2.4 implies that at least one of the decision variables must be an integer, which is also the criteria of being MILP. From Eq.2.4, x_i is the ith-variable from $\bar{x}$, $\mathbb{Z}$ is the set of integer, and I is the set of indices for integer variables.

2.2.1 Branch-and-Bound Algorithm

A branch-and-bound is the fundamental method in solving and optimizing the majority of the MILP problems [17, 34, 35]. Its concept involves solving linear programming relaxation of the original MILP whose integrality restrictions are discarded. In other words, the MILP becomes linear programming during this relaxation [17, 34].

The following descriptions of the branch-and-bound algorithm are based on the academic publications [17, 31, 35]. To find the bounded, feasible, and optimal solutions, the formation of tree search usually occurs after receiving the non-practical solution that is not aligned with the integrality constraints. The tree search requires branching into nodes to partition the feasible region into sub-area. Then, the algorithm will assay the MILP problem that emerged from the sub-domain of each node. The branching of the particular node will stop under three instances. The first one is when the problem returns a bounded solution, i.e., all variable values align with the integrality constraints. The second case is when the problem becomes infeasible. The last one is when the yielded value of the objective function is inferior to the current best.

The search tree in Fig.2.1 demonstrates the branch-and-bound algorithm on the MILP example, which is based on the principles described by the published journals [17, 31, 35]. This MILP example comprises two variables (i.e., x_1 and x_2), the objective function (Eq.2.5), linear constraints (Eq.2.6 and 2.7), bound constraints (Eq.2.8 and 2.9), and integrality restrictions (Eq.2.10). The algorithm begins by dropping all integrality restrictions in the original MILP and then solving that relaxed problem with the remaining constraints.

Optimization objective:

$$Minimize(z = 5x_1 + 3x_2 - 2.5) \tag{2.5}$$

Subject to:

$$4x_1 + 3x_2 \geq 3 \tag{2.6}$$

$$x_1 + 2x_2 \geq 3 \tag{2.7}$$

$$0 \leq x_1 \tag{2.8}$$

$$0 \leq x_2 \tag{2.9}$$

$$x_i \in \mathbb{Z}, \forall i \in [1, 2] \tag{2.10}$$

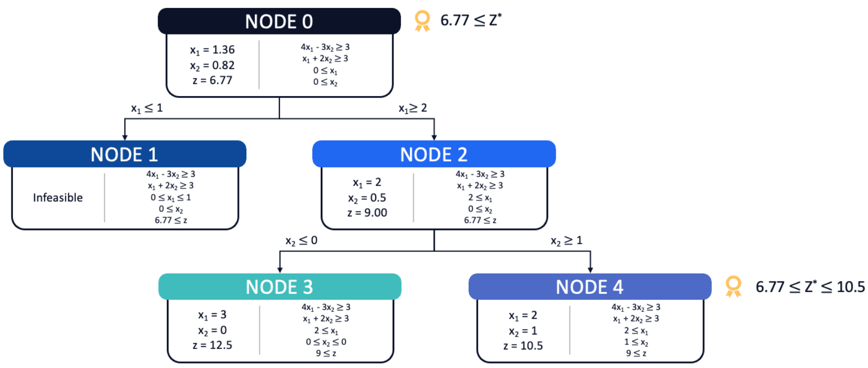

Figure 2.1: Example of search tree in branch-and-bound algorithm.

This linear programming relaxation establishes the root of the search tree as represented by node 0 in Fig.2.1. The optimal solution of node 0 indicates that its minimum objective function (z) is 6.77. As the goal is to minimize the objective function, 6.77 becomes the absolute lower bound of the original MILP's objective function (z^*). Adding more constraints into the system will only cause the objective function to be greater or equal to this lower bound. As a result, the branch-and-bound method aims to find the feasible solution that returns the optimal answer near or at this lower bound.

Since x_1 and x_2 in this node are not yet integers as the original MILP requires, the branching will force either one of these variables to become integers. In other words, it is the partition of the feasible region. In this scenario, two choices of partition exist between subdividing x_1 and x_2. Assuming x_1 is selected, then x_1 must hold the integer value, not the continuous value like 1.36 in node 0. Thus, the x_1 domain must be divided into its proximity integer area, i.e. $x_1 \leq 1$ or $x_1 \geq 2$, and the branching will emerge into two nodes (i.e., node 2 and 3).

Supposing the algorithm picks node 2 for evaluation, the bound constraints of $x_1 \geq 2$ will merge with all constraints from its previous node. After merging, the process is similar to node 0, which involves solving the current node's problem and performs branching if needed. Even x_1 is an integer in the optimal solution, x_2 with the value of 0.5 is not. So, the branch-and-bound process performs branching on x_2 by diving the feasible region into its closet integer range, i.e., $x_2 \leq 0$ or $x_2 \geq 1$. Consequently, the branching creates node 3 and node 4.

Additionally, the objective function's value of 9.00 in node 2 implies that its children branches with the added bound constraints will never overcome this optimum. Hence, if the algorithm keeps digging down node 2's family, it is aware that the optimal solution would yield no lower than 9.00 of the objective function, even when it finds the feasible one.

At this point, the process is continuously occurring from node to node until it falls into one of the three previously-mentioned stopping criteria. For instance, if the algorithm selected node 1 and solving node 1 returns an infeasible solution, the process will stop branching and start looking for other nodes (if any) instead.

In this case, the feasible solutions exist on node 3 and node 4, which align with integrality restrictions. As a result, the branching is ended. Since other nodes are also terminated, and the upper bound contributed from node 4 (i.e., $6.7 \leq z^* \leq 10.5$) is the closest one to the absolute lower bound, the solution from node 4 is concluded to be the most optimum for the original MILP problem.

Nevertheless, not all nodes are required to reach the termination of the branching. In some scenarios, if the acceptable gap between the upper and the absolute lower bound is achieved at a particular node, and its solution is compiled with all constraints, the algorithm will end and return that solution as the most optimal one.

2.2.2 Branch-and-Cut algorithm

The partition of the feasible region and the selection of nodes are diverse, thus forming several branch-and-bound-based algorithms [31]. Their principal aims are to reduce the size of tree search while improving speed to minimize the duality gap (i.e., the difference between the upper and lower bound of the MILP) [17]. Since combining the branch-and-bound approach with the cutting plane method proves its efficacy in finding the optimal solution, a branch-and-cut algorithm has been claimed as the state-of-the-art method for solving MILP. Its approach includes the generation of cutting planes when solving the relaxation problem defined in the particular node [35, 34, 36, 37]. The cutting planes introduce a set of constraints to that node's problem, not just a single bound constraint like the case of the branch-and-bound algorithm [34]. These cuttings considerably reduce each node's duality gap and likewise lowering the number of its children nodes. As a result, the tree size is smaller, and the optimum is obtained via fewer nodes comparing to the original branch-and-bound algorithm [34, 37]. Nonetheless, performing numerous cuts in each node leads to the more complicated formulation and significantly delays the solving process [34, 36, 37].

Besides the cutting method, the branch-and-cut also involves several search strategies, e.g., the decision on node choices and the partition of variables domain. As a result, most solvers enhance and apply different techniques to these elements in the branch-and-cut algorithm [36]. The branch-and-cut-based solvers cover the commercial type, such as CPLEX and Gurobi, and the open-source solver like CBC [38, 33, 39]. Additionally, the benchmark tests of different solvers were conducted by some study groups with the following results [40, 39]. It is found that even though the computation time depends on the problem characteristics (size and complexity), commercial solvers like CPLEX and Gurobi usually compute MILP faster than any open-source solvers. Nevertheless, because of the required licensing for commercial solvers, the open-source CBC is the alternative candidate. Moreover, CBC habitually outperforms other open-source solvers like GLPK, LP_solve and SYMPHONY.

2.3 Genetic Algorithm

The genetic algorithm (GA) is the stochastic optimization approach. It replicates the evolving nature of the population combined with the biological process of genetics [17, 41]. M. Wahde [41] defined four crucial GA entities: population, chromosome, gene, and generation (as illustrated in Fig.2.2). The population is the set of chromosomes, where the chromosome is the string of encoded genes. Each numerical bit established in the chromosome represents the gene. Moreover, GA requires the iterative evolution of its population in the form of generation.

From Fig.2.3, M. Wahde [41] explains the algorithm flowchart containing six elemental activities: population initialization, decoding, tournament selection, crossover, mutation, and elitism. The cyclic process occurs between the decoding and elitism node, and it will end once the termination criteria have been reached. The termina-

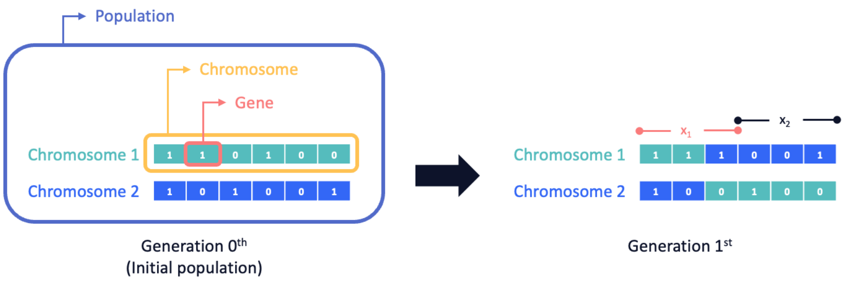

Figure 2.2: Genetic algorithm entities.

tion criteria include the acquisition of an anticipated fitness value and the completion of pre-defined generations.

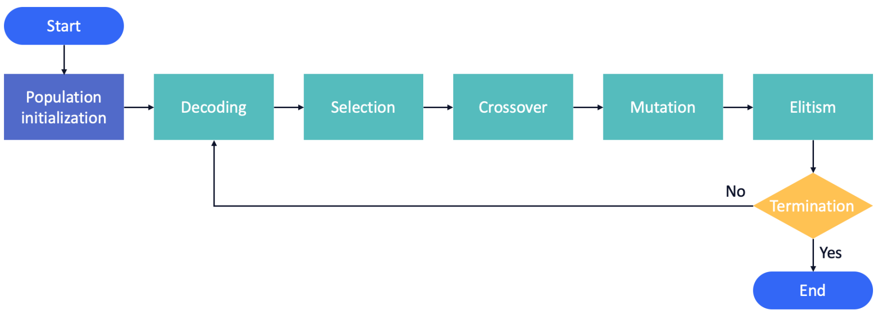

Figure 2.3: Genetic algorithm flowchart.

2.3.1 Population initialization

The following details of population initialization are primarily referred to M. Wahde's work [41]. Before iterating the GA process, an initial population must be constituted as the raw material for the algorithm. So, the population initialization must create chromosomes of the first generation based on the encoding system. The most commonly used system is binary encoding, in which each bit or gene in the chromosome string can only be zero or one. However, other encoding systems are available depending on the context of the problem, such as real number encoding and permutation encoding.

By applying binary encoding, the chromosomes in the first generation are composed of binary random genes. In addition, the number of genes in each chromosome must align with the design of the variables encoding scheme. For example, if there are two variables in the problem, i.e., x_1 and x_2, and each variable is binary encoded by three bits as illustrated in Fig.2.2, the chromosomal length ($CHROMOSOME\_LENGTH$) corresponding to these two variables would be six genes. Hence, the first three genes are encoding for x_1, and the remaining genes are encoding for x_2. Once all the chromosomes in the first generation are successfully produced per the defined population size, the initialization process is completed.

Besides, the works by H. Maaranen et al. [42] and S. Poles et al. [43] show that the initial population does not have to always be in the form of random numbers. Their studies experimented with other types of the initial population and found a correlation between the initial population and the convergence of GA.

2.3.2 Decoding

According to M. Wahde's publication [41], the details on the decoding process are given as follows. Decoding is the first step of the GA cyclic process. It is the step where each chromosome in the population will be evaluated for its fitness (i.e., the objective function of the problem) using the variables decoded from the chromosome. To decode the variables from a chromosome, one must follow the design of the variables encoding scheme. There are many types of the encoding scheme, such as many-bits-per-variable (as in Fig.2.2) and one-bit-per-variable. By assuming binary encoding, the decoding of the many-bits-per-variable system is obtained using Eq.2.11. In Eq.2.11, x is the variable to be decoded, r_{lower} and r_{upper} are the lower and upper bound of x, respectively. g_i is the value of the i^{th} bit encoding x, and k is the total number of encoding bits of x. In the one-bit-per-variable system, the value of x is decoded from Eq.2.12. Eq.2.12's elements are similar to Eq.2.11; the single difference is that only one gene is responsible for encoding each variable. These two schemes contribute different consequences in the stochastic evolution. In the many-bits-per-variable system, the random change in one bit may lead to a slighter modification of the decoded variable than the one-bit-per-variable scheme. This step will be completed once all chromosomes have been decoded and evaluated. Therefore, the final products of this process are the fitness values of chromosomes.

$$x = -r_{lower} + \frac{2r_{upper}}{1-2^{-k}} \sum_{i=1}^{k} 2^{-i} g_i \tag{2.11}$$

$$x = -r_{lower} + 2r_{upper} g \tag{2.12}$$

2.3.3 Selection

The details on the decoding process in this section are wholly based on the publication from M. Wahde [41]. After obtaining the fitness values of the current population, the selection process is executed to indicate the pair of chromosomes for the crossover. There are two commonly-used selections: a roulette-wheel selection and a tournament selection.

The roulette-wheel selection imitates the scenario of spinning a roulette wheel. As demonstrated in Fig.2.4, slots in the wheel represent each chromosome's normalized fitness value stacking cumulatively. In total, all fitness values will add up to 1, corresponding to the wheel's complete circle. To start the selection process, one generates a random number (r) whose value is ranging from 0 to 1. If that random number falls into which section of the wheel, the owner of that section will be selected. For instance, if r is randomly generated as 0.2 as shown in Fig.2.4, this value will fall into the section of chromosome 1. Consequently, chromosome 1 is selected by the roulette wheel.

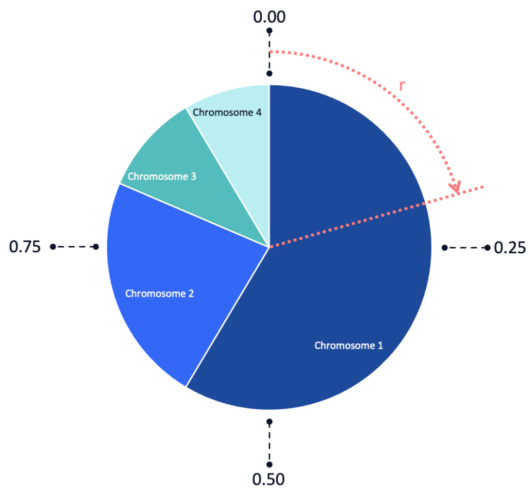

Figure 2.4: Roulette-wheel selection process.

The tournament selection starts with building the tournament batch by equally picking a chromosome set from the current generation. The sampling size from the population is called the tournament size (N_{tnm}). Another parameter in this process is the tournament probability ($PROB_{tnm}$) which is the probability to select a superior individual during the tournament.

As shown in Fig.2.5, the tournament occurs in consecutive rounds, and each round will have one random number generated (whose value ranges from 0 to 1). The random number of round i^{th} or r_i plays a crucial role in determining the tournament's winner. If r_i is less than $PROB_{tnm}$, the current fittest chromosome from the tournament batch will be selected. In contrast, if r_i is greater than or equal to $PROB_{tnm}$, the current fittest chromosome will be removed from the batch, and the next round of the tournament will begin. The tournament repeats until a chromosome is selected or only one chromosome remains in the batch. In the latter case, that one remaining chromosome will be selected.

To completely identify the pair of chromosomes, each pair requires two selection processes. Additionally, both roulette-wheel and tournament selection do not mean that one chromosome can be selected only once. In fact, all selected chromosome returns to the initial population and will participate in the following selections.

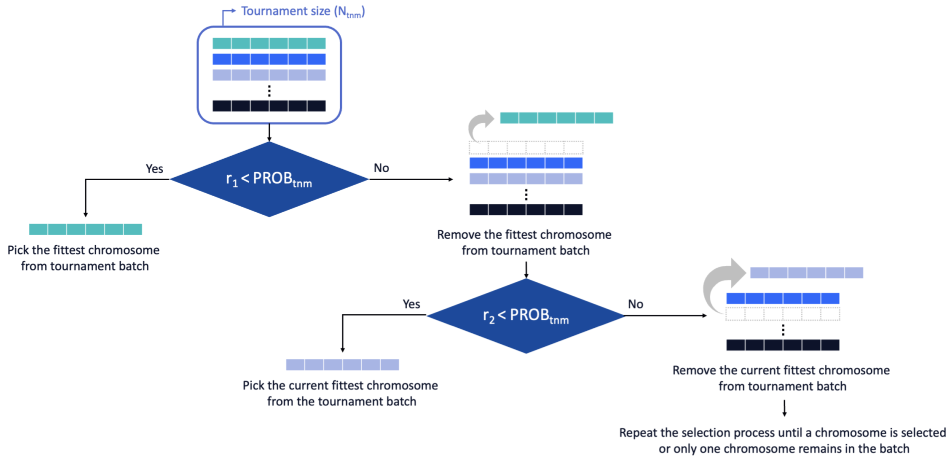

Figure 2.5: Tournament selection process.

The main issue in the selection process is premature convergence. It is a phenomenon that the population converge to a local optimum rather than a global one. Since the roulette-wheel selection prefers the fittest chromosome even in the primary generation, there is a high chance that the algorithm will favour this primarily-fit individual, thus raising the chance of premature convergence. As a result, this issue is much more prominent in the roulette-wheel selection. In addition, the roulette-wheel approach is not applicable with the negative fitness value, while the tournament selection is. Therefore, the problem that contains negative penalty terms in the objective function usually uses the tournament selection.

2.3.4 Crossover

Once the tournament selection defined all parents, crossover processes are initiated to introduce offspring as the next-generation population [41]. By classifying based on the crossover points, there are three major crossover types, i.e., single-point, multi-point, and uniform crossover [41, 44, 45]. The following explanation regarding the crossover process are also based on M. Wahde's work [41]. Regardless of the types, all methods require the value of the crossover probability. There are two possible outcomes in each crossover process: the pair proceeding to the crossover or no crossover at all, as illustrated in Fig.2.6 to 2.9. This critical decision will be made by the crossover probability and the random number generated priorly in each crossover. The crossover will initiate if the random number (whose value ranges from 0 to 1) is lower than the crossover probability. On the other hand, if the random number is at least the crossover probability, there will be no crossover.

According to M. Wahde [41] and K. Sastry and D. E. Goldberg [45], the single-point crossover (Fig.2.6) is when only one random crossover point is simultaneously applied to the pair of chromosomes. In this case, each chromosome will be separated at the crossover point into two parts. Then this pair will exchange their part with one another, i.e., the head part of the reference chromosome will be attached to the tail part of its pair, and vice versa.

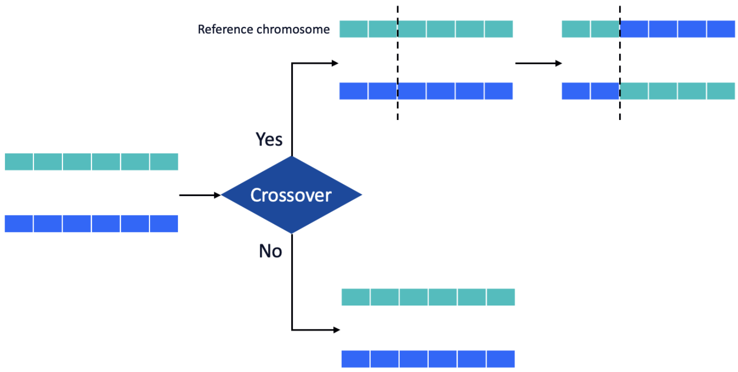

Figure 2.6: Single-point crossover process.

The multi-point crossover is when the number of crossover points is larger than one [41, 44, 45]. It consists of two different approaches, as displayed in Fig.2.7 and 2.8. The main difference between these two approaches is the adoption of switching probability. The first approach is called multi-point crossover with consistent switching. The pair will exchange their parts alternatingly after partitioning each chromosome into sections based on the crossover points, as depicted in Fig.2.7 [44, 45]. In the second approach as described by M. Wahde [41] (Fig.2.8), this multi-point crossover utilizes the switching probability to decide whether each of the reference chromosome's portions will be switched with its pair or not. These choices are decided by comparing the switching probability (usually set to 0.5) with the random number generated in each section (whose value ranges from 0 to 1). If the random number is smaller than the switching probability, the part of the reference chromosome will switch with its pair. On the contrary, if the random number is greater than or equal to the switching probability, there will be no exchange between two chromosomes on that part.

Another type of crossover is the uniform crossover, whose details are described by M. Wahde [41] and K. Sastry and D. E. Goldberg [45]. The uniform crossover is when each gene in the chromosome has a chance to be switched with its pair. According to Fig.2.9, it applies the same principle as the multi-point crossover with switching probability in deciding whether the exchange will be conducted or not.

The crossover is completed once all selected pairs have been processed through the crossover decisions (either proceeding to crossover or no crossover). The results from the crossover decisions constitute a base of the next-generation population [41].

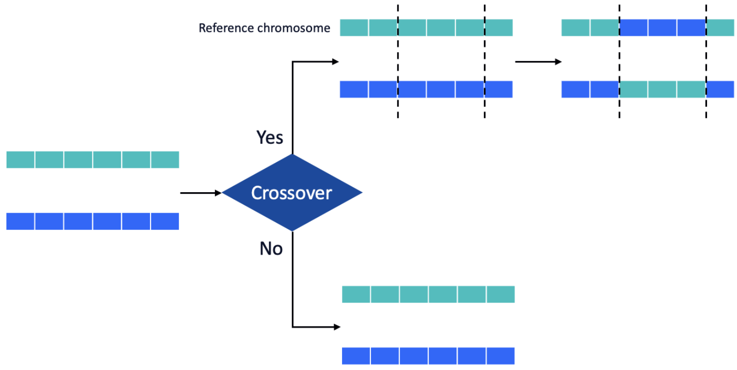

Figure 2.7: Multi-point crossover process with consistent switching.

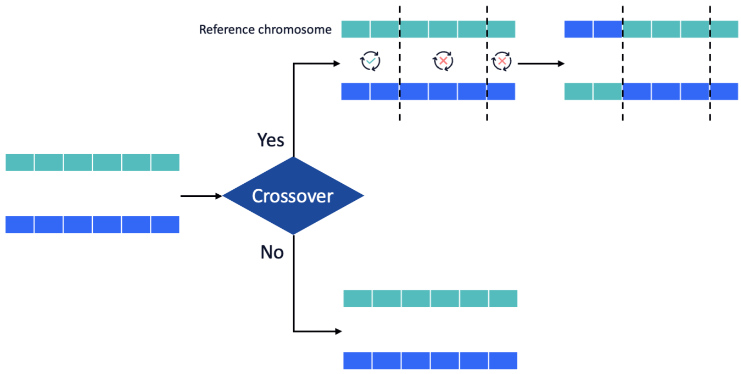

Figure 2.8: Multi-point crossover process with switching probability.

2.3.5 Mutation

The mutation process described in this section is also based on M. Wahde's publication [41]. After obtaining next-generation offspring from the crossover, the mutation introduces new material to the population through slight random changes on the chromosome. This process includes one important parameter, i.e., the mutation constant ($CONST_{mutate}$). The mutation constant is responsible for computing the mutation probability ($PROB_{mutate}$) together with the chromosomal length as expressed in Eq.2.13.

Every gene is subjected to the decision of whether the mutation would occur or not. This decision is judged by comparing the mutation probability with the random number generated in each section (whose value ranges from 0 to 1). If the random number is less than the mutation probability, that particular gene will mutate. In binary encoding, the mutation inversely changes the value of each gene, i.e., from zero to one or one to zero. However, other mutation types also exist, such as

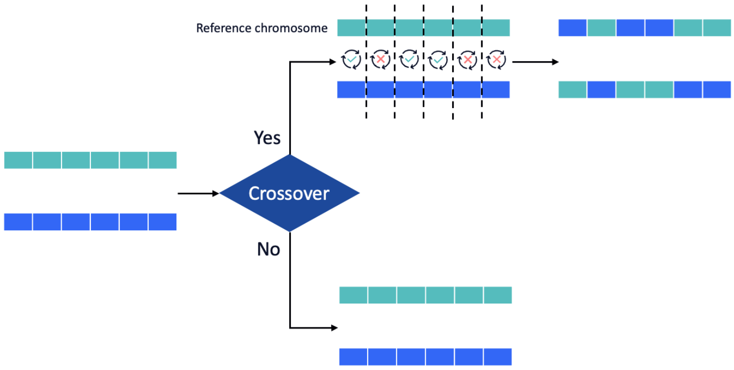

Figure 2.9: Uniform crossover process.

creep mutation for the real number encoding gene. This process is ended when all chromosomes of the next-generation batch process through the mutation decisions.

$$PROB_{mutate} = \frac{CONST_{mutate}}{CHROMOSOME\_LENGTH} \tag{2.13}$$

2.3.6 Elitism

The explanation of elitism is referenced from M. Wahde's book [41] as follows. Elitism is performed as the last process of each generation. Its function is to assure that the highest fitness of the next generation will not decrease from the current one. This process keeps copies of the best chromosome from the pre-crossover population and places them in the post-mutation population (i.e., the batch of the next-generation population).

3 Methods

3.1 Requirements gathering

In order to obtain primary insights for further analysis, a requirements gathering step was conducted. This process enclosed one-on-one interviews with representative users (i.e., RNs) from the ED of Siriraj Hospital.

Regarding the selection of subjects, the inclusion of all stakeholders' perspectives was considered in an attempt to develop a user-centric solution. Therefore, the ED head nurse selected four representatives from both shift managers and governed staff for the interviews. The shift managers' side consisted of one ED head nurse and one on-duty-roster scheduler. On the governed staff side, there were one senior RN and one junior RN.

The one-on-one user interviews through a 30-minutes phone call each were conducted once the selection of subjects had been completed. The users were asked to describe as-is states and to-be aspects of the scheduling process, including process steps, considered elements, pains, satisfaction, and preferences. To get users' insights without any bias contributed from the questions, the interviews were performed with open-ended questions (e.g., what, why, and how-type of questions).

3.2 Qualitative analysis

A qualitative analysis was conducted on the gathered requirements to understand a scheduling process and earn additional in-depth comprehension. The analysis step included a service blueprint mapping and an in-depth insights extraction.

The first step of the qualitative analysis was to visualize the as-is scheduling process using a service blueprint tool. The blueprint was developed by converting the collected process insights into the diagrams of five service components, i.e., customer/user actions, onstage actions, backstage actions, support processes, evidence, and time metric. Following the development, the blueprint was validated by the ED head nurse to establish that a mutual understanding of the process was yielded.

In addition to utilising process details in the service blueprint mapping, the remaining requirements were subjected to an in-depth insights extraction process. The collected information was classified into four categories, i.e., considered elements, pains, satisfaction, and to-be preferences. These extracted insights were further used to define the development direction in terms of the model parameters, constraints, and objective functions.

3.3 Mathematical model formulation

A mathematical model was formulated in accordance with the extracted requirements. All of the considered elements, policy, and goals in the scheduling process were translates into the system of indices, parameters, and variables in mathematical model. This step provided the model with the essential building blocks and environment for the optimization of the scheduling problem.

3.3.1 Model indices

Indices were applied to aid the explication of parameters and variables in the model. They were defined to serve all the domains in the scheduling settings and the optimization process. Table 3.1 shows the employed indices and their corresponding details. Additionally, the definitions of index s, l, and o were presented in Table 3.2, 3.3, and 3.4, respectively

Table 3.1: Model sets, indices, and domains.

Set	Index	Domain	Description
W	w	$\{1, \ldots, W_{max}\}$	Weeks in a schedule
D	d	$\{1, \ldots, D_{max}\}$	Days in a schedule
PD	pd	$\{1, \ldots, PD_{max}\}$	Last days in a previous schedule
S	s	$\{1, \ldots, S_{max}\}$	Shifts in a day
N	n	$\{1, \ldots, N_{max}\}$	Nurses in a roster
L	l	$\{1, \ldots, L_{max}\}$	Levels of nurses
O	o	$\{1, \ldots, O_{max}\}$	Optimization objectives in a model

Table 3.2: Definition of index s.

Index s	Representation
1	Morning shift
2	Afternoon shift
3	Night shift

Table 3.3: Definition of index l.

Index l	Representation
6	Senior 1
5	Senior 2
4	Senior 3
3	Middle
2	Junior 1
1	Junior 2

Table 3.4: Definition of index o.

Index o	Representation
1	To minimize the maximum number of working shifts among all nurses
2	To maximize the balance of total working shifts among all nurses
3	To minimize the maximum number of double-shifts days among all nurses
4	To maximize the minimum number of weekends off among all nurses
5	To minimize the maximum number of disapprovals for vacation requests among all nurses
6	To minimize the maximum number of disapprovals for off requests among all nurses
7	To minimize the maximum number of disapprovals for not-to-be-assigned shift requests among all nurses
8	To minimize the maximum number of disapprovals for shift requests among all nurses

3.3.2 The model's primary parameters

The primary parameters were obtained directly from user's requests or initially defined as scheduling settings and constraints. In other words, they served their purpose as the primary inputs of the model. The formulated parameters were displayed in Table 3.5.

Table 3.5: The model's primary parameters.

Parameter	Description	Example value
W_{max}	Maximum number of weeks in a schedule	5
D_{max}	Maximum number of days in a schedule	35
PD_{max}	Maximum number of last days in a previous schedule	2
S_{max}	Maximum number of shifts in a day	3
N_{max}	Maximum number of nurses in a roster	Month dependent
$ONDUTY\_DAYS$	Number of on-duty days	Month dependent
$OFF\_CALENDAR_d$	State of being a weekend or a holiday on day d	$\{0,1\}$
L_{max}	The highest level of nurses	$NURSE\_LEVELS_{max}$
$NURSE\_LEVELS_n$	Level of nurse n	$l \in L$
$EXTRA\_NURSES_{d,s}$	Number of extra nurse(s) provided from other departments on day d in shift s	$[0,4]$
$PREV\_SCHEDULE_{n,pd,s}$	State of having a shift in a previous schedule for nurse n on day pd in shift s	$\{0,1\}$
$WEEK\_OFF\_REQUEST_{n,d}$	State of having a request for a weekly-off quota usage from nurse n on day d	$\{0,1\}$
$BANK\_OFF\_REQUEST_{n,d}$	State of having a request for an off-bank quota usage from nurse n on day d	$\{0,1\}$
$OFF\_REQUEST\_VIP_{n,d}$	State of having a VIP request for a day off by nurse n on day d	$\{0,1\}$
$OUT\_REQUEST\_VIP_{n,d}$	State of having a VIP request for an out-of-unit position by nurse n on day d	$\{0,1\}$
$VAC\_REQUEST_{n,d}$	State of having a request for a vacation from nurse n on day d	$\{0,1\}$
$SHIFT\_REQUEST_{n,d,s}$	State of having a request for a shift by nurse n on day d in shift s	$\{0,1\}$
$NO\_SHIFT\_REQUEST_{n,d,s}$	State of having a request for a not-to-be-assigned shift by nurse n on day d in shift s	$\{0,1\}$
$SHIFT\_REQUEST\_VIP_{n,d,s}$	State of having a VIP request for a shift by nurse n on day d in shift s	$\{0,1\}$
$NO\_SHIFT\_REQUEST\_VIP_{n,d,s}$	State of having a VIP request for a not-to-be-assigned shift by nurse n on day d in shift s	$\{0,1\}$
$WEEKDAY\_STAFFING\_MIN_{l,d,s}$	Minimum number of nurses level l required on weekday d in shift s	Table A.1
$WEEKEND\_STAFFING\_MIN_{l,d,s}$	Minimum number of nurses level l required on weekend or holiday d in shift s	Table A.2
$TOTAL\_SHIFTS_{max}$	Maximum number of total shifts per one nurse	30
$TOTAL\_SHIFTS_{min}$	Minimum number of total shifts per one nurse	$ONDUTY\_DAYS$
$TOTAL\_A\_SHIFTS_{max}$	Maximum number of total afternoon shifts per one nurse	33% of $ONDUTY\_DAYS$
$TOTAL\_N\_SHIFTS_{max}$	Maximum number of total night shifts per one nurse	33% of $ONDUTY\_DAYS$
$WEEK\_WORKDAYS_{max}$	Maximum number of weekly working days per one nurse	5
$WEEK\_SHIFTS_{max}$	Maximum number of weekly shifts per one nurse	6
$WEEK\_N\_SHIFTS_{max}$	Maximum number of weekly night shifts per one nurse	3
$CONSEC\_WORKDAYS_{max}$	Maximum number of consecutive working days per one nurse	6
$CONSEC\_A\_SHIFTS_{max}$	Maximum number of consecutive afternoon shifts per one nurse	2
$CONSEC\_N\_SHIFTS\_SENIOR_{max}$	Maximum number of consecutive night shifts per one senior nurse	2
$CONSEC\_N\_SHIFTS\_NONSENIOR_{max}$	Maximum number of consecutive night shifts per one non-senior nurse	3
$TOTAL\_DOUBLE\_SHIFTS\_DAYS_{max}$	Maximum number of total double-shifts days per one nurse	15
$TOTAL\_WEEKENDS\_OFF_{max}$	Maximum number of total weekends off per one nurse	2
$TOTAL\_WEEKENDS\_OFF_{min}$	Minimum number of total weekends off per one nurse	1
O_max	Maximum number of objectives in a model	8
$WEIGHT\_OBJ_o$	Weight of objective o	≥ 0

3.3.3 The model's secondary parameters

The secondary parameters were set to facilitate the formulation of constraints and objective functions. As shown in Appendix A.1, they were solely computed from the primary parameters. Table 3.6 shows the secondary parameters utilized in the model.

Table 3.6: The model's secondary parameters.

Parameter	Description	Referred equation
$WORKDAY\_REQUEST_{n,d}$	State of having a request for a working day from nurse n on day d	A.1,A.2,
$WEEKEND\_WORK\_REQUEST_{n,w}$	State of having a request for working on weekend from nurse n in week w	A.3,
$TOTAL\_WEEKEND\_WORK\_REQUEST_{n}$	Number of total weekend-working requests from nurse n	A.4,
$DOUBLE\_SHIFTS\_DAY\_REQUEST_{n,d}$	State of having a request for a double-shifts-day from nurse n on day d	A.5,A.6
$TOTAL\_DOUBLE\_SHIFTS\_DAY\_REQUEST_{n}$	Number of total double-shifts-day requests from nurse n	A.7
$TOTAL\_OUT\_REQUEST\_VIP_{n}$	Number of total VIP requests for out-of-unit positions from nurse n	A.8
$WEEK\_OUT\_REQUEST\_VIP_{n,w}$	Number of weekly VIP requests for out-of-unit positions from nurse n in week w	A.9
$TOTAL\_OFF\_REQUEST\_VIP_{n}$	Number of total VIP requests for days off from nurse n	A.10
$ALL\_OFF\_REQUEST_{n,d}$	State of having a request for a day off (including off-bank quota and weekly-off quota) from nurse n on day d	A.11
$TOTAL\_ALL\_OFF\_REQUEST_{n}$	Number of total requests for days off from nurse n	A.12
$TOTAL\_ALL\_OFF\_REQUEST_{max}$	Maximum number of total non-VIP requests for days off	A.13
$TOTAL\_BANK\_OFF\_REQUEST_{n}$	Number of total requests for off-bank quota usages from nurse n	A.14
$TOTAL\_ACTUAL\_BANK\_OFF\_USES_{n}$	Number of total and actual off-bank quota usages from nurse n	A.15,A.16,A.17
$TOTAL\_VAC\_REQUEST_{n}$	Number of total requests for vacations from nurse n	A.18
$TOTAL\_VAC\_REQUEST_{max}$	Maximum number of total requests for vacations	A.19
$TOTAL\_SHIFT\_REQUEST_{n}$	Number of total requests for shifts from nurse n	A.20
$TOTAL\_SHIFT\_REQUEST_{max}$	Maximum number of total requests for shifts	A.21
$TOTAL\_NO\_SHIFT\_REQUEST_{n}$	Number of total requests for not-to-be-assigned shifts from nurse n	A.22
$TOTAL\_NO\_SHIFT\_REQUEST_{max}$	Maximum number of total requests for not-to-be-assigned shifts	A.23
$ACTIVE\_OBJ_{o}$	State of having objective o being active in model's objective function	A.24,A.25
$TOTAL\_ACTIVE\_OBJ$	Number of total active objectives in a model	A.26

3.3.4 The model's decision variables

The following decision variables, as represented in Table 3.7, were utilized. They were composed as the key elements that the model should solve to yield the optimized solution and to align with the defined constraints.

Table 3.7: Model's decision variables.

Variable	Description	Domain
$ws_{n,d,s}$	State of having a working shift for nurse n on day d in shift s	$\{0,1\}$
$wd_{n,d}$	State of having a working day for nurse n on day d	$\{0,1\}$
$ad_{n,d}$	State of being absent for nurse n on day d	$\{0,1\}$
ws_{max}	Maximum number of working shifts among all nurses	$[TOTAL\_SHIFTS_{min}, TOTAL\_SHIFTS_{max}]$
ws_{min}	Minimum number of working shifts among all nurses	$[TOTAL\_SHIFTS_{min}, TOTAL\_SHIFTS_{max}]$
$dsd_{n,d}$	State of having double-shifts day for nurse n on day d	$\{0,1\}$
dsd_{max}	Maximum number of double-shifts days among all nurses	$[0, TOTAL\_DOUBLE\_SHIFTS\_DAYS_{max}]$
$wko_{n,w}$	State of having a weekend off for nurse n in week w	$\{0,1\}$
wko_{min}	Minimum number of weekends off among all nurses	$[TOTAL\_WEEKENDS\_OFF_{min}, TOTAL\_WEEKENDS\_OFF_{max}]$
$oa_{n,d}$	State of having an off request approved for nurse n on day d	$\{0,1\}$
$va_{n,d}$	State of having a vacation request approved for nurse n on day d	$\{0,1\}$
$sa_{n,d,s}$	State of having a shift request approved for nurse n on day d in shift s	$\{0,1\}$
$nsa_{n,d,s}$	State of having a not-to-be-assigned shift request approved for nurse n on day d in shift s	$\{0,1\}$
dor_{max}	Maximum number of disapprovals for off requests among all nurses	$[0, TOTAL\_ALL\_OFF\_REQUEST_{max}]$
dvr_{max}	Maximum number of disapprovals for vacation requests among all nurses	$[0, TOTAL\_VAC\_REQUEST_{max}]$
dsr_{max}	Maximum number of disapprovals for shift requests among all nurses	$[0, TOTAL\_SHIFT\_REQUEST_{max}]$
$dnsr_{max}$	Maximum number of disapprovals for not-to-be-assigned shift requests among all nurses	$[0, TOTAL\_NO\_SHIFT\_REQUEST_{max}]$

3.3.5 The model's objectives

According to the collected requirement from users, eight objectives were developed and enclosed in a model's objective function. The definition of each objective was described in Table 3.4. Moreover, each objective was subjected to the min-max normalization approach to make them being comparable to one another and not contributing any bias toward the objective function. The variables representing these normalized objectives were shown in Table 3.8 and in Eq.3.1 to 3.8.

Table 3.8: Model's objective-related variable.

Variable	Description	Domain
no_o	Normalized objective o	$[0,1]$
z	Model's objective function	≥ 0

$$no_1 = \frac{ws_{max} - TOTAL\_SHIFTS_{min}}{TOTAL\_SHIFTS_{max} - TOTAL\_SHIFTS_{min}} \tag{3.1}$$

$$no_2 = \frac{(ws_{max} - ws_{min}) - 0}{(TOTAL\_SHIFTS_{max} - TOTAL\_SHIFTS_{min}) - 0} \tag{3.2}$$

$$no_3 = \frac{dsd_{max} - 0}{TOTAL\_DOUBLE\_SHIFTS\_DAYS_{max} - 0} \tag{3.3}$$

$$no_4 = \frac{TOTAL\_WEEKENDS\_OFF_{max} - wko_{min}}{TOTAL\_WEEKENDS\_OFF_{max} - TOTAL\_WEEKEND\_OFF_{min}} \tag{3.4}$$

$$no_5 = \frac{dvr_{max} - 0}{TOTAL\_VAC\_REQUEST_{max} - 0} \tag{3.5}$$

$$no_6 = \frac{dor_{max} - 0}{TOTAL\_ALL\_OFF\_REQUEST_{max} - 0} \tag{3.6}$$

$$no_7 = \frac{dnsr_{max} - 0}{TOTAL\_NO\_SHIFT\_REQUEST_{max} - 0} \tag{3.7}$$

$$no_8 = \frac{dsr_{max} - 0}{TOTAL\_SHIFT\_REQUEST_{max} - 0} \tag{3.8}$$

All eight normalized objectives were aggregated into the objective function using Eq.3.9. Each active objective was assumed to have an equal weight of 1 by default. Therefore, the objective function was further normalized by the number of total active objectives to yield its primitive value in the range of $[0, 1]$. Nevertheless, the value of z can be greater than the initial range depending on the weights.

$$z = \sum_{o=1}^{O_{max}} \frac{WEIGHT\_OBJ_o \times no_o}{TOTAL\_ACTIVE\_OBJ} \tag{3.9}$$

3.4 Acquisition of scheduling data

The RNs from the emergency department of Siriraj Hospital have used Google Sheets as a platform for RNs to request their schedules, as well as to let the shift manager perform the scheduling task. Therefore, the pairs of input parameters and their corresponding manual schedule in the past were retrieved from the repository of those Google Sheets. There were two pairs of input and output enclosed in the development of this project, i.e., the May-June schedule and the July-August schedule of 2021. The acquisition of scheduling data served its purpose as model's inputs and performance-comparing reference. The characteristics of scheduling inputs in May-June period and July-August period were not homologous as described in Table 3.9.

Table 3.9: Characteristics of scheduling inputs in May-June period and July-August period.

	May - June	July - August
Number of nurses	71	69
Schedule duration (days)	35	35
Number of on-duty days	22	23
Number of VIP requests for out-of-unit positions	117	99
Number of VIP requests for days off	49	35
Number of normal requests for days off	390	372
Number of normal requests for vacations	23	58
Number of VIP requests for shifts	26	25
Number of normal requests for shifts	126	141
Number of VIP requests for not-to-be-assigned shifts	59	56
Number of normal requests for not-to-be-assigned shifts	252	260

3.5 Mixed integer linear programming development

A multi-objective MILP development enclosed two main steps, i.e., the design of mathematical model and a computer implementation. Altogether, these steps allowed the model to receive the deterministic optimization on the scheduling problem.

3.5.1 Mathematical model

A mathematical model was constructed by utilizing the elements in the previously-defined mathematical model to form a system of constraints and an optimization objective. The constraints were appointed in relation to scheduling aims and policy collected from users. The mathematical model used in this MILP was formulated as shown below.

Optimization objective:

$$Minimize(z) \tag{3.10}$$

Subject to:

Working shift and day relation

$$wd_{n,d} \leq \sum_{s=1}^{S_{max}} ws_{n,d,s}, \forall n \in N, \forall d \in D \tag{3.11}$$

$$wd_{n,d} \geq ws_{n,d,s}, \forall n \in N, \forall d \in D, \forall s \in S \tag{3.12}$$

$$ad_{n,d} = 1 - wd_{n,d}, \forall n \in N, \forall d \in D \tag{3.13}$$

Weekly working shifts and days policy

$$\sum_{d=7(w-1)+1}^{7w} \sum_{s=1}^{S_{max}} ws_{n,d,s} \leq WEEK\_SHIFTS_{max}, \forall n \in N, \forall w \in W \tag{3.14}$$

$$\sum_{d=7(w-1)+1}^{7w} wd_{n,d} + WORVn, w^{\mathrm{a}} \leq WWDMX, \forall n \in N, \forall w \in W \tag{3.15}$$

$$\sum_{d=7(w-1)+1}^{7w} wd_{n,d} \leq CONSEC\_WORKDAYS_{max}, \forall n \in N, \forall w \in W^{i} \tag{3.16}$$

Afternoon shift policy

$$\sum_{d}^{d+CONSEC\_A\_SHIFTS_{max}} ws_{n,d,2} \leq CONSEC\_A\_SHIFTS_{max}, \forall n \in N, \forall d \in D^{i} \tag{3.17}$$

[a] WORV: $WEEK\_OUT\_REQUEST\_VIP$
[b] WWDMX: $WEEK\_WORKDAYS_{max}$

Night shift policy

$$\sum_{d=7(w-1)+1}^{7w} ws_{n,d,3} \leq WEEK\_N\_SHIFTS_{max}, \forall n \in N, \forall w \in W^{i} \tag{3.18}$$

If $NURSE\_LEVELS_n \geq 3$ then Eq.3.19

$$\sum_{d}^{d+CNSSMX^{\mathrm{c}}-1} ws_{n,d,3} - (CNSSMX^{\mathrm{c}} - 1) \leq 2ad_{n,d+CNSSMX^{\mathrm{c}}}, \forall n \in N, \forall d \in D^{i,ii} \tag{3.19}$$

If $NURSE\_LEVELS_n < 3$ then Eq.3.20

$$\sum_{d}^{d+CNSNSMX^{\mathrm{d}}-1} ws_{n,d,3} - (CNSNSMX^{\mathrm{d}} - 1) \leq 2ad_{n,d+CNSNSMX^{\mathrm{d}}}, \forall n \in N, \forall d \in D^{i,ii} \tag{3.20}$$

Consecutive shifts policy

$$ws_{n,d,2} + ws_{n,d,3} \leq 1, \forall n \in N, \forall d \in D \tag{3.21}$$

$$ws_{n,d,3} + ws_{n,d+1,1} \leq 1, \forall n \in N, \forall d \in D \tag{3.22}$$

$$ws_{n,d,2} + ws_{n,d+1,1} \leq 1, \forall n \in N, \forall d \in D^{i} \tag{3.23}$$

$$ws_{n,d,3} + ws_{n,d+1,2} \leq 1, \forall n \in N, \forall d \in D^{i} \tag{3.24}$$

Total shifts policy

$$\sum_{d=1}^{D_{max}} \sum_{s=1}^{S_{max}} ws_{n,d,s} + \sum_{d=1}^{D_{max}} va_{n,d} + TOURV_n{}^{\mathrm{e}} + TABOFU_n{}^{\mathrm{f}} \leq ws_{max}, \forall n \in N^{i} \tag{3.25}$$

$$\sum_{d=1}^{D_{max}} \sum_{s=1}^{S_{max}} ws_{n,d,s} + \sum_{d=1}^{D_{max}} va_{n,d} + TOURV_n{}^{\mathrm{e}} + TABOFU_n{}^{\mathrm{f}} + TOFRV_n{}^{\mathrm{g}} \geq ws_{min}, \forall n \in N \tag{3.26}$$

Staffing policy

If $NURSE\_LEVELS_n \geq l$ and $OFF\_CALENDAR_d \neq 0$ then Eq.3.27 and 3.28

$$\sum_{n=1}^{N_{max}} ws_{n,d,s} \geq \sum_{l}^{L_{max}-1} WDSMI_{l,d,s}{}^{\mathrm{h}}, \forall l \in \{2, \ldots, L_{max} - 1\}, \forall d \in D, \forall s \in S \tag{3.27}$$

$$\sum_{n=1}^{N_{max}} ws_{n,d,s} \geq \sum_{l}^{L_{max}-1} WESMI_{l,d,s}{}^{\mathrm{i}}, \forall l \in \{2, \ldots, L_{max} - 1\}, \forall d \in D, \forall s \in S \tag{3.28}$$

[c] $CNSSMX$: $CONSEC\_N\_SHIFTS\_SENIOR_{max}$
[d] $CNSNSMX$: $CONSEC\_N\_SHIFTS\_NONSENIOR_{max}$
[e] $TOURV$: $TOTAL\_OUT\_REQUEST\_VIP$
[f] $TABOFU$: $TOTAL\_ACTUAL\_BANK\_OFF\_USES$
[g] $TOFRV$: $TOTAL\_OFF\_REQUEST\_VIP$

If $NURSE\_LEVELS_n \geq 1$ and $OFF\_CALENDAR_d \neq 0$ then Eq.3.29 and 3.30

$$\sum_{n=1}^{N_{max}} ws_{n,d,s} \geq \sum_{l=1}^{L_{max}-1} WDSMI_{l,d,s}{}^{\mathrm{h}} - EXTRA\_NURSES_{d,s}, \forall d \in D, \forall s \in S \tag{3.29}$$

$$\sum_{n=1}^{N_{max}} ws_{n,d,s} \geq \sum_{l=1}^{L_{max}-1} WESMI_{l,d,s}{}^{\mathrm{i}} - EXTRA\_NURSES_{d,s}, \forall d \in D, \forall s \in S \tag{3.30}$$

Double-shifts day relation

$$dsd_{n,d} \geq \sum_{s=1}^{S_{max}} ws_{n,d,s} - 1, \forall n \in N, \forall d \in D \tag{3.31}$$

$$2dsd_{n,d} \leq \sum_{s=1}^{S_{max}} ws_{n,d,s}, \forall n \in N, \forall d \in D \tag{3.32}$$

If $DOUBLE\_SHIFTS\_DAY\_REQUEST_{n,d} \neq 0$ then Eq.3.33

$$dsd_{max} \geq \sum_{d=1}^{D_{max}} dsd_{n,d}, \forall n \in N \tag{3.33}$$

Weekend off relation

$$wko_{n,w} \geq ad_{n,d=7w-1} + ad_{n,d=7w} - 1, \forall n \in N, \forall w \in W \tag{3.34}$$

$$wko_{n,w} \leq ad_{n,d=7w-1}, \forall n \in N, \forall w \in W \tag{3.35}$$

$$wko_{n,w} \leq ad_{n,d=7w}, \forall n \in N, \forall w \in W \tag{3.36}$$

If $TWWR_n{}^{\mathrm{j}} \leq W_{max} - TWOMX^{\mathrm{k}}$ then Eq.3.37

$$wko_{min} \leq \sum_{w=1}^{W_{max}} wko_{n,w}, \forall n \in N \tag{3.37}$$

Working preference-related policy
If $NURSE\_LEVELS_n = 6$ then Eq.3.38

$$ws_{n,d,2} + ws_{n,d,3} \leq 0, \forall n \in N, \forall d \in D^i \tag{3.38}$$

If $NURSE\_LEVELS_n = 5$ then Eq.3.39

$$ws_{n,d,3} \leq 0, \forall n \in N, \forall d \in D^i \tag{3.39}$$

Off approval relation

$$oa_{n,d} \geq ALL\_OFF\_REQUEST_{n,d} + ad_{n,d} - 1, \forall n \in N, \forall d \in D \tag{3.40}$$

$$oa_{n,d} \leq ALL\_OFF\_REQUEST_{n,d}, \forall n \in N, \forall d \in D \tag{3.41}$$

[h] $WDSMI$: $WEEKDAY\_STAFFING\_MIN$
[i] $WESMI$: $WEEKEND\_STAFFING\_MIN$
[j] $TWWR$: $TOTAL\_WEEKEND\_WORK\_REQUEST$
[k] $TWOMX$: $TOTAL\_WEEKENDS\_OFF_{max}$

$$oa_{n,d} \leq ad_{n,d}, \forall n \in N, \forall d \in D \tag{3.42}$$

$$dor_{max} \geq TOTAL\_ALL\_OFF\_REQUEST_n - \sum_{d=1}^{D_{max}} oa_{n,d}, \forall n \in N \tag{3.43}$$

Vacation approval relation

$$va_{n,d} \geq VAC\_REQUEST_{n,d} + ad_{n,d} - 1, \forall n \in N, \forall d \in D \tag{3.44}$$

$$va_{n,d} \leq VAC\_REQUEST_{n,d}, \forall n \in N, \forall d \in D \tag{3.45}$$

$$va_{n,d} \leq ad_{n,d}, \forall n \in N, \forall d \in D \tag{3.46}$$

$$dvr_{max} \geq TOTAL\_VAC\_REQUEST_n - \sum_{d=1}^{D_{max}} va_{n,d}, \forall n \in N \tag{3.47}$$

Shift approval relation

$$sa_{n,d,s} \geq SHIFT\_REQUEST_{n,d,s} + ws_{n,d,s} - 1, \forall n \in N, \forall d \in D, \forall s \in S \tag{3.48}$$

$$sa_{n,d,s} \leq SHIFT\_REQUEST_{n,d,s}, \forall n \in N, \forall d \in D, \forall s \in S \tag{3.49}$$

$$sa_{n,d,s} \leq ws_{n,d,s}, \forall n \in N, \forall d \in D, \forall s \in S \tag{3.50}$$

$$dsr_{max} \geq TOTAL\_SHIFT\_REQUEST_n - \sum_{d=1}^{D_{max}} \sum_{s=1}^{S_{max}} sa_{n,d,s}, \forall n \in N \tag{3.51}$$

Not-to-be assigned shift approval relation

$$nsa_{n,d,s} \geq NO\_SHIFT\_REQUEST_{n,d,s} + (1 - ws_{n,d,s}) - 1, \forall n \in N, \forall d \in D, \forall s \in S \tag{3.52}$$

$$nsa_{n,d,s} \leq NO\_SHIFT\_REQUEST_{n,d,s}, \forall n \in N, \forall d \in D, \forall s \in S \tag{3.53}$$

$$nsa_{n,d,s} \leq 1 - ws_{n,d,s}, \forall n \in N, \forall d \in D, \forall s \in S \tag{3.54}$$

$$dnsr_{max} \geq TOTAL\_NO\_SHIFT\_REQUEST_n - \sum_{d=1}^{D_{max}} \sum_{s=1}^{S_{max}} nsa_{n,d,s}, \forall n \in N \tag{3.55}$$

Out-of-unit policy
If $OUT\_REQUEST\_VIP_{n,d} = 1$ then Eq.3.56

$$ad_{n,d} + OUT\_REQUEST\_VIP_{n,d} = 2, \forall n \in N, \forall d \in D \tag{3.56}$$

VIP off request policy
If $OFF\_REQUEST\_VIP_{n,d} = 1$ then Eq.3.57

$$ad_{n,d} + OFF\_REQUEST\_VIP_{n,d} = 2, \forall n \in N, \forall d \in D \tag{3.57}$$

VIP shift request policy
If $SHIFT\_REQUEST\_VIP_{n,d} = 1$ then Eq.3.58

$$ws_{n,d,s} + SHIFT\_REQUEST\_VIP_{n,d,s} = 2, \forall n \in N, \forall d \in D \tag{3.58}$$

VIP not-to-be-assigned shift request policy
If $NO\_SHIFT\_REQUEST\_VIP_{n,d} = 1$ then Eq.3.59

$$ws_{n,d,s} + NO\_SHIFT\_REQUEST\_VIP_{n,d,s} \leq 1, \forall n \in N, \forall d \in D \tag{3.59}$$

Shifts' limitation policy

$$\sum_{d=1}^{D_{max}} ws_{n,d,2} \leq TOTAL\_A\_SHIFTS_{max}, \forall n \in N, \forall d \in D^i \tag{3.60}$$

$$\sum_{d=1}^{D_{max}} ws_{n,d,3} \leq TOTAL\_N\_SHIFTS_{max}, \forall n \in N, \forall d \in D^i \tag{3.61}$$

Remarks:
[i] This constraint will be halted, if a nurse willingly requested for a case that violate the constraint.
[ii] This constraint is also applied to the period between a previous schedule and a current one.

3.5.2 Computer implementation

The mathematical model was implemented using Python and the open-source library PulP to facilitate the optimization process. The open-source solver used in this MILP was the CBC. The termination criteria of the optimization process included the configuration of the relative gap tolerance and time limit. The relative gap tolerance of the objective lower and upper bound was set to 0.175. The time limit for the solver to stop optimizing was fixed to 4000 seconds.

3.6 Genetic algorithm programming development

In addition to the deterministic optimization as in the MILP programming, a GA programming was also developed as an alternative model utilizing stochastic optimization. The elements and the flowchart of GA implemented in this study are in accordance with Fig.2.3. The termination criterion was the number of generations ($N_{generation}$) which was set to be 300. Apart from the designs of these algorithm activities, the tuning process for some parameters was investigated.

3.6.1 Population initialization

In this study, binary encoding was used for the population initialization, where each bit represents the state of having a working shift. Three types of the initial population were evaluated in this thesis. The first type was the set of binary random schedules with equal probability to introduce the random alternatives of solutions. The second type was the copies of the non-optimized feasible schedule obtained from the MILP model (with starting fitness value of approximately 0.928) to aid the initial convergent phase of GA. The last type was the mixture of the previously defined types to provide the algorithm with the appropriate balance of pros and cons between both types. The additional detail of the experimental apparatus is described in Section 3.6.7. In this study, the population size ($N_{population}$) was fixed to 100 individuals in each generation. The default ratio between the non-optimized MILP solution and the binary random schedule was set to 0.75:0.25.

After constructing the first population's schedules, the translation step was performed to convert the initial population's format into the algorithm's eligible form, i.e., the chromosomal strings. The format of an individual in the initial population is an array; each row represents one staff's schedule, and each column represents each shift. For instance, in the May-June schedule of 2021, there were 71 staff in a 35-days schedule, each day with three possible shifts. Hence, the array of this schedule has the size of 71×105. To serve the GA's requirement, this array-like individual was reshaped into a strand of a chromosome where the head of the following nurse's schedule was attached to the tail of the prior nurse's schedule, as illustrated in Fig.3.1. Recalling the previous example, the chromosome size ($CHROMOSOME\_LENGTH$) becomes 7455×1. This process was repeated until all schedules were transformed into chromosomes.

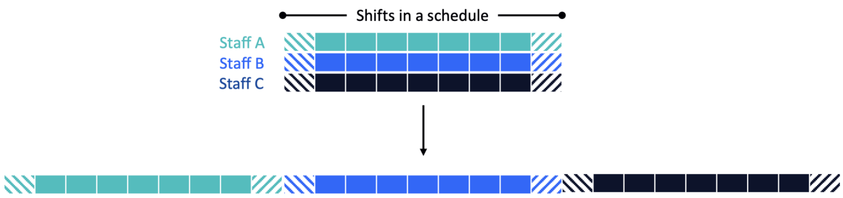

Figure 3.1: Translation process.

3.6.2 Decoding

A decoding process started with the transformation of each chromosome strand into a perceivable array. Since the binary encoding scheme was adopted with the one-bit-per-variable scheme, the decoding step on each bit yielded the binary state of a shift assignment. Following the initial decoding, each schedule array in the current generation was subjected to an evaluation process. The evaluation assayed the violation of constraints, as well as identified the value of all variables required to compute an objective function (z from Eq.3.9) based on the decoded state of a working shift. The evaluation algorithm is described in Section A.3, where each violation of constraint results in the addition of penalty (p) by 100. After finishing the evaluation, the value of the GA's objective function (z_{GA}) was assigned to each schedule as its fitness score using Eq.3.62. Unlike MILP in Eq.3.10, the GA's objective function was modified to be the maximization problem. This process was terminated once the fitness values had been assigned to all chromosomes in the generation.

Optimization objective:

$$Maximize(z_{GA}) = \frac{1}{z} + p \tag{3.62}$$

3.6.3 Tournament selection

Since the fitness score may experience some penalty, thus existing as a negative value, the tournament selection that suits this characteristic was chosen. There are two parameters involved in this process, i.e., a tournament size (N_{tnm}) and a tournament probability ($PROB_{tnm}$). The tournament size was fixed to 3. The tournament probability was set to 0.75. This selection process was conducted in pair until 50 pairs of chromosome were selected as parents for the next-generation population.

3.6.4 Crossover

The crossover process involves four parameters: the number of random crossover points (N_{xover}), a crossover probability ($PROB_{xover}$), a momentum-adjusted crossover probability ($PROB\_ADJ_{xover}$) and a switching probability ($PROB_{switch}$). The random crossover points and the switching probability were defined as 100 points and 0.2, respectively. Other parameters were varied during the tuning process as explained in Section 3.6.7. The crossover decisions were processed on all pairs of the selected parents until the process offspring (including non-crossed parents and crossed offspring) established the next-generation population of 100 individuals.

This study utilized the multi-point crossover with switching probability to increase the enhancement opportunity. Because the chromosomal length is extensive in this study, the switching probability of less than 0.5 is applied to reduce the stochastic degree. Additionally, the momentum-adjusted crossover probability was investigated as the extra approach to reduce the stochastic. The momentum faded the original crossover probability once the algorithm had approached half of the entire generations.

3.6.5 Mutation

The conventional mutation for binary encoding was adopted in this study. This process encloses two parameters, i.e., the mutation constant ($CONST_{mutate}$) and the momentum-adjusted mutation constant ($CONST\_ADJ_{mutate}$). The mutation constant was fixed to 3, and it was required to compute the mutation probability ($PROB_{mutate}$) as expressed in Eq.2.13. Additionally, once the algorithm had approached half of the whole generations, the momentum reduced the mutation constant, thus lowering the degree of stochastic. The presence of the momentum was a part of the parameter tuning process described in Section 3.6.7.

3.6.6 Elitism

The number of the fittest chromosome from the pre-crossover population ($N_{elitism}$) was selected as 15 copies. These copies replaced the first 15 individuals in the next-generation population. As a result, the highest fitness of the next generation remains greater than or equal to the previous generation.

3.6.7 Parameter tuning

As mentioned in the previous sections, some parameters were primarily fixed and did not participated in a tuning process. Those fixed parameters included $N_{generation}$, $N_{population}$, N_{tnm}, $PROB_{switch}$, $CONST_{mutate}$, and $N_{elitism}$. Therefore, the remaining parameters were subjected for the tuning process where the test parameters were compared with the default setting using only the May-June scheduling data. The judgement was based on the fitness values or z_{GA} that the test parameters contributed.

The tuning process classified into four tests, namely Test 1, Test 2, Test 3, and Test 4 as listed in Table 3.10. The tuning process ran consecutively starting from Test 1 and ending with Test 4, where each tuned parameters were repeated for five measurements. If the results in each test shows insignificant different with the default value, that default one will be selected as the winner.

Table 3.10: Parameter tuning scheme

Parameters	Default	Test 1	Test 2	Test 3	Test 4
$N_{generation}$	300	300	300	300	300
$N_{population}$	100	100	100	100	100
MILP solution : binary random	0.75 : 0.25	1.00 : 0.25, 0.75 : 0.25, 0.50 : 0.50, 0.25 : 0.75, 0.00 : 1.00	Test 1's winner	Test 1's winner	Test 1's winner
N_{tnm}	3	3	3	3	3
$PROB_{tnm}$	0.75	0.75	0.75	0.75	0.75
N_{xover}	100	100	100	100	100
$PROB_{xover}$	0.7	0.7	0.1,0.3,0.5,0.7,0.9	Test 2's winner	Test 2's winner
$PROB\_ADJ_{xover}$	-	-	-	Test 2's winner, half of Test 2's winner	Test 3's winner
$PROB_{switch}$	0.2	0.2	0.2	0.2	0.2
$CONST_{mutate}$	3	3	3	3	3
$CONST\_ADJ_{mutate}$	-	-	-	-	$3, 0.01$
$N_{elitism}$	15	15	15	15	15

Test 1 was responsible for the identification of the proper ratio between the non-optimized MILP solution and the binary random schedule, i.e., 1.00 : 0.25, 0.75 : 0.25, 0.50 : 0.50, 0.25 : 0.75, and 0.00 : 1.00. Other parameters apart from the tuned ratio remained as in the default setting. As shown in Table 3.11 and 3.12, Test 1 implied that the initial population with completely binary random genes yielded the minimum mean of z_{GA} with significance. However, the other mixing ratios between the non-optimized MILP solution and the binary random schedule did not show significant difference in their means (i.e., around 1). Therefore, the selected candidate from Test 1 was the default ratio of 0.75 : 0.25. In terms of the average running times, each sub-test took approximately 4800-5400 seconds to finish the algorithm.

Test 2 involved the tuning of crossover probability which were 0.1, 0.3, 0.5, 0.7, and 0.9. Other parameters remained the same as the default one, except the initial solution ratio whose value was set in accordance with the winner from Test 1. Test 2's results from Table 3.13 and 3.14 demonstrated that there was no significant difference in the means of z_{GA} (i.e., approximately 1) among all tested crossover probabilities. So, the default crossover probability of 0.7 was selected as the Test 2's winner. Additionally, their average running times were ranging from 4700-5300 seconds.

Test 3 was performed to identify the benefit of having momentum-adjusted crossover probability. In Test 3 setting, the first sub-test was the case without impeding momentum, and the second sub-test had the momentum adjustment exerted on the crossover probability to reduce it by halve. Other parameters had their values as specified in the default setting, except those winner parameters from Test 1 and Test 2. Table 3.15 and 3.16 showed the finding of Test 3 that the means of each sub-tests were not significantly different, and their means remained about 1 as the previous tests. Thus, the momentum adjustment on the crossover probability did not cause significant changes to the algorithm. The default setting of non-momentum adjustment on the crossover probability was proceeded as the candidate from Test 3. Besides, their average running times were comparable to Test 2.

Test 4 investigated the presence of the momentum-adjusted mutation constant. Its first sub-task inspected the case where mutation constant remain unchanged, while its second sub-tasks observed the setting where mutation constant was decreased by the momentum to 0.001. Similar to the prior tests, the remaining parameters except those winners from the previous tests remained as the default setting. From Table 3.17 and 3.18, Test 4 exhibited that the momentum-adjusted mutation constant significantly decreased the mean of z_{GA}. As a result, the default setting of non-momentum adjustment on the mutation constant was the winner of this test. Moreover, their average running times were approximately 4700 - 4800 seconds.

According to the results of all tuning tests, the final setting of parameters remained as the default one in Table 3.10.

Table 3.11: Statistical analysis of Test 1.

MILP solution : binary random	Mean of z_{GA}	Variance of z_{GA}	Mean of running time (s)
1.00:0.00	-248978.573873	98261345.85578860	4845.953424
0.75:0.25	1.010324	0.00035905	4872.280604
0.50:0.50	1.014476	0.00005637	4881.628854
0.25:0.75	1.022586	0.00016319	5267.536469
0.00:1.00	1.021572	0.00006046	5414.115283

Table 3.12: F-tests and t-tests between Test 1's parameters and the default setting.

MILP solution : binary random	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
1.00:0.25	602145332955.68400	0.00000	Yes	-56.16392	0.00000	Yes
0.75:0.25	2.20023	0.23190	No	-1.19982	0.26453	No
0.50:0.50	2.89497	0.16390	No	-1.22391	0.25581	No
0.25:0.75	1.00000	0.50000	No	0.00000	1.00000	No
0.00:1.00	2.69928	0.17972	No	-0.15158	0.88327	No

Table 3.13: Statistical analysis of Test 2.

$PROB_{xover}$	Mean of z_{GA}	Variance of z_{GA}	Mean of running time (s)
0.10	1.019348	0.00038595	5168.007981
0.30	1.019806	0.00026072	5231.832413
0.50	1.012388	0.00004371	5324.421193
0.70	1.012678	0.00055328	4719.319653
0.90	1.011577	0.00032537	4715.692095

Table 3.14: F-tests and t-tests between Test 2's parameters and the default setting.

$PROB_{xover}$	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
0.10	1.43355	0.36780	No	0.48663	0.63957	No
0.30	2.12216	0.24204	No	0.55863	0.59170	No
0.50	12.65934	0.01529	Yes	-0.02655	0.97992	No
0.70	1.00000	0.50000	No	0.00000	1.00000	No
0.90	1.70045	0.30983	No	-0.08302	0.93587	No

Table 3.15: Statistical analysis of Test 3.

$PROB\_ADJ_{xover}$	Mean of z_{GA}	Variance of z_{GA}	Mean of running time (s)
0.70	1.012678	0.00055328	4719.319653
0.35	1.022229	0.00002602	5268.469406

Table 3.16: F-tests and t-tests between Test 3's parameters and the default setting.

$PROB\_ADJ_{xover}$	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
0.70	1.00000	0.50000	No	0.00000	1.00000	No
0.35	21.26519	0.00587	Yes	0.88735	0.42099	No

Table 3.17: Statistical analysis of Test 4.

$CONST\_ADJ_{mutate}$	Mean of z_{GA}	Variance of z_{GA}	Mean of running time (s)
3.00	1.012678	0.00055328	4719.319653
0.01	0.952196	0.00117582	4806.018731

Table 3.18: F-tests and t-tests between Test 4's parameters and the default setting.

$CONST\_ADJ_{mutate}$	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
3.00	1.00000	0.50000	No	0.00000	1.00000	No
0.01	2.12518	0.24164	No	-3.25235	0.01166	Yes

3.7 Performance comparison

Performance comparisons were evaluated on the pairs of models using three metric classes: statistical features, model performance, and request satisfaction. All three classes contain multiple metrics with corresponding outcome characteristics, as tabulated in Table 3.19. The outcome characteristics of each metric yield one of the three possible consequences, i.e., deteriorated outcome, unchanged outcome, and optimized outcome, when comparing a particular model to its pair. For instance, if MILP provides a greater number of the minimum weekends off than GA, MILP will have an optimized outcome for this metric following the condition in Table 3.19. Moreover, the statistical features also evaluated each metric in the dimension of nurse's levels, i.e., Senior 1, Senior 2, Senior 3, Middle, Junior 1, Junior 2, and all levels. As a result, the statistical features contain 63 metrics (7 levels, each with 9 metrics). In total, each comparison has 76 metrics: 63 metrics from statistical features, 9 metrics from model performance, and 4 metrics from request satisfaction.

In this thesis, two comparisons were conducted to indicate the superior solution in the scheduling task. The first comparison was performed on two developed models (i.e., MILP and GA). The data set used in this assessment was from the May-June scheduling input. The second comparison was between the manual approach and the candidate model from the first comparison (i.e., the model that owns less deteriorated outcomes in the first comparison). This evaluation employed two data sets: the May-June scheduling input and the July-August scheduling input.

Additionally, the setting of objective weights in each scheduling data set is varied due to the different input characters (e.g., number of requests and number of staff). In May-June input, eight objectives' weights were set as 1, 0, 2, 1, 1, 3, 1 and 1, respectively. While in July-August data, the weights were established as 1, 0, 1.5, 1, 1.5, 1.5, 1 and 1, sequentially.

Table 3.19: Evaluation standard for performance comparison

		Outcome characteristics		
Metric classes	Metrics	Deteriorated	Unchanged	Optimized
Statistical features	Average total shifts	Significantly greater	Insignificantly different	Significantly lesser
	Average total working days	Significantly greater	Insignificantly different	Significantly lesser
	Average total morning shifts	Significantly lesser	Insignificantly different	Significantly greater
	Average total afternoon shifts	Significantly greater	Insignificantly different	Significantly lesser
	Average total night shifts	Significantly greater	Insignificantly different	Significantly lesser
	Average total double-shifts days	Significantly greater	Insignificantly different	Significantly lesser
	Average total weekends off	Significantly lesser	Insignificantly different	Significantly greater
	Average total off-bank quota transactions	Significantly greater	Insignificantly different	Significantly lesser
	Average total days off after night shifts	Significantly greater	Insignificantly different	Significantly lesser
Model performance	Maximum number of working shifts	Greater	Equivalent	Lesser
	Maximum number of double-shifts days	Greater	Equivalent	Lesser
	Minimum number of weekend offs	Lesser	Equivalent	Greater
	Maximum disapprovals for off requests	Greater	Equivalent	Lesser
	Maximum disapprovals for vacation requests	Greater	Equivalent	Lesser
	Maximum disapprovals for shift requests	Greater	Equivalent	Lesser
	Maximum disapprovals for not-to-be-assigned shift requests	Greater	Equivalent	Lesser
	Fitness score or z_{GA}	Lesser	Equivalent	Greater
	Scheduling time	Greater	Equivalent	Lesser
Request satisfaction	Approval percentage of total off requests	Lesser	Equivalent	Greater
	Approval percentage of total vacation requests	Lesser	Equivalent	Greater
	Approval percentage of total shift requests	Lesser	Equivalent	Greater
	Approval percentage of total not-to-be-assigned shift requests	Lesser	Equivalent	Greater

3.8 Statistical analysis

Apart from an absolute value and a percentage, the statistical measures included a mean, a variance, the maximum, the minimum, an F-test statistic, and a t-test statistic. The F-tests were used to verify the null hypothesis that the two population variances are equivalent, and the t-tests were responsible for testing the null hypothesis that the means of two populations are equal [46]. There were two types of t-tests adopted in this study, i.e., the student's t-tests and the Welch's t-test. The student's t-test was applied when F-test had implied an insignificant difference in variances [46]. The Welch's t-test was utilized when the F-test had indicated a significant difference [47]. All statistical tests used the p-value threshold of 0.05 to identify significance.

3.9 Development platform

The models and statistical analysis were developed on a virtual machine with CPU as processing unit. The virtual machine size was 2 cores, 7GB RAM, and 14 GB disk. All codes were written in Python using Jupyter Notebooks.

4 Results

4.1 Service blueprint mapping

Service blueprint mapping (Fig.4.1) was composed to provide the visual perception of the RN's scheduling process. The blueprint demonstrates the activities that each governed RN performed along the process. Apart from the RNs, five additional people were involved, i.e., on-duty-roster scheduler, ED head nurse, admin, OT-roster scheduler, and task-assigning scheduler.

The first action began with the RNs sending requests within one week for their desired on-duty schedules, such as shift requests, days off requests, and many more. They saved and updated their appeals via Google Sheet that the on-duty-roster scheduler had prepared. Typically, the requests open around one month before starting.

While waiting for the publication of the monthly on-duty schedule, the primary onstage task was the scheduling, which took around 3-5 days to complete. The on-duty-roster scheduler constituted the schedule by considering requests and multiple constraints. The scheduler also relied on the mathematics functions in Google Sheets (like the summation and conditional statements) to facilitate the scheduling. Additionally, the backstage action required the scheduler to manage the off-bank quota. Like the vacation quota, the off-bank quota informed the additional days off each RN could have (more detail on these terms were enclosed in Section 4.2). Once the scheduler finished scheduling, the ER head nurse would inspect the result and manage the vacation quota. The instances of vacation management included the scenarios when some nurses forgot to use their quota whose expiration was approaching. Furthermore, the admin assisted this step by notifying the current vacation quota to the ER head nurse. Both managements for day-of bank and vacation utilized the records, which were manually documented in Google Sheets. Before publishing the schedule, the scheduler had to assign staff for the in-charge nurse and shift-transferring roles.

After RNs received the monthly on-duty schedule, another format (i.e., weekly table) was supplied to the RNs. This format contained more readability than the monthly one. In onstage actions, the admin was responsible for composing the weekly on-duty schedule.

When the on-duty schedules in all formats were issued, RNs had the right to exchange shifts. Before submitting any exchange request to the drop-off box, the RN had to find another qualified staff that they would trade. Additional detail on shift exchange rule was explained in Section 4.2. The admin would gather all requests and perform daily adjustments on all published schedules. Depending on the availability and agreement, the exchange process could take more than one day to finish. Also, this exchange process did not require any approval since the parties were assumed to have credits and responsibility to perform the eligible exchange.

Following the on-duty shifts, RNs could sign up for the OT slots as well. The onstage action in this step allowed RNs to submit OT requests via Google Sheets that the OT scheduler had provided. The request period opened approximately five days. Therefore, OT was optional, and each nurse was not forced to do it unwillingly. Following the end of the request period, RNs would receive the OT schedule within 1-2 days. These durations covered the onstage action of the OT scheduler, whose task was to schedule OT by considering the balance of approvals. In other words, requestors would receive comparable numbers of OT.

If RNs did not want some OT slots, they can either exchange or give them away to their agreed parties. This action was possible since OT shifts were not just exchangeable but passable too. The process was similar to the on-duty shift exchange. Only one difference was that OT could be exchanged with anyone regardless of staff's levels.

In addition to the schedules for on-duty and OT, RNs acquired the tasking table at least one day before starting each weekly schedule. Therefore, the task-assigning scheduler was needed to allocate tasks during that previous week. After the weekly tasking table had been published, staff would only have to work according to their schedule. Additionally, RNs still could exchange on-duty shifts as long as that shift was not commencing yet.

As stated in this project's scope, on-duty scheduling was solely the optimization model's target. Hence, only the first three customer actions (in Fig.4.1) were involved in the in-depth insights extraction and the model formulation.

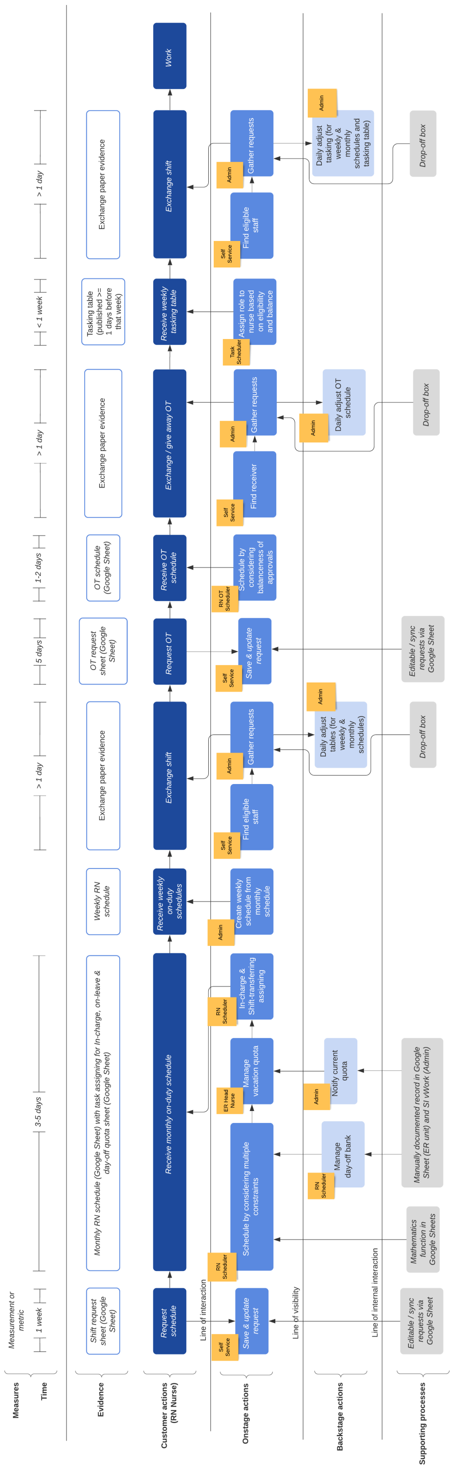

Figure 4.1: Service blueprint of RN's scheduling process.

4.2 In-depth insights extraction

The results of in-depth insights extraction were organised into four dimensions: considered elements, pains, satisfaction, and to-be preferences. These findings served their purpose as the development foundation for the user-centred solution.

4.2.1 Considered elements

The considered elements explained the background of the scheduling system. They were listed into two following viewpoints: a context of scheduling and a scheduling policy.

4.2.1.1 Context of scheduling

The setting for the RN's schedule covered five weeks or 35 days. Each day was composed of three non-overlapping shifts: morning (7:00 to 15:00), afternoon (15:00 to 23:00), and night (23:00-7:00). RN's levels were sorted into the following hierarchy: Senior 1, Senior 2, Middle, Junior 1, and Junior 2. Senior 1 was the highest rank who can work in any position, while Junior 2 had the fewest experience. Each shift may not contain only ED employees, for some slots received the extra staff from other departments.

There were three classes of absent days in the scheduling system, i.e., a regular off, a VIP off, and a vacation. The conventional one contained two sub-types, which are a weekly-off allowance and off-bank quota.

The total number of weekly days off emerged from the summation of weekends and holidays in that schedule. In other words, the number represented the base of absent days that each nurse equally obtained in each table.

The off-bank quota was established with a similar concept as OT. Instead of earning extra wages in return, each shift greater than the on-duty threshold yielded one additional day in the so-called off-bank quota. For instance, if a nurse had worked 25 shifts, and the entire on-duty days was 22 shifts in that schedule, then the nurse received three days off to the bank.

The VIP absence was designated for some staff with the following criteria: an employee who was about to resign but still had not used all off quotas, a nurse who temporarily left for studying, and a mother who recently gave birth. These eligible people could not work on some days or some periods. Thus, they needed a VIP status to guarantee that their requests would be approved.

The vacation was the annual leaves that each nurse could use. Each year, everyone received 15 days of vacations, and these quotas were reset yearly. Thus, nurses must use all of their allowances before expiring. This requirement explained the different priorities among absence classes, especially the expirable type.

There were eight kinds of requests allowed in the scheduling systems, where four of them were normal, and the remaining fours were VIP. The normal category covered a request for a day away (the combination of the weekly day off and the off-bank quota), a vacation, a shift, and a not-to-be-assigned shift. Furthermore, the VIP category enclosed a must-approve demand for a day off, a working shift, and a not-to-be-assigned shift. Like the VIP off request, some nurses could earn VIP status for asking shift and not-to-be-assigned shift due to specific reasons, such as the health problem and the accommodation distance.

4.2.1.2 Scheduling policy

RN's scheduling regulation consisted of ten policies. The first one was the weekly working shifts and days policy. A nurse must not work either at the ED and other units beyond six shifts or five working days each week. A nurse should not unwillingly and continuously operate within the department greater than six days in a row.

Likewise, the afternoon shift policy stated that each nurse should have consecutive afternoon shifts at a maximum of two unless inquired.

The night shift rule allowed a nurse to work a maximum of three-night shifts each week. A senior nurse and a non-senior nurse must not have connected night shifts greater than two and three, respectively. This requirement also applied to the period between a previous schedule (i.e., last weekend) and a current one. Additionally, the night shift policy could be lifted if a nurse had voluntarily requested against the restriction.

The consecutive shifts restraint did not authorise the following pattern at all cost, i.e., afternoon-and-night shifts and night-and-next-morning shifts. However, afternoon-and-next-morning shifts and night-and-next-afternoon shifts were possible if the nurse had explicitly asked for them. If not, the scheduler would not assign staff with those patterns.

Apart from the ED's operation shifts, the total shifts policy included the out-of-unit positions, the vacations, and the usage of the off-bank quota as working hours.

As addressed in Table A.1 and A.2, the staffing rule adopted two criteria based on day types, i.e., working day and weekend or holiday. The regulation allowed the higher-level nurses to work in a lower hierarchy, but not the other way around. Moreover, the staffing rule was also related to the RN's shift exchange practice. The level of the requestee should be applicable to work in the requestor's position. For example, if the requestor had been assigned to the class of Senior 3, then the requestee must only be either Senior 3 or higher. Additionally, the other department would provide extra staff to help the ED in Junior 2's positions.

The weekend off policy declared that each nurse in all levels must earn at least one weekend off (both Saturday and Sunday) in the schedule.

According to the working preference-related standard, Senior 1 would work morning shifts only, while Senior 2 would not be assigned to any night shift. This standard could be paused if the nurses had asked to work in those prohibited slots.

The out-of-unit policy applied to the nurses that had been scheduled to work in other units. The policy forced all out-of-unit requests to be admitted.

The VIP request regulation provided extra privileges to some staff. With that status entitled, their requests (i.e., working shifts, not-to-be-assigned shifts, and days off) must be approved.

The shifts' limitation rule restrained the number of shift types to all nurses. A nurse shall not have excessive afternoon and night shifts; each must be within 33% of on-duty days in that schedule.

4.2.2 As-is pains

Request satisfaction was alleged by the shift managers as the primary source of difficulty in scheduling. When the number of applications was high, the scheduler would take longer to process while satisfying those demands. When not all requests could be granted simultaneously, the calls from junior nurses got denied first. Nevertheless, this scenario did not happen very often. If it did, the affected requestor usually do shift exchange, which was time-consuming and burdensome to find the eligible requestee. Indeed, this shift exchange happened because the requestor could not work in that slot, which was why the appeal was submitted in the first place. The governed staff further commented that they performed ad-hoc shift exchange approximately five times per schedule. Therefore, they did not want to do an extra trade process, if not necessary.

The scheduler's goal of balancing shifts was achievable, yet it was complicated to manage. Junior nurses frequently ended up having more night shifts than other levels. Furthermore, the management stakeholders stated that it was hard to limit the number of nurse's working shifts within the on-duty limits. So, each nurse would gain quotas to the days-off bank. This continuous addition of days-off bank was likewise a problem in the managers' aspect. Also, the scheduler could not allocate two weekends off to all, particularly junior nurses who received only one weekend off.

In the staffing aspect, the managers encountered inadequate staff on some days. It was because some nurses had been assigned for out-of-unit positions, and some were off or on vacation. Then the available nurses on that day would have to work double shifts. Hence, It was rare for the managers to reach the aim of zero double-shifts days, and usually ended up with 1-2 double shifts per schedule instead. Alternatively, the scheduler might assign less staff than the criteria to prevent double-shifts assignments. Either way, the consequence corrupted staff's efficacy by introducing exhaustion.

4.2.3 As-is satisfaction

Both managers and governed staff were satisfied with the balance result of shifts arrangement within the same generation. Each nurse under the same level would receive an equivalent number of shifts, including total shifts, morning shifts, afternoon shifts, and night shifts. All stakeholders were pleased with the approval percentage of staff's requests. Particularly the requests for days off or vacations, all of them were usually granted. Besides, the nurses did not mind working beyond the on-duty threshold since they were compensated with additional days in the off-bank quota.

4.2.4 To-be preference

Request satisfaction was the topmost priority in scheduling. All stakeholders wished that staff's requests should be approved as much as possible regardless of seniority. The shift managers highlighted that each request type needed different precedence, especially the vacations request whose usable period is annual. Thus, this expirable quota would have a higher priority than the regular days off credit.

In terms of the arrangement of the shifts, all parties agreed that nurses should work relatively close to the number of on-duty days presented in each month schedule. The shift managers strongly agreed with this goal to resolve the excessive formation of days-off bank. Moreover, the management stakeholders yearned the balance within each shift type (i.e., morning, afternoon, and night) and the total number of shifts. Significantly, each nurse should minimally receive a comparable number of afternoon shifts and night shifts. Additionally, the aim to reduce the unwilling double-shifts days was likewise vital, especially in junior nurses. Sometimes, the double-shifts days were unavoidable because of the inadequate staff. However, these extra burdens should be allotted to each nurse in an equal manner.

The days off allocation could add time values to staff. Because the nurse's working time is dynamic and may not align well with other people (e.g., friends and family), the managers desired to provide all nurses with two weekends off in each schedule.

Staffing fulfilment should be reached in every shift. The governed nurses emphasised that this criterion would decrease the workload and prevent the probability of having demand-supply unmatched.

4.3 Performance comparison

The metrics comparisons were first conducted on the pair of MILP and GA to identify the superior candidate. Then, the schedule optimized by the superior model (i.e., MILP) and the manual outcome were collated to find the pros and cons of each approach.

4.3.1 The collation of MILP-optimized and GA-optimized schedule

The statistical features of the MILP-optimized schedule and the GA-optimized schedule in the May-June period were tabulated in Table 4.3 and 4.1, respectively. The F-tests and the t-tests were conducted to imply the significance of statistical findings. Their results were attached in Section A.4. Fig. 4.2 shows the significant changes of the statistical features in the MILP-optimized schedule in relation to the GA-optimized results. According to Fig. 4.2, 45 features were insignificantly different among the two optimization approaches. However, there were three deteriorated features that MILP could not overcome GA, i.e., the increase in average working days (by 20% in Senior 1 and 11% in all levels) and the 580% surplus of the Middle's off-bank quota transaction. Nevertheless, 15 MILP's features were found to be significantly optimized when comparing to GA's results. The mean morning shifts of Senior 1 was raised by 20%. The average double-shifts days in all levels and individual levels (from Senior 2 to Junior 2) were significantly reduced by

54%-84%. The expected weekends off in all levels and the particular levels starting from Senior 2 to Junior 1 were extended by 40%-83%. The average off-bank quota transaction of Junior 1 was lower by100%. Additionally, the average days off after night shifts were significantly decreased by 32% and 24% in Middle and all levels, orderly.

Table 4.4 affirmed that all nine MILP's metrics were much more optimized than GA in terms of the model's performance. The GA's fitness score, which correlates to the overall performance, was also 4.6 times lower than MILP. Besides, the GA's computation time was about three times slower.

Fig. 4.5 demonstrated that GA's capability was inferior to MILP in all four metrics concerning request satisfaction. GA yielded the highest approval percentage of approximately 60%, while most request types received less than 40% approvals. In contrast, the minimum approval rates of MILP was almost as much as 96%.

According to the evaluation of 76 metrics, MILP showed 59.21% with unchanged outcomes, 36.84% with optimized consequences, and only 3.95% with deteriorated results compared with GA. Therefore, MILP was claimed to be more superior model to GA.

Table 4.1: Statistical features of the GA-optimized schedule of May-June period.

	Senior 1	Senior 2	Senior 3	Middle	Junior 1	Junior 2	All levels
Average shifts ± SD	20.167 ± 4.298	22.111 ± 5.858	20.333 ± 4.061	19.526 ± 4.5	23.214 ± 4.491	21.75 ± 8.452	21.056 ± 5.363
Average working days ± SD	18.833 ± 1.951	19.0 ± 4.472	18.0 ± 3.54	16.737 ± 3.416	18.929 ± 3.453	16.25 ± 6.398	17.845 ± 4.086
Average morning shifts ± SD	18.833 ± 1.951	13.889 ± 4.932	5.6 ± 2.245	5.947 ± 2.038	9.286 ± 2.185	10.125 ± 4.567	9.099 ± 5.016
Average afternoon shifts ± SD	1.333 ± 2.981	8.222 ± 1.227	6.867 ± 2.247	6.737 ± 2.048	7.214 ± 1.612	6.0 ± 2.828	6.507 ± 2.711
Average night shifts ± SD	0.0 ± 0.0	0.0 ± 0.0	7.867 ± 2.363	6.842 ± 1.954	6.714 ± 2.519	5.625 ± 2.955	5.451 ± 3.575
Average double-shifts days ± SD	1.333 ± 2.981	3.111 ± 1.728	2.333 ± 0.869	2.789 ± 1.734	4.286 ± 1.708	5.5 ± 2.55	3.211 ± 2.175
Average weekends off ± SD	1.333 ± 0.943	1.333 ± 0.471	1.2 ± 0.542	1.368 ± 0.581	1.429 ± 0.623	1.875 ± 1.364	1.394 ± 0.76
Average off-bank quota transactions ± SD	-1.667 ± 4.23	2.556 ± 2.266	0.4 ± 1.925	0.263 ± 2.825	2.571 ± 3.375	-0.25 ± 8.452	0.817 ± 4.14
Average days off after night shifts ± SD	0.0 ± 0.0	0.0 ± 0.0	6.0 ± 2.033	5.158 ± 1.565	4.786 ± 1.859	4.375 ± 2.342	4.085 ± 2.746

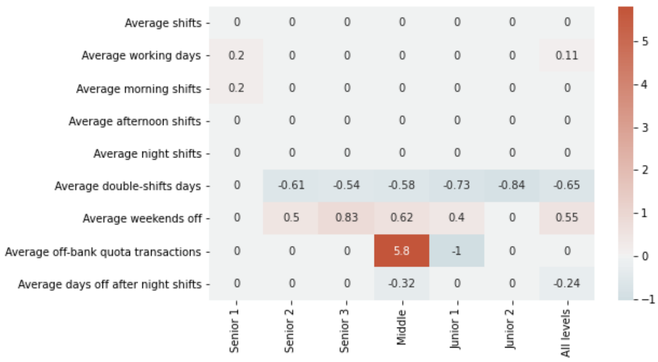

Figure 4.2: Heat map representing the significant changes (in percentage) of the MILP-optimized schedule in relation to the GA-optimized schedule.

4.3.2 The collation of manual and MILP-optimized schedule

4.3.2.1 May-June schedules

Table 4.3 and 4.3 contain the statistical features of May-June schedules created by the manual approach and MILP, respectively. The significance tests (F-tests and t-tests) were applied with results presented in Section A.4. The heatmap in Fig.4.3 illustrates the significant changes in the features of the MILP-optimized schedule when compared with the manual one. There were 51 insignificantly different metrics between the two methods. Nevertheless, four features possessed significantly deteriorated characteristics. The MILP's average working days in all levels were lifted by 8.3% relative to the manual approach. The average night shifts and the average days off after night shifts of Senior 3 were 47% and 66% higher in MILP. Furthermore, the MILP's average off-bank quota transactions in Senior 1 were 250% greater than the manual schedule. Regardless of the downsides, MILP provided eight features with significantly optimized outcomes. The average afternoon shifts of Senior 3 and the average night shifts of Junior 1 were subsided by 30% and 22% in MILP, sequentially. The MILP's average double-shifts days were declined by 54-71% in all levels and individual levels (from Senior 3 to Junior 2). Moreover, the average weekend offs in Junior 1 were raised by 65% in MILP.

For the model's performance (Table 4.4), there were eight metrics that MILP superiorly optimized the outcomes. Especially the fitness score, the manual approach possessed a negative value, which implies constraint violations. The further inspection indicated that the manual approach conducted 139 violations. Fig. 4.4 visualized the proportion of violation types that existed in the manual schedule. The 85% of the infringement were dominated by these top four entities, i.e., the violations in the consecutive shifts policy (40%), the staffing policy (19%), the shifts' limitation policy (14%), and the afternoon shift policy (12%). By disregarding the penalty in the manual approach's fitness score (where one penalty resulted in the penalty of -100), it would yield 2.266. This violation-excluded score was still approximately two times lesser than the MILP. Additionally, MILP performed the scheduling task about 12.5 times faster than the manual scheduling. Apart from the MILP's superior outcomes, the manual approach and MILP equally granted all vacation requests.

Both MILP and the manual approach catered 100% approvals on vacations in terms of request satisfaction. However, MILP earned lower approval rates compared to the manual schedule in not-to-be-assigned shift requests. Nonetheless, MILP provided perfect approval percentages of 100% for shift requests and off requests, while the manual approach strove 95.395% and 99.544% consecutively.

Out of 76 evaluated indicators, both approaches owned 53% with unchanged consequences. Although MILP had 6.58% with deteriorated outcomes, it gave 23.68% with optimized results collated with manual scheduling.

Table 4.2: Statistical features of the manual schedule of May-June period.

	Senior 1	Senior 2	Senior 3	Middle	Junior 1	Junior 2	All levels
Average shifts ± SD	22.333 ± 0.943	20.0 ± 5.437	20.267 ± 3.511	20.684 ± 4.305	21.429 ± 3.793	21.125 ± 8.085	20.845 ± 4.692
Average working days ± SD	20.833 ± 2.672	18.444 ± 4.901	17.933 ± 3.275	18.0 ± 3.907	18.143 ± 3.583	18.125 ± 7.008	18.324 ± 4.321
Average morning shifts ± SD	20.5 ± 3.403	12.778 ± 4.825	5.933 ± 1.948	5.947 ± 2.114	8.571 ± 3.678	9.875 ± 3.822	9.0 ± 5.276
Average afternoon shifts ± SD	1.833 ± 4.099	6.778 ± 3.966	9.133 ± 2.156	6.579 ± 2.255	4.5 ± 1.722	7.375 ± 2.87	6.423 ± 3.385
Average night shifts ± SD	0.0 ± 0.0	0.444 ± 1.257	5.2 ± 1.796	8.158 ± 1.598	8.357 ± 2.635	3.875 ± 1.536	5.423 ± 3.579
Average double-shifts days ± SD	1.5 ± 3.354	1.556 ± 1.257	2.333 ± 1.35	2.684 ± 0.862	3.286 ± 1.097	3.0 ± 1.225	2.521 ± 1.582
Average weekends off ± SD	2.667 ± 1.795	2.222 ± 1.03	2.133 ± 0.806	2.0 ± 0.725	1.214 ± 0.558	1.5 ± 1.323	1.901 ± 1.064
Average off-bank quota transactions ± SD	0.667 ± 0.745	0.444 ± 1.343	0.467 ± 2.093	1.895 ± 1.552	1.0 ± 1.964	-0.875 ± 8.085	0.817 ± 3.264
Average days off after night shifts ± SD	0.0 ± 0.0	0.222 ± 0.629	2.933 ± 1.181	4.316 ± 1.416	4.571 ± 1.591	2.5 ± 1.225	2.986 ± 2.066

Table 4.3: Statistical features of the MILP-optimized schedule of May-June period.

	Senior 1	Senior 2	Senior 3	Middle	Junior 1	Junior 2	All levels
Average shifts ± SD	24.0 ± 1.414	21.111 ± 5.28	20.6 ± 3.684	20.579 ± 4.546	20.357 ± 3.734	21.25 ± 8.058	20.972 ± 4.806
Average working days ± SD	22.667 ± 2.867	19.889 ± 5.238	19.533 ± 3.685	19.421 ± 4.476	19.214 ± 3.707	20.375 ± 7.729	19.845 ± 4.764
Average morning shifts ± SD	22.667 ± 2.867	12.444 ± 4.4	6.533 ± 2.446	6.579 ± 2.908	7.857 ± 3.248	8.0 ± 3.428	9.085 ± 5.53
Average afternoon shifts ± SD	1.333 ± 2.981	8.667 ± 0.943	6.4 ± 2.154	7.211 ± 2.462	6.0 ± 2.236	7.0 ± 2.784	6.465 ± 2.896
Average night shifts ± SD	0.0 ± 0.0	0.0 ± 0.0	7.667 ± 2.44	6.789 ± 2.546	6.5 ± 1.88	6.25 ± 2.681	5.423 ± 3.547
Average double-shifts days ± SD	1.333 ± 2.981	1.222 ± 0.629	1.067 ± 0.249	1.158 ± 0.67	1.143 ± 0.515	0.875 ± 0.331	1.127 ± 1.006
Average weekends off ± SD	2.167 ± 1.462	2.0 ± 0.0	2.2 ± 0.748	2.211 ± 0.408	2.0 ± 0.378	2.375 ± 0.992	2.155 ± 0.705
Average off-bank quota transactions ± SD	2.333 ± 1.106	1.556 ± 1.771	0.8 ± 1.973	1.789 ± 1.196	-0.071 ± 2.219	-0.75 ± 8.058	0.944 ± 3.306
Average days off after night shifts ± SD	0.0 ± 0.0	0.0 ± 0.0	4.867 ± 1.784	3.526 ± 0.939	3.5 ± 1.296	3.75 ± 1.714	3.085 ± 2.095

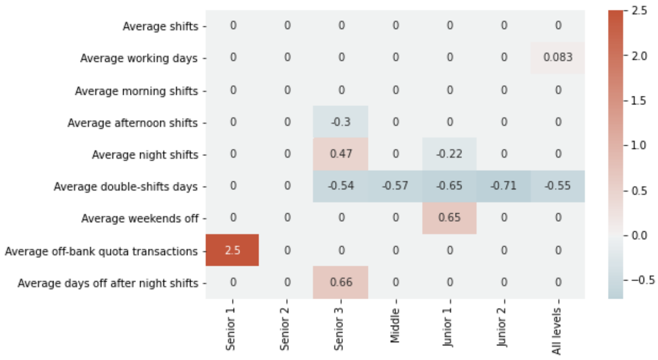

Figure 4.3: Heat map representing the significant changes (in percentage) of the MILP-optimized schedule in relation to the manual schedule of May-June period.

Table 4.4: The values of decision variables computed from the manual schedule, MILP optimization, and GA optimization in May-June schedule.

	Manual	MILP	GA
Maximum number of working shifts	28	25	29
Maximum number of double-shifts days	4	1	8
Minimum number of weekend offs	1	2	1
Maximum disapprovals for off requests	2	0	10
Maximum disapprovals for vacation requests	0	0	4
Maximum disapprovals for shift requests	4	0	13
Maximum disapprovals for not-to-be-assigned shift requests	5	1	16
Fitness score or z_{GA}	-13897.734	4.547	0.992
Scheduling time (minutes)	240 - 360	24	76

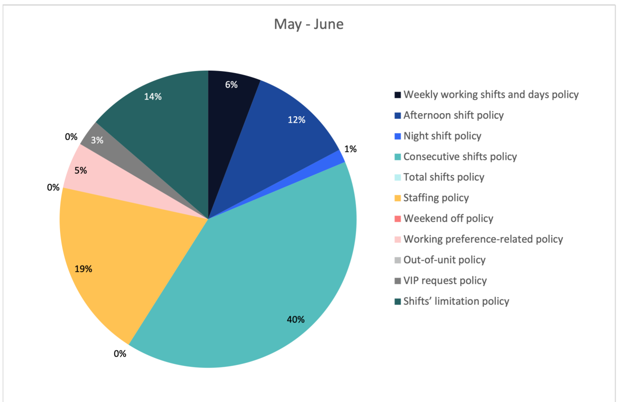

Figure 4.4: Policy violations of manual schedule in May-June period.

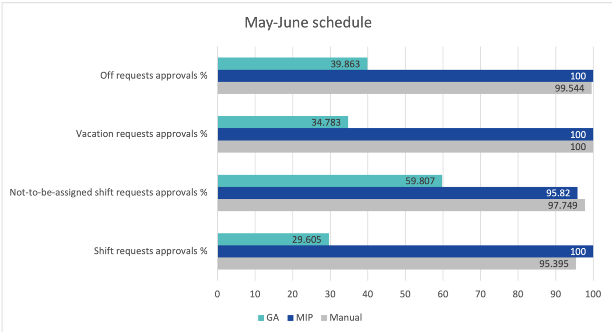

Figure 4.5: Request approvals percentage among all types of schedule in May-June period.

4.3.2.2 July-August schedules

The statistical characteristics of the July-August schedule prepared by the manual method and MILP were listed in Table 4.5 and 4.6, respectively. The test results for statistical significance were presented in Section A.4. Fig. 4.6 exhibited the significant changes of each MILP's features in collation to the manual approach. Forty-eight traits were tested to be in-significantly different in both solutions. Nevertheless, there were 5 MILP's features with deteriorated results. The MILP's average working days were 14%, 7.7%, and 9.1% greater in Senior 3, Junior 2, and all levels. The average night shifts and the average days off after night shifts of Senior 3 were 73% and 79% higher in MILP. Apart from these adverse changes, MILP owned ten features with significantly optimized updates. In MILP, the average total shifts, the average night shifts, the average off-bank quota transactions, and the average days off after night shifts were declined by 4.4%, 25%, 36%, and 27%, respectively in Junior 2. The MILP's average double-shifts days in each level (starting from Senior 3 to Junior 2) and all levels were attenuated by 55-74%. Also, the average weekends off in Junior 2 was raised by 100% in MILP.

Table 4.7 described the model's performance between the manual scheduling and MILP. In MILP, 7 out of 9 metrics held more optimized results than the manual method. Similar to the May-June period, the fitness score was negative in the manual case. Additional diagnostic indicated that it violated 78 constraints. Fig. 4.7 illustrated that 85% of the infraction were caused by the top-4 violations of consecutive shifts policy (41%), staffing policy (25%), afternoon shift policy (10%), and weekly working shifts and days policy(9%). By neglecting the penalty values, its score would be 2.253, which was still two times slower than MILP's score. Also, the computation duration of MILP was around 8.5 times faster than the manual scheduling. Nevertheless, the manual scheduling overcame MILP only in the zero disapproval of off requests. Besides, both methods equally gave zero rejection of vacations.

As shown in Fig. 4.8, MILP delivered more excellent approval rates than the manual one, with 92.771% in shift requests and 100% in not-to-be-assigned shifts requests. Both methods granted 100% approvals for vacation. Nonetheless, the manual scheduling superiorly gave 100% satisfaction to off requests, while MILP provided 87.224%.

In the total of 76 evaluations, all methods provided 65.79% of unchanged results. Although 9.21% of MILP's metrics offered the deteriorated outcomes, MILP contributed 25% of more optimized consequences than the manual scheduling.

Table 4.5: Statistical features of the manual schedule of July-August period.

	Senior 1	Senior 2	Senior 3	Middle	Junior 1	Junior 2	All levels
Average shifts ± SD	19.833 ± 3.484	22.444 ± 2.006	20.867 ± 3.423	20.684 ± 6.358	21.154 ± 6.926	26.143 ± 0.35	21.522 ± 5.21
Average working days ± SD	18.333 ± 3.3	21.111 ± 1.663	18.333 ± 3.155	17.526 ± 5.716	17.692 ± 6.043	22.286 ± 0.452	18.754 ± 4.695
Average morning shifts ± SD	18.167 ± 3.484	14.667 ± 4.397	8.467 ± 2.418	6.684 ± 4.702	6.308 ± 2.462	9.286 ± 1.03	9.304 ± 5.148
Average afternoon shifts ± SD	1.667 ± 3.727	7.444 ± 4.349	7.667 ± 1.491	6.632 ± 2.738	6.846 ± 2.537	7.143 ± 0.99	6.623 ± 3.167
Average night shifts ± SD	0.0 ± 0.0	0.333 ± 0.943	4.733 ± 1.879	7.368 ± 2.96	8.0 ± 2.66	9.714 ± 0.452	5.594 ± 3.85
Average double-shifts days ± SD	1.5 ± 3.354	1.333 ± 0.816	2.533 ± 1.147	3.158 ± 1.348	3.462 ± 1.216	3.857 ± 0.35	2.768 ± 1.678
Average weekends off ± SD	2.667 ± 1.795	1.556 ± 0.685	2.333 ± 0.869	2.0 ± 1.124	1.615 ± 1.077	1.0 ± 0.0	1.899 ± 1.131
Average off-bank quota transactions ± SD	-1.5 ± 2.814	0.0 ± 1.944	-0.4 ± 2.215	1.737 ± 2.022	1.154 ± 7.102	3.143 ± 0.35	0.797 ± 3.817
Average days off after night shifts ± SD	0.0 ± 0.0	0.0 ± 0.0	2.867 ± 1.204	4.0 ± 1.556	4.462 ± 1.599	5.714 ± 0.452	3.145 ± 2.202

Table 4.6: Statistical features of the MILP-optimized schedule of July-August period.

	Senior 1	Senior 2	Senior 3	Middle	Junior 1	Junior 2	All levels
Average shifts ± SD	21.167 ± 2.911	23.444 ± 1.892	22.067 ± 2.954	20.211 ± 6.031	19.846 ± 6.96	25.0 ± 0.0	21.536 ± 5.003
Average working days ± SD	19.667 ± 3.35	22.444 ± 1.892	20.933 ± 3.172	19.263 ± 6.12	18.769 ± 6.784	24.0 ± 0.0	20.464 ± 5.044
Average morning shifts ± SD	19.667 ± 3.35	14.556 ± 2.362	7.067 ± 1.806	7.105 ± 4.876	6.769 ± 3.285	9.143 ± 2.587	9.304 ± 5.314
Average afternoon shifts ± SD	1.5 ± 3.354	8.889 ± 1.197	6.8 ± 2.315	6.632 ± 4.771	6.154 ± 2.656	8.571 ± 1.678	6.623 ± 3.687
Average night shifts ± SD	0.0 ± 0.0	0.0 ± 0.0	8.2 ± 2.535	6.474 ± 3.662	6.923 ± 2.868	7.286 ± 2.119	5.609 ± 4.026
Average double-shifts days ± SD	1.5 ± 3.354	1.0 ± 0.0	1.133 ± 0.34	0.947 ± 0.394	1.077 ± 0.615	1.0 ± 0.0	1.072 ± 1.068
Average weekends off ± SD	2.0 ± 1.155	2.0 ± 0.0	2.067 ± 0.249	2.105 ± 0.718	2.154 ± 0.863	2.0 ± 0.0	2.072 ± 0.644
Average off-bank quota transactions ± SD	-0.167 ± 2.115	1.0 ± 1.491	0.8 ± 1.939	1.263 ± 1.207	-0.154 ± 6.62	2.0 ± 0.0	0.812 ± 3.258
Average days off after night shifts ± SD	0.0 ± 0.0	0.0 ± 0.0	5.133 ± 1.628	3.789 ± 2.041	3.538 ± 1.646	4.143 ± 1.457	3.246 ± 2.386

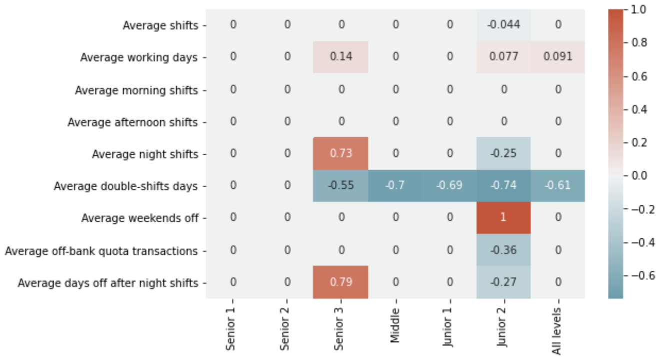

Figure 4.6: Heat map representing the significant changes (in percentage) of the MILP-optimized schedule in relation to the manual schedule of July-August period.

Table 4.7: The values of decision variables computed from the manual schedule, MILP optimization, and GA optimization in July-August schedule.

	Manual	MILP
Maximum number of working shifts	29	25
Maximum number of double-shifts days	5	1
Minimum number of weekend offs	0	2
Maximum disapprovals for off requests	0	1
Maximum disapprovals for vacation requests	0	0
Maximum disapprovals for shift requests	12	1
Maximum disapprovals for not-to-be-assigned shift requests	12	0
Fitness score or z_{GA}	-7797.747	4.659
Scheduling time (minutes)	240 - 360	35

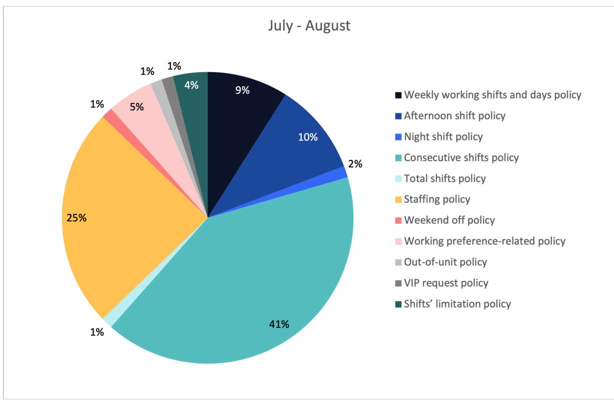

Figure 4.7: Policy violations of manual schedule in July-August period.

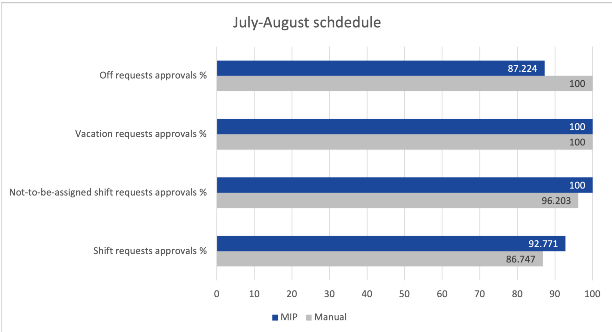

Figure 4.8: Request approvals percentage among all types of schedule in July-August period.

5 Discussion

5.1 Service blueprint mapping

According to the service blueprint, the bottleneck activity of the whole scheduling system is found to be on-duty scheduling, where nurses are waiting for the publication of the working schedule. With only one activity, at least three management people (i.e., on-duty-roster scheduler, ED head nurse, and admin) are involved in both onstage and backstage actions. Moreover, the process can take at least three days to complete. In other words, it is the process that consumes the vastest resource of the management party in terms of time and the number of participating staff.

Because the scheduler must consider the working policy and more than 1000 requests from approximately 70 nurses (according to the scheduling inputs from Section 3.4), the complexity in scheduling is the leading cause of time consumption.

Additionally, the assistive tool (i.e., mathematics functions in Google Sheets) for scheduling is not effective enough to ease the complexity of the process. Even though it can facilitate staffing fulfilment in each shift, the scheduler still executes most shift arrangements. Moreover, the tool is usually incapable of notifying the violation of working constraints, thus raising the chance of having a schedule with policy infringement. This finding highlights the need for a more effective scheduling tool.

5.2 In-depth insights extraction

For undertaking user's requirements, pains, and preferences and, maintaining current satisfactions, most of the issues are incorporated into the models, either in parameters, constraints or the objective function. The optimization objectives subsidize work stress while promoting request approvals and fairness among all nurses. Additionally, from the inspection of scheduling policy, one prominent policy plays a vital role in enhancing the shift balance. That policy is the shifts' limitation rule. With this policy incorporated in the model's constraints, the shift balance features can be easily acquired without significantly expanding the model's size.

In terms of the scheduling elements, the model is designed to intake three request types for the absent days (i.e., a regular off, a VIP off, and a vacation), rather than merging into one single type. The main reason is that each type has a different priority. For example, the VIP off request is a must-approve type; the vacation is the class preferred to be approved more than the regular one due to its expirable usage. Likewise, different priorities reasoning applies for VIP and a regular class of other requests, i.e., working shift and not-to-be-assigned shift requests. Therefore, it is an unavoidable trade-off between satisfying user requirements and limiting the size of the model.

Nevertheless, the complete solution for resolving all user's requirements needs the cooperation between technology and the hospital, especially in the staffing and the off-bank quota. According to the definition of the off-bank quota, it is shown to be an alternative solution that the hospital applies to compensate for extra working shifts. There are two possible implications regarding the increment of the off-bank quota. One is that the scheduler does not limit the number of unavailable staff in each shift. As a result, nurses may simultaneously ask for many days off in that schedule, causing less available staff to be scheduled. So, some remaining nurses may have to work extra to cover for the insufficient staff. The second implication is that ED currently has insufficient staff that do not align with the staffing criteria. Therefore, when the schedule is created following the staffing requirements, some nurses usually have extra shifts. As a result, the off-bank quota problem correlates to unmatched staffing. These issues can be effectively resolved through the hospital's management, such as the procurement of extra staff and the additional policy determining quota for staff requests in each shift. The staff procurement may include hiring part-time or additional full-time staff and requesting more supportive personnel from other departments.

The future development involves the integration of the optimization model into staff scheduling software. The software should allow nurses to submit and edit their requests and allow the scheduler to prioritize the optimization goals. Furthermore, the software should include a shift exchange system that syncs with the optimized schedule, so the burden of manually updating the schedule will be dissipated. The additional feature like demand forecasting is also feasible since the hospital has recorded patient numbers in each shift. The demand forecasting will define the optimal staffing requirement to align with the anticipated patient, thus promoting demand-supply matching. Moreover, the development should expand to other types of the working plan, such as OT schedule and tasking table.

Moreover, even though the developed prototype is tailored to fit the ED of Siriraj Hospital, the basic model is obtained by disabling some constraints or specific context from the models. Therefore, another beneficial output in this study is the fundamental components of the healthcare staff scheduling model that can be applied to other hospitals.

5.3 GA's parameter tuning

In Test 1, the case with entirely binary random genes gave a negative fitness value, thus failing to converge within the limited generation. The primary cause is the highly constrained problem in the scheduling task. With many restrictions incorporated in the model, the stochastic optimization usually fails to align due to the randomness of its search direction. Additionally, this model employed binary encoding with the one-bit-per variable scheme. So, the slight variation of a gene can lead to a significant change in the decoded variable.Then, it is common for GA to introduce random changes to the chromosomes, and more changes imply higher chances of violating the constraints. However, it does not mean that the binary random population will never overcome other population types. Once the binary random population converges to the feasible solution and starts optimizing, a more optimal solution is probable. Because of its origin with colossal stochastic, it is less prone to stuck in some local optimum.

The remaining sub-tests on using other mixing ratios show insignificant results. Regardless of the small or large portion of the MILP's solution, the initial population always contains two groups of chromosomes: the superior one (i.e., the feasible solution from MILP) and the poor one (i.e., the binary random individuals). The algorithm may biasedly favour the superior group since the early generations, causing the model to experience premature convergence.Nonetheless, the initial population mixing both types yields the best outcome under the termination criteria of limited generations.

In Test 2 and 3, the crossover probability and the momentum adjustment do not correlate to the fitness values. In other words, any change introduced from the multi-point crossover (either small or large change) usually causes the penalty in a highly constrained problem. Especially when the multi-point crossover involves random changes of multiple genes, the nature of binary encoding with the one-bit-per-variable scheme expeditely highlight the chance of violations. Since the algorithm always has the best chromosomes with positive fitness from its mixing initial population and elitism, those individuals with a penalty added from the crossover will be eased out by the algorithm. Therefore, the crossover in this context may not be the source of offspring enhancement but rather constraint violations.

In Test 4, the result exhibits that the momentum adjustment on mutation constant reduces the population's fitness. The possible rationality is that the mutation is the primary drive toward optimization in this context. Unlike the crossover that introduces a massive change to the chromosome, the mutation only occurs to few genes. Consequently, if the mutation happens on the superior chromosome (i.e., the non-optimized feasible solution from MILP), there is a chance that a small change may not cause a penalty but lead to the optimization of some aspects. In other words, the superior chromosome will receive less probability of optimization if the mutation is decreased.

According to the analysis of all tuning tests, the termination criterion of the limited number of generations may not be practical. In the highly constrained problem, it is not easy for the GA to converge a binary random population into the optimal one. Then, the remaining solution is to rely on the non-optimized feasible solution from MILP, which can cause premature convergence. Therefore, future work should concentrate on the fitness threshold as the termination criterion and focus on binary random population parameter tunings. Alternatively, it is also interesting to apply a set of MILP's non-optimized feasible solutions into the initial population of GA, which may introduce more variety to the batch, thus reducing premature convergence. Moreover, this study included only some parameters in the tuning process. So, there may be a chance that the tuning of the remaining parameters, e.g., the crossover points and the tournament size, can enhance the algorithm performance.

5.4 The collation of MILP-optimized and GA-optimized schedule

From the collation of MILP-optimized and GA-optimized schedules, 59.21% of the evaluation metrics have unchanged outcomes. These homogenous results are probably contributed from the GA's premature convergence, as discussed in Section 5.3. This premature convergence makes the characteristic of GA's solution similar to the MILP's non-optimized feasible solution. Furthermore, some evaluation metrics can be satisfied via the hard constraints, such as the number of afternoon and night shifts and the number of days off after night shifts. Then, those constraint-based metrics are homogenously satisfied by both models, as both GA and MILP provided penalty-free schedules.

Nonetheless, MILP has 3.95% of the metric with deteriorated outcomes. These deteriorations include the number of working days and the off-bank quota transaction in some nurse's levels. The number of working days in MILP is higher than in GA because MILP reduces 54-84% of the double-shifts days. As a result, MILP's schedule contains more working days but with fewer double-shifts days. For the off-bank quota transaction, the increment leap of the off-bank quota in MILP may be constituted from the request satisfaction. MILP proves its efficacy in satisfying all request types at almost 100%. Therefore, nurses may simultaneously ask for many days off in that schedule, causing less available staff to be scheduled. Then, some remaining nurses may have to work extra to cover the insufficient staff, thus earning an extra off-bank quota.

There are 36.84% of metrics that MILP can superiorly optimize. The vast difference in optimization capability is caused by the principle of solution search in MILP and GA. In MILP, the CBC solver with the branch-and-bound foundation applies the search tree to explore the search space with deterministic and logical direction toward the optimum. However, the core principle of GA is the randomness in its search direction. Hence, the basic GA cannot perceive which features on the chromosome are favoured; it only knows whether that chromosome is fit or weak as a whole. Particularly in the highly constrained problem, the random search direction regularly brings constraint violations to the population.

Additionally, the GA solution in this study is anticipated to have premature convergence to the non-optimization feasible solution. The stochastic evolution is not expected to raise much fitness, especially in the problem with many constraints. The small increment of 7% in the last generation's fitness value from the initial population's fitness value supports this assumption. To be comparable with MILP, GA has plenty of room to be improved. The future development of GA should start with parameter tuning, as explained in Section 5.3.

5.5 The collation of manual and MILP-optimized schedule

Comparing the manual and MILP-optimized schedules in May-June and July-August, all approaches yield more than half of the evaluation metrics with unchanged outcomes. The results from both periods imply that MILP has comparable performance with human resources in optimizing most of the schedule features.

Still, MILP provides 6.58% and 9.21% of evaluation metrics with deteriorated outcomes in the May-June and July-August, respectively. These inferior features of MILP in comparison with the manual one are discussed as follows. The MILP's schedules from both periods raise the number of working days. However, the scenarios happen under insignificant changes in the number of the shift while significantly lowering the double-shifts days. So, the increase in working days is beneficial in MILP, as it reduces work stress.

The MILP's schedules from both periods also possess the lift of the night shifts and the days off after the night shifts in Senior 3. One reason accounted for this result is the eligibility of Senior 3 to work in any position in the night shift. In order to fulfil the staffing requirement of the night shift, Senior 3 nurses are usually the primary candidates for the MILP algorithm to utilize. On the contrary, manual scheduling tends to overuse Junior nurses to cover the staffing requirements. MILP also shows that night shifts in Junior nurses significantly decrease in both periods, thus confirming the overuse of Junior in manual scheduling.

Compared to manual scheduling, MILP fails to provide superior approvals in some request types, i.e., not-to-be-assigned shifts requests in May-June and days off requests in July-August. The inference for this issue is the tradeoff between granting all absent requests and having enough staff in the shift. In the manual case, the scheduler tends to approve all requests as much as possible, thus leaving some days or shifts with inadequate staff. Therefore, staff in the manual schedule usually work more than one shift a day to cover the inadequacies or work in a shift with mismatched staffing. In contrast, MILP perceives staffing criteria as hard constraints, so it may have to sacrifice some approvals on absent requests to align with the staffing. Since the number of requests for not-to-be-assigned shifts and days off takes a significant portion in absent requests, these groups priorly get denials.

In the May-June schedule, MILP gives the surge in off-bank quota transactions in Senior 1, while the manual schedule supplies almost zero transactions in the off-bank quota. The same assumption for the Senior 3's increasing night shifts can be applied here. Because Senior 1 nurses typically work in the morning shifts, their highest rank makes them eligible to serve any role. So, if there are lacking staff in the morning shift, Senior 1 nurses are the primary candidates to be selected, thus raising the number of their working shifts. The insignificantly higher number of Senior 1's shift in MILP shows that this figure exceeds the on-duty threshold. At the same time, the manual schedule has its number approximately at the threshold. Then, some quotas are added to the off bank in MILP due to its extra working shifts.

Despite the few deteriorated results, MILP significantly optimizes nearly 24% to 25% of the evaluation metrics compared to the manual approach in both periods. By implementing mathematical logic for the complicated scheduling task, MILP is more capable than manpower in optimizing schedules. The optimization exerted by MILP includes the decline of work stress, staff satisfaction, and fairness. Especially with the drop of double-shifts days, the work stress is likewise lifted. Apart from request approvals, MILP can satisfy each nurse with at least two weekends off per schedule. Moreover, almost every value of the decision variables is superior in MILP than in manual scheduling, and their values imply either maximum or minimum limit. So, MILP can provide more fairness to staff.

For example, the manual scheduling in May-June can approve not-to-be-assigned shift requests approximately 2% greater than MILP. Nevertheless, its maximum disapprovals are five applications. This characteristic may imply unfairness in request approvals, e.g., some nurses who submit only one request may receive a denial, or few unfortunate nurses may receive at most five denials while most of the staff earn approvals. In contrast, MILP assures that no one will receive disapproval greater than 1, regardless of the number of requests each staff submitted.

Another critical downside of manual scheduling is human error. It is found that the manual approach violates the scheduling policy by 139 and 78 times in May-June and July-August, respectively. Most of the infringements are related to work stress and staffing criteria. The consequence may adversely affect work-life quality, staff satisfaction, care quality, and patient safety. In addition, the complexity of the task makes manual scheduling consume enormous time to execute. So, even though manual scheduling can produce few metrics with better outcomes than the MILP; the price it has to pay is the processing time and potential policy violations.

In terms of the future development of MILP, there are vast opportunities to be explored which includes the adaptive objective weights, the shift balance improvement, and the computation time. Currently, the objective weights of each schedule have to be manually tuned until desirable results are acquired. There is no one-size-fits-all set of weights that can be applied to any scheduling problem because the weights majorly depend on the characteristics of inputs that varied among schedules. Moreover, not all objectives are required to be active. For instance, the objective for balancing total shifts is suitable only when nurses have numbers of shifts within the on-duty threshold to prevent them from working extra. This condition is the reason why the second object is disabled in this study. Consequently, it is interesting to develop adaptive objective weights that automatically give suitable weights of each scheduling problem by learning from the input characteristics and user needs.

The current method tackles the shift balance aspect by applying the shifts' limitation policy as hard constraints. Still, this approach only limits each shift type's upper range, but not the lower bound. Then, the concept of soft constraints for shift balance is plausible for optimizing the balance in both upper and lower bound. However, this implementation means an increase in the model's size. Thus additional investigation is needed to acquire the balance without adding much computation time.

For the computation time, MILP took 24 to 35 minutes to optimize schedules from both periods. These time durations are rather extensive for computer capability. Additionally, the relative gap tolerance of the objective's lower and upper bound was set to 0.175. This setting means that there might be other optimal solutions whose duality gaps are less than 17.5%. Nevertheless, the CBC solver may not be efficient enough to provide superior solutions under short computation time. The possible reason for this ineffectiveness is its search strategies technique, leading to extensive tree node cuts. Compared to other commercial solvers like CPLEX and Gurobi, these solvers adopt more exceptional techniques of search strategies to the branch-and-cut algorithm, thus making them approach the optimum in a faster time. As a result, the choice of buying a licence of the commercial solver may be beneficial to the hospital in obtaining a more optimal solution under a short computation time.

6

Conclusion

To enhance the efficacy of human resource management, this thesis investigates the optimization model's capability in on-duty staff scheduling at the ED of Siriraj Hospital. Many publications prove the correlation between scheduling outcomes and staff satisfaction, which further affect the turnover rate, service quality, and care outcome. The literature reviews also establish the potentials of both deterministic and stochastic optimization models in healthcare staff scheduling applications. Thus, the preliminary research in this thesis suggests the strong feasibility of applying an optimization model to improve healthcare staff resource management through effective scheduling.

Following the first objective, the requirements for staff scheduling are collected from the interviews with both the shift managers and governed staff. The service blueprint is constructed to provide the perception of the scheduling process. The elements in the blueprint illuminate scheduling-related activities starting from requesting a schedule until working in a shift. The process analysis implies that on-duty scheduling is the bottleneck activity that consumes most hospital resources, including people and time. The leading causes of this bottleneck are the complexity of scheduling and the ineffectiveness of the assistive tool. Following the development of the service blueprint, the in-depth insights extraction is performed to define the direction of model development. Those insights are translated into the model parameters, constraints, and objective functions. Nevertheless, the optimization model alone cannot satisfy some requirements simultaneously. Cooperation between the optimization model and the hospital's management is essential.

In the second objective, two models, each from different optimization types, are developed, i.e., MILP from deterministic optimization and GA from stochastic optimization. The objective function applied in both optimization methods is the summation of multiple objectives through weighting and normalization. The optimization goals include three principal aspects: reducing works stress, promoting request approvals, and providing fairness and balance.

From GA's parameter tuning results, the initial population with any proportion of MILP's non-optimized solution can converge to a feasible solution within the limited generations, yet exist with solid evidence of premature convergence. In a highly constrained problem, the stochastic of the multi-point crossover is always too exessive, regardless of the crossover probability and the momentum implementation. However, the mutation is found to be the source of optimization in this context.

The performance comparison between MILP and GA proves the superior capability of MILP in scheduling optimization. The collation shows that both methods yield 59.21% of the evaluation metrics with unchanged outcomes, but MILP can supply 36.84% of optimized outcomes. Only 3.95% of MILP metrics fail to defeat GA because of the trade-off between succeeding in the major goals and missing the trivial features. These trade-off pairs include reducing double-shift days while gaining working days and approving many absent requests while raising day-off bank transactions. The inferiority of GA is contributed by its random search for the solution that usually introduces penalties to its population, especially in the highly constrained problem. In contrast, the deterministic search in MILP applies branch-and-bound logic to converge its solution to the optimum.

In comparing manual and MILP-optimized schedules, both approaches provide more than half of the evaluation metrics with unchanged outcomes, thus proving the comparable performance in optimizing most of the schedule's features. Moreover, MILP overcomes manual scheduling by significantly optimizing 24% to 25% of the metrics. The improvements urged by MILP include reducing work stress, staff satisfaction, fairness, zero violation of policy, and cutting scheduling time. Only 6.58% to 9.21% of MILP's metrics are inferior to the manual approach. Similar to the case of GA-MILP comparison, the plausible reason for this inferiority is the trade-off between thriving the significant purposes and dropping the minor traits. The exchange pairs include decreasing double-shift days while increasing working days, as well as satisfying staffing requirements while gaining extra shifts or denying absent requests. Additionally, the manual approach violates the scheduling restrictions more than 78 times, mainly in the policy related to work stress and staffing criteria.

According to the findings in this thesis, the MILP optimization model shows more superior performance than the GA model and manual approach in optimizing the scheduling of healthcare staff at the ED of Siriraj Hospital. The MILP-optimization outcomes in diminishing work stress, enhancing staff satisfaction, providing fairness, aligning with policy, and shortening the processing time can lead to excellence in service quality and care outcome while lowering the turnover rate. As a result, the optimization of healthcare staff scheduling with the MILP model exerts the capability of human resource management to its greater extent.

Bibliography

[1] J. Marchesi, S. Hamacher, and J. Fleck, “A stochastic programming approach to the physician staffing and scheduling problem,” *Computers & Industrial Engineering*, vol. 142, p. 106281, 02 2020.

[2] D. Kellogg and S. Walczak, “Nurse scheduling: From academia to implementation or not?” *Interfaces*, vol. 37, pp. 355–369, 08 2007.

[3] B. Ang, S.-W. Lam, Y. Pasupathy, and M. Ong, “Nurse workforce scheduling in the emergency department: A sequential decision support system considering multiple objectives,” *Journal of Nursing Management*, vol. 26, 12 2017.

[4] I. Rizany, R. Hariyati, E. Afifah, and Rusdiyansyah, “The impact of nurse scheduling management on nurses’ job satisfaction in army hospital: A cross-sectional research,” *SAGE Open*, vol. 9, p. 215824401985618, 04 2019.

[5] Al-Najjar, Sabah and Ali, Samir, “Staffing and scheduling emergency rooms in two public hospitals: A case study,” *International Journal of Business Administration*, vol. 2, 05 2011.

[6] M. Warner, *Personnel Staffing and Scheduling*, 11 2013, pp. 253–271.

[7] S. Siferd and W. Benton, “Workforce staffing and scheduling - hospital nursing specific models,” *European Journal of Operational Research*, vol. 60, pp. 233–246, 02 1992.

[8] L. H. Aiken, S. P. Clarke, D. M. Sloane, J. Sochalski, and J. H. Silber, “Hospital Nurse Staffing and Patient Mortality, Nurse Burnout, and Job Dissatisfaction,” *JAMA*, vol. 288, no. 16, pp. 1987–1993, 10 2002. [Online]. Available: https://doi.org/10.1001/jama.288.16.1987

[9] S. Knox and J. Irving, “An interactive quality of work life model applied to organizational transition,” *The Journal of nursing administration*, vol. 27, pp. 39–47, 02 1997.

[10] WHO, *State of the world’s nursing 2020: investing in education, jobs and leadership.* World Health Organization, 2020.

[11] W. Srisuphan, W. Kunaviktikul, O. Tonmukayakul, C. Charoenyuth, and N. Sirikanokwilai, “Supply and requirement projection of professional nurses in thailand over the next two decades (1995 -2015 a.d.),” 08 2021.

[12] P. Sirisub, M. Tipayamongkholgul, N. Howteerakul, and T. Noree, “Intention to extend working life among thai registered nurses in ministry of public health: A national survey,” *Nursing Research and Practice*, vol. 2019, pp. 1–11, 06 2019.

[13] A. Dumrongsiri and P. Chongphaisal, “Nurse scheduling in a hospital emergency department: A case study at a thai university hospital,” *Songklanakarin Journal of Science and Technology*, vol. 40, pp. 187–196, 02 2018.

[14] P. Rerkjirattikal, V.-N. Huynh, S. Olapiriyakul, and T. Supnithi, *A Framework for a Practical Nurse Scheduling Approach: A Case of Operating Room of a Hospital in Thailand*, 07 2020, pp. 259–264.

[15] S. Hospital, “History,” Mar 2021. [Online]. Available: https://www2.si.mahidol.ac.th/en/history/

[16] M.-H. Lin, J.-F. Tsai, and C.-S. Yu, “A review of deterministic optimization methods in engineering and management,” *Mathematical Problems in Engineering*, vol. 2012, 06 2012.

[17] M. Cavazzuti, *Optimization methods: from theory to design*, 01 2013.

[18] M. A. Centeno, R. Giachetti, R. Linn, and A. Ismail, “A simulation-ilp based tool for scheduling er staff,” *Proceedings of the 2003 Winter Simulation Conference, 2003.*, vol. 2, pp. 1930–1938 vol.2, 2003.

[19] M. Isken, “An implicit tour scheduling model with applications in healthcare,” *Annals OR*, vol. 128, pp. 91–109, 04 2004.

[20] S. Topaloglu, “A shift scheduling model for employees with different seniority levels and an application in healthcare,” *European Journal of Operational Research*, vol. 198, pp. 943–957, 11 2009.

[21] J. Brunner and G. Edenharter, “Long term staff scheduling of physicians with different experience levels in hospitals using column generation,” *Health care management science*, vol. 14, pp. 189–202, 04 2011.

[22] N. Zinouri, “Improving healthcare resource management through demand prediction and staff scheduling,” 2016.

[23] T. Chawasemerwa, I. Taifa, and D. Hartmann, “Development of a doctor scheduling system: A constraint satisfaction and penalty minimisation scheduling model,” vol. 7, pp. 396–422, 12 2018.

[24] J. C. Spall, *Stochastic Optimization.* Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 173–201. [Online]. Available: https://doi.org/10.1007/978-3-642-21551-3_7

[25] L. Rosocha, S. Vernerova, and R. Verner, “Medical staff scheduling using simulated annealing,” *Quality Innovation Prosperity*, vol. 19, 07 2015.

[26] N. Mohd Rasip, A. S. Basari, B. Hussin, and N. Khilwani, “A guided particle swarm optimization algorithm for nurse scheduling problem,” *Applied Mathematical Sciences*, vol. 8, pp. 5625 – 5632, 07 2014.

[27] W. Abo-Hamad and A. Arisha, “Towards operations excellence: Optimising staff scheduling for new emergency department,” 06 2013.

[28] M. Mutingi, *Healthcare Staff Scheduling: Emerging Fuzzy Optimization Approaches*, 09 2015.

[29] K. Leksakul and S. Phetsawat, "Nurse scheduling using genetic algorithm," *Mathematical Problems in Engineering*, vol. 2014, pp. 1–16, 11 2014.

[30] A. Amindoust, M. Asadpour, and S. Shirmohammadi, "A hybrid genetic algorithm for nurse scheduling problem considering the fatigue factor," *Journal of Healthcare Engineering*, vol. 2021, pp. 1–11, 03 2021.

[31] S. Bradley, A. Hax, and T. Magnanti, "Applied mathematical programming," 01 1977.

[32] D. Salvagnin, "Constraint programming techniques for mixed integer linear programs," Ph.D. dissertation, Jan 2009. [Online]. Available: http://www.dei.unipd.it/~salvagni/

[33] Gurobi, "Mixed-integer programming (mip) - a primer on the basics," Jun 2019. [Online]. Available: https://www.gurobi.com/resource/mip-basics/

[34] H. Williams, *Model Building in Mathematical Programming.* Wiley, 2013. [Online]. Available: https://books.google.co.th/books?id=YJRh0tOes7UC

[35] N. Pisaruk, *Mixed Integer Programming: Models and Methods*, 05 2019.

[36] M. Karamanov, "Branch and cut: An empirical study," Ph.D. dissertation, Carnegie Mellon University, Sep 2006. [Online]. Available: https://www.cmu.edu/tepper/programs/phd/program/assets/dissertations/2006-operations-research-karamanov-dissertation.pdf

[37] C. Roos, "Linear optimization," in *Encyclopedia of Physical Science and Technology (Third Edition)*, third edition ed., R. A. Meyers, Ed. New York: Academic Press, 2003, pp. 597–616. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B0122274105003793

[38] IBM, "Branch and cut in cplex," 2019. [Online]. Available: https://www.ibm.com/docs/en/icos/12.10.0?topic=concepts-branch-cut-in-cplex

[39] B. Meindl and M. Templ, "Analysis of commercial and free and open source solvers for linear optimization problems," 08 2013.

[40] J. O. Hanson, "Solver benchmarks." [Online]. Available: https://cran.r-project.org/web/packages/prioritizr/vignettes/solver_benchmark.html

[41] M. Wahde, *Biologically inspired optimization methods: an introduction.* WIT Press, 2008.

[42] H. Maaranen, K. Miettinen, and A. Penttinen, "On initial populations of a genetic algorithm for continuous optimization problems," *Journal of Global Optimization*, vol. 37, pp. 405–436, 03 2007.

[43] S. Poles, Y. Fu, and E. Rigoni, *The Effect of Initial Population Sampling on the Convergence of Multi-Objective Genetic Algorithms*, 01 2009, vol. 618, pp. 123–133.

[44] S. Abdulazeez and H. H. O. Nasereddin, "Enhanced solutions for misuse network intrusion detection system using sga and ssga," Ph.D. dissertation, 06 2015.

[45] K. Sastry and D. E. Goldberg, "Analysis of mixing in genetic algorithms: A survey," Tech. Rep., 2002.

[46] T. D. Gauthier and M. E. Hawley, "Chapter 5 - statistical methods," in *Introduction to Environmental Forensics (Third Edition)*, third edition ed., B. L. Murphy and R. D. Morrison, Eds. San Diego: Academic Press, 2015, pp. 99–148. [Online]. Available: https://www.sciencedirect.com/science/article/pii/B9780124046962000059

[47] Z. Lu and K.-H. Yuan, *Welch's t test*, 01 2010, pp. 1620–1623.

A

Appendix

A.1 Acquisition of model's secondary parameters

If $\sum_{s=1}^{S_{max}} SHIFT\_REQUEST_{n,d,s} + SHIFT\_REQUEST\_VIP_{n,d,s} > 0$,

$$WORKDAY\_REQUEST_{n,d} = 1, \forall n \in N, \forall d \in D \tag{A.1}$$

If $\sum_{s=1}^{S_{max}} SHIFT\_REQUEST_{n,d,s} + SHIFT\_REQUEST\_VIP_{n,d,s} = 0$,

$$WORKDAY\_REQUEST_{n,d} = 0, \forall n \in N, \forall d \in D \tag{A.2}$$

If $\sum_{d=7w-1}^{7w} WORKDAY\_REQUEST_{n,d} > 0$,

$$WEEKEND\_WORK\_REQUEST_{n,w} = 1, \forall n \in N, \forall w \in W \tag{A.3}$$

$$TOTAL\_WEEKEND\_WORK\_REQUEST_n = \sum_{w=1}^{W_{max}} WEEKEND\_WORK\_REQUEST_{n,w}, \forall n \in N \tag{A.4}$$

If $\sum_{s=1}^{S_{max}} (SHIFT\_REQUEST_{n,d,s} + SHIFT\_REQUEST\_VIP_{n,d,s}) > 1$,

$$DOUBLE\_SHIFTS\_DAY\_REQUEST_{n,d} = 1, \forall n \in N, \forall d \in D \tag{A.5}$$

If $\sum_{s=1}^{S_{max}} (SHIFT\_REQUEST_{n,d,s} + SHIFT\_REQUEST\_VIP_{n,d,s}) \leq 1$,

$$DOUBLE\_SHIFTS\_DAY\_REQUEST_{n,d} = 0, \forall n \in N, \forall d \in D \tag{A.6}$$

$$TOTAL\_DOUBLE\_SHIFTS\_DAY\_REQUEST_n = \sum_{d=1}^{D_{max}} DOUBLE\_SHIFTS\_DAY\_REQUEST_{n,d}, \forall n \in N \tag{A.7}$$

$$TOTAL\_OUT\_REQUEST\_VIP_n = \sum_{d=1}^{D_{max}} OUT\_REQUEST\_VIP_{n,d}, \forall n \in N \tag{A.8}$$

$$WEEK\_OUT\_REQUEST\_VIP_{n,w} = \sum_{d=7(w-1)+1}^{7w} OUT\_REQUEST\_VIP_{n,d}, \forall n \in N, \forall w \in W \tag{A.9}$$

$$TOTAL\_OFF\_REQUEST\_VIP_n = \sum_{d=1}^{D_{max}} OFF\_REQUEST\_VIP_{n,d}, \forall n \in N \tag{A.10}$$

$$ALL\_OFF\_REQUEST_{n,d} = WEEK\_OFF\_REQUEST_{n,d} + BANK\_OFF\_REQUEST_{n,d}, \forall n \in N, \forall d \in D \tag{A.11}$$

$$TOTAL\_ALL\_OFF\_REQUEST_n = \sum_{d=1}^{D_{max}} ALL\_OFF\_REQUEST_{n,d}, \forall n \in N \tag{A.12}$$

$$TOTAL\_ALL\_OFF\_REQUEST_{max} = \max_{n \in N}(TOTAL\_ALL\_OFF\_REQUEST_n), \forall n \in N \tag{A.13}$$

$$TOTAL\_BANK\_OFF\_REQUEST_n = \sum_{d=1}^{D_{max}} BANK\_OFF\_REQUEST_{n,d}, \forall n \in N \tag{A.14}$$

If $\sum_{d=1}^{D_{max}} WEEK\_OFF\_REQUEST_{n,d} > (D_{max} - ONDUTY\_DAY)$ and $TOTAL\_ALL\_OFF\_REQUEST_n - (D_{max} - ONDUTY\_DAY) \geq TOTAL\_DOUBLE\_SHIFTS\_DAY\_REQUEST_n$,

$$\begin{aligned} TOTAL\_ACTUAL\_BANK\_OFF\_USES_n = \\ TOTAL\_ALL\_OFF\_REQUEST_n - \\ (D_{max} - ONDUTY\_DAY) - \\ TOTAL\_DOUBLE\_SHIFTS\_DAY\_REQUEST_n, \\ \forall n \in N \end{aligned} \tag{A.15}$$

If $\sum_{d=1}^{D_{max}} WEEK\_OFF\_REQUEST_{n,d} \leq (D_{max} - ONDUTY\_DAY)$ and $TOTAL\_BANK\_OFF\_REQUEST_n \geq TOTAL\_DOUBLE\_SHIFTS\_DAY\_REQUEST_n$,

$$TOTAL\_ACTUAL\_BANK\_OFF\_USES_n = TOTAL\_BANK\_OFF\_REQUEST_n - TOTAL\_DOUBLE\_SHIFTS\_DAY\_REQUEST_n, \forall n \in N \tag{A.16}$$

If previous conditions of Equation A.15 and A.16 do not hold,

$$TOTAL\_ACTUAL\_BANK\_OFF\_USES_n = 0, \forall n \in N \tag{A.17}$$

$$TOTAL\_VAC\_REQUEST_n = \sum_{d=1}^{D_{max}} VAC\_REQUEST_{n,d}, \forall n \in N \tag{A.18}$$

$$TOTAL\_VAC\_REQUEST_{max} = \max_{n \in N}(TOTAL\_VAC\_REQUEST_n) \tag{A.19}$$

$$TOTAL\_SHIFT\_REQUEST_n = \sum_{d=1}^{D_{max}} \sum_{s=1}^{S_{max}} SHIFT\_REQUEST_{n,d,s}, \forall n \in N \tag{A.20}$$

$$TOTAL\_SHIFT\_REQUEST_{max} = \max_{n \in N}(TOTAL\_SHIFT\_REQUEST_n) \tag{A.21}$$

$$TOTAL\_NO\_SHIFT\_REQUEST_n = \sum_{d=1}^{D_{max}} \sum_{s=1}^{S_{max}} NO\_SHIFT\_REQUEST_{n,d,s}, \forall n \in N \tag{A.22}$$

$$TOTAL\_NO\_SHIFT\_REQUEST_{max} = \max_{n \in N}(TOTAL\_NO\_SHIFT\_REQUEST_n) \tag{A.23}$$

If $WEIGHT\_OBJ_o > 0$,

$$ACTIVE_OBJ_o = 1, \forall o \in O \tag{A.24}$$

If $WEIGHT\_OBJ_o = 0$,

$$ACTIVE\_OBJ_o = 0, \forall o \in O \tag{A.25}$$

$$TOTAL\_ACTIVE\_OBJ = \sum_{o=1}^{O_{max}} ACTIVE\_OBJ_o \tag{A.26}$$

A.2 Staffing criteria

Table A.1: Staffing criteria for weekday

	Morning shift	Afternoon shift	Night shift
Senior 1 and 2	4	2	0
Senior 3	3	2	3
Middle	3	3	2
Junior 1	0	0	4
Junior 2	10	10	4

Table A.2: Staffing criteria for weekend or holiday

	Morning shift	Afternoon shift	Night shift
Senior 1 and 2	4	2	0
Senior 3	3	2	3
Middle	3	3	2
Junior 1	0	0	4
Junior 2	6	10	4

A.3 Evaluation algorithm

Data: Table 3.5 and 3.6
Result: $p, ws_{max}, ws_{min}, dsd_{max}, wko_{min}, dor_{max}, dvr_{max}, dsr_{max}, dnsr_{max}$
Initialization: $p = 0, ws_{max} = 0, ws_{min} = TOTAL\_SHIFTS_{max}, dsd_{max} = 0, wko_{min} = W_{max}, dor_{max} = 0, dvr_{max} = 0, dsr_{max} = 0, dnsr_{max} = 0$;

```
for n in N_max do
    for w in W_max do
        if Nurse n works > WEEK_SHIFTS_max shifts in week w then
            Add penalty to p;
        end
        if Nurse n works > WEEK_WORKDAYS_max days in week w then
            Add penalty to p;
        end
        if Nurse n works night shifts > WEEK_N_SHIFTS_max in week w^i then
            Add penalty to p;
        end
    end
    for d in D_max do
        if Nurse n works consecutively > CONSEC_WORKDAYS_max days starting from
          day d^i then
            Add penalty to p;
        end
        if Nurse n works afternoon shifts consecutively > CONSEC_A_SHIFTS_max days
          starting from day d^i then
            Add penalty to p;
        end
        if Nurse n is a senior nurse who works night shifts consecutively for
          CONSEC_N_SHIFTS_SENIOR_max days starting from day d, but does not
          have a day off afterwards^i then
            Add penalty to p;
        end
        if Nurse n is not a senior nurse who works night shifts consecutively for
          CONSEC_N_SHIFTS_NONSENIOR_max days starting from day d, but does
          not have a day off afterwards ^i then
            Add penalty to p;
        end
        if Nurse n works in afternoon-and-night shifts on day d, or either
          afternoon-and-morning^i shifts, night-and-morning shifts, or night-and-afternoon
          shifts^i on both day d and one day afterward then
            Add penalty to p;
        end
        if Nurse n has a level of Senior 1 and works either in an afternoon shift or a night
          shift on day d^i then
            Add penalty to p;
        end
        if Nurse n has a level of Senior 2 and works in a night shift on day d^i then
            Add penalty to p;
        end
        if Nurse n has a VIP request for a day off or an out-of-unit position on day d, but
          the request is not granted then
            Add penalty to p;
        end
    end
end
```

Continue from the previous page

for n *in* N_{max} **do**
    for d *in* D_{max} **do**
        for s *in* S_{max} **do**
            if *Nurse n has a VIP request for a shift or a not-to-be-assigned shift on day d in shift s, but the request is not granted* **then**
                Add penalty to p;
            end
        end
    end
    Count the total on-duty shifts of nurse n, which include working shifts, out-of-unit position, vacation, and off-bank quota usage;
    if *Nurse n has the total on-duty shifts* $> ws^i_{max}$ **then**
        Update ws_{max}
    end
    if *Nurse n has the total on-duty shifts* $< ws^i_{min}$ **then**
        Update ws_{min}
    end
    Count the total afternoon shifts of nurse n;
    if *Nurse n has the total afternoon shifts* $> TOTAL\_A\_SHIFTS^i_{max}$ **then**
        Add penalty to p;
    end
    Count the total night shifts of nurse n;
    if *Nurse n has the total night shifts* $> TOTAL\_N\_SHIFTS^i_{max}$ **then**
        Add penalty to p;
    end
    Count the total double-shifts days of nurse n;
    if *Nurse n has the total double-shifts days* $> dsd^i_{max}$ **then**
        Update dsd_{max};
    end
    Count total weekends off of nurse n;
    if *Nurse n has weekends off* $< TOTAL\_WEEKENDS\_OFF^i_{min}$ **then**
        Add penalty to p;
    end
    if *Nurse n has weekends off* $< wko_{min}$ **then**
        Update wko_{min}
    end
    Count the total disapprovals for off requests of n;
    if *Nurse n receives the total disapprovals for off requests* $> dor_{max}$ **then**
        Update dor_{max};
    end
    Count the total disapprovals for vacation requests of n; **if** *Nurse n receives the total disapprovals for vacation requests* $> dvr_{max}$ **then**
        Update dvr_{max};
    end
    Count the total disapprovals for shift requests of n; **if** *Nurse n receives the total disapprovals for shift requests* $> dsr_{max}$ **then**
        Update dsr_{max};
    end
    Count the total disapprovals for not-to-be-assigned shift requests of n; **if** *Nurse n receives the total disapprovals for not-to-be-assigned shifts requests* $> dnsr_{max}$ **then**
        Update $dnsr_{max}$;
    end
end

Continue from the previous page
for d *in* D_{max} **do**
    for s *in* S_{max} **do**
        for l *in* l_{max} **do**
            if *The number of nurses level l in shift s is* $< WEEKDAY\_STAFFING\_MIN_{l,d,s}$ *on weekday d or* $WEEKEND\_STAFFING\_MIN_{l,d,s}$ *on weekend or holiday d* **then**
                Add penalty to p;
            end
        end
    end
end

Remarks:
[i] This constraint will be halted, if a nurse willingly requested for a case that violate the constraint.

A.4 Results of statistical tests

Table A.3: F-tests and t-tests of the average shifts between MILP-optimized and GA-optimized schedules (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	9.23611	0.01454	Yes	1.89443	0.10643	No
Senior 2	1.23118	0.38788	No	-0.35864	0.72455	No
Senior 3	1.21480	0.36043	No	0.18198	0.85691	No
Middle	1.02052	0.48305	No	0.69819	0.48954	No
Junior 1	1.44640	0.25756	No	-1.76380	0.08952	No
Junior 2	1.10010	0.45154	No	-0.11328	0.91142	No
All levels	1.24503	0.18072	No	-0.09819	0.92193	No

Table A.4: F-tests and t-tests of the average working days between MILP-optimized and GA-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	2.16058	0.20889	No	2.47155	0.03302	Yes
Senior 2	1.37160	0.33277	No	0.36505	0.71985	No
Senior 3	1.08369	0.44130	No	1.12267	0.27111	No
Middle	1.71700	0.13045	No	2.02263	0.05059	No
Junior 1	1.15233	0.40104	No	0.20335	0.84044	No
Junior 2	1.45916	0.31525	No	1.08772	0.29509	No
All levels	1.35940	0.10073	No	2.66621	0.00857	Yes

Table A.5: F-tests and t-tests of the average morning shifts between MILP-optimized and GA-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	2.16058	0.20889	No	2.47155	0.03302	Yes
Senior 2	1.25638	0.37732	No	-0.61817	0.54516	No
Senior 3	1.18695	0.37647	No	1.05188	0.30185	No
Middle	2.03467	0.07063	No	0.75460	0.45540	No
Junior 1	2.20940	0.08310	No	-1.31568	0.19976	No
Junior 2	1.77527	0.23332	No	-0.98455	0.34156	No
All levels	1.21563	0.20803	No	-0.01578	0.98743	No

Table A.6: F-tests and t-tests of the average afternoon shifts between MILP-optimized and GA-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.00000	0.50000	No	0.00000	1.00000	No
Senior 2	1.69444	0.23610	No	0.81228	0.42855	No
Senior 3	1.08812	0.43834	No	-0.56096	0.57929	No
Middle	1.44518	0.22117	No	0.62757	0.53425	No
Junior 1	1.92534	0.12536	No	-1.58845	0.12427	No
Junior 2	1.03226	0.48384	No	0.66667	0.51582	No
All levels	1.14167	0.29045	No	-0.08911	0.92912	No

Table A.7: F-tests and t-tests of the average night shifts between MILP-optimized and GA-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	N/A	N/A	N/A	N/A	N/A	N/A
Senior 2	N/A	N/A	N/A	N/A	N/A	N/A
Senior 3	1.06688	0.45265	No	-0.22031	0.82723	No
Middle	1.69811	0.13534	No	-0.06958	0.94491	No
Junior 1	1.79509	0.15208	No	-0.24577	0.80779	No
Junior 2	1.21522	0.40181	No	0.41441	0.68485	No
All levels	1.01596	0.47370	No	-0.04680	0.96274	No

Table A.8: F-tests and t-tests of the average double-shift days between MILP-optimized and GA-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.00000	0.50000	No	0.00000	1.00000	No
Senior 2	7.56250	0.00486	Yes	-2.90482	0.01558	Yes
Senior 3	12.14286	0.00002	Yes	-5.24093	0.00008	Yes
Middle	6.70370	0.00009	Yes	-3.72298	0.00110	Yes
Junior 1	11.00000	0.00006	Yes	-6.35085	0.00001	Yes
Junior 2	59.42857	0.00001	Yes	-4.75971	0.00188	Yes
All levels	4.67346	0.00000	Yes	-7.27807	0.00000	Yes

Table A.9: F-tests and t-tests of the average weekends off between MILP-optimized and GA-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	2.40625	0.17865	No	1.07088	0.30939	No
Senior 2	inf	0.00000	Yes	4.00000	0.00395	Yes
Senior 3	1.90909	0.11932	No	4.05046	0.00037	Yes
Middle	2.03333	0.07081	No	5.03177	0.00001	Yes
Junior 1	2.71429	0.04165	Yes	2.82843	0.00994	Yes
Junior 2	1.88889	0.21027	No	0.78446	0.44584	No
All levels	1.16042	0.26767	No	6.14022	0.00000	Yes

Table A.10: F-tests and t-tests of the average off-bank quota transactions between MILP-optimized and GA-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	14.63636	0.00524	Yes	2.04598	0.08937	No
Senior 2	1.63780	0.25047	No	-0.98345	0.34003	No
Senior 3	1.05036	0.46403	No	0.54290	0.59150	No
Middle	5.58527	0.00032	Yes	2.11068	0.04530	Yes
Junior 1	2.31295	0.07181	No	-2.35940	0.02609	Yes
Junior 2	1.10010	0.45154	No	-0.11328	0.91142	No
All levels	1.56826	0.03089	Yes	0.20020	0.84163	No

Table A.11: F-tests and t-tests of the average days off after night shifts between MILP-optimized and GA-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	N/A	N/A	N/A	N/A	N/A	N/A
Senior 2	N/A	N/A	N/A	N/A	N/A	N/A
Senior 3	1.29888	0.31565	0.00000	-1.56783	0.12815	0.00000
Middle	2.77987	0.01806	1.00000	-3.79355	0.00069	1.00000
Junior 1	2.05775	0.10330	0.00000	-2.04619	0.05098	0.00000
Junior 2	1.86702	0.21449	0.00000	-0.56980	0.57784	0.00000
All levels	1.71912	0.01239	1.00000	-2.42237	0.01679	1.00000

Table A.12: F-tests and t-tests of the average shifts between manual and MILP-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	2.25000	0.19713	No	2.19265	0.05310	No
Senior 2	1.06023	0.46805	No	0.41469	0.68387	No
Senior 3	1.10094	0.42988	No	0.24506	0.80820	No
Middle	1.11510	0.40988	No	-0.07133	0.94353	No
Junior 1	1.03183	0.47790	No	-0.72577	0.47446	No
Junior 2	1.00650	0.49670	No	0.02897	0.97730	No
All levels	1.04903	0.42093	No	0.15789	0.87477	No

Table A.13: F-tests and t-tests of the average working days between manual and MILP-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.15175	0.44029	No	1.04596	0.32020	No
Senior 2	1.14183	0.42789	No	0.56954	0.57690	No
Senior 3	1.26595	0.33255	No	1.21418	0.23482	No
Middle	1.31252	0.28499	No	1.01480	0.31697	No
Junior 1	1.07035	0.45215	No	0.74935	0.46037	No
Junior 2	1.21635	0.40135	No	0.57060	0.57732	No
All levels	1.21557	0.20809	No	1.97881	0.04980	Yes

Table A.14: F-tests and t-tests of the average morning shifts between manual and MILP-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.40878	0.35801	No	1.08864	0.30185	No
Senior 2	1.20281	0.40015	No	-0.14438	0.88700	No
Senior 3	1.57611	0.20253	No	0.71795	0.47874	No
Middle	1.89095	0.09309	No	0.74532	0.46091	No
Junior 1	1.28240	0.33021	No	-0.52481	0.60416	No
Junior 2	1.24335	0.39058	No	-0.96623	0.35032	No
All levels	1.09893	0.34710	No	0.09251	0.92643	No

Table A.15: F-tests and t-tests of the average afternoon shifts between manual and MILP-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.89063	0.25070	No	-0.22056	0.82987	No
Senior 2	17.69444	0.00025	Yes	1.31060	0.22280	No
Senior 3	1.00192	0.49860	No	-3.35564	0.00229	Yes
Middle	1.19172	0.35692	No	0.80258	0.42748	No
Junior 1	1.68675	0.17892	No	1.91641	0.06637	No
Junior 2	1.06250	0.46916	No	-0.24816	0.80761	No
All levels	1.36541	0.09754	No	0.07936	0.93686	No

Table A.16: F-tests and t-tests of the average night shifts between manual and MILP-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	N/A	N/A	N/A	N/A	N/A	N/A
Senior 2	inf	0.00000	Yes	-1.00000	0.34659	No
Senior 3	1.84573	0.13182	No	3.04579	0.00501	Yes
Middle	2.53796	0.02772	Yes	-1.93138	0.06284	No
Junior 1	1.96392	0.11845	No	-2.06845	0.04868	Yes
Junior 2	3.04636	0.08242	No	2.03368	0.06139	No
All levels	1.01791	0.47051	No	0.00000	1.00000	No

Table A.17: F-tests and t-tests of the average double-shifts days between manual and MILP-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.26563	0.40116	No	-0.08305	0.93545	No
Senior 2	4.00000	0.03334	Yes	-0.67082	0.51529	No
Senior 3	29.28571	0.00000	Yes	-3.45251	0.00357	Yes
Middle	1.65432	0.14738	No	-5.93335	0.00000	Yes
Junior 1	4.53846	0.00518	Yes	-6.37377	0.00000	Yes
Junior 2	13.71429	0.00133	Yes	-4.43179	0.00218	Yes
All levels	2.47315	0.00010	Yes	-6.22232	0.00000	Yes

Table A.18: F-tests and t-tests of the average weekends off between manual and MILP-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.50649	0.33196	No	-0.48287	0.63959	No
Senior 2	inf	0.00000	Yes	-0.60999	0.55879	No
Senior 3	1.15873	0.39334	No	0.22687	0.82217	No
Middle	3.16667	0.00938	Yes	1.07331	0.29219	No
Junior 1	2.17857	0.08683	No	4.20406	0.00027	Yes
Junior 2	1.77778	0.23278	No	1.40000	0.18328	No
All levels	2.27534	0.00036	Yes	1.66227	0.09903	No

Table A.19: F-tests and t-tests of the average off-bank quota transactions between manual and MILP-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	2.20000	0.20359	No	2.79508	0.01895	Yes
Senior 2	1.73973	0.22529	No	1.41421	0.17646	No
Senior 3	1.12557	0.41398	No	0.43355	0.66793	No
Middle	1.68605	0.13855	No	-0.22792	0.82100	No
Junior 1	1.27646	0.33318	No	-1.30368	0.20377	No
Junior 2	1.00650	0.49670	No	0.02897	0.97730	No
All levels	1.02532	0.45850	No	0.22828	0.81976	No

Table A.20: F-tests and t-tests of the average days off after night shifts between manual and MILP-optimized schedule (May-June).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	N/A	N/A	N/A	N/A	N/A	N/A
Senior 2	inf	0.00000	Yes	-1.00000	0.34659	No
Senior 3	2.28025	0.06752	No	3.38099	0.00215	Yes
Middle	2.27673	0.04474	Yes	-1.97149	0.05758	No
Junior 1	1.50760	0.23468	No	-1.88294	0.07094	No
Junior 2	1.95833	0.19754	No	1.56996	0.13874	No
All levels	1.02808	0.45406	No	0.28039	0.77959	No

Table A.21: F-tests and t-tests of the average shifts between manual and MILP-optimized schedule (July-August).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.43279	0.35137	No	0.65671	0.52619	No
Senior 2	1.12414	0.43630	No	1.02565	0.32032	No
Senior 3	1.34216	0.29465	No	0.99302	0.32921	No
Middle	1.11133	0.41263	No	-0.22932	0.81992	No
Junior 1	1.00962	0.49352	No	-0.46134	0.64871	No
Junior 2	inf	0.00000	Yes	-8.00000	0.00020	Yes
All levels	1.08457	0.36940	No	0.01654	0.98682	No

Table A.22: F-tests and t-tests of the average working days between manual and MILP-optimized schedule (July-August).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.03061	0.48721	No	0.63404	0.54028	No
Senior 2	1.29464	0.36185	No	1.49708	0.15384	No
Senior 3	1.01071	0.49219	No	2.17435	0.03829	Yes
Middle	1.14652	0.38746	No	0.87994	0.38473	No
Junior 1	1.26021	0.34756	No	0.41061	0.68500	No
Junior 2	inf	0.00000	Yes	9.29516	0.00009	Yes
All levels	1.15409	0.27810	No	2.04664	0.04262	Yes

Table A.23: F-tests and t-tests of the average morning shifts between manual and MILP-optimized schedule (July-August).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.08168	0.46672	No	0.69395	0.50351	No
Senior 2	3.46460	0.04901	Yes	-0.06296	0.95081	No
Senior 3	1.79292	0.14330	No	-1.73543	0.09366	No
Middle	1.07542	0.43955	No	0.26370	0.79351	No
Junior 1	1.78125	0.16532	No	0.38947	0.70036	No
Junior 2	6.30769	0.02065	Yes	-0.12566	0.90316	No
All levels	1.06562	0.39701	No	0.00000	1.00000	No

Table A.24: F-tests and t-tests of the average afternoon shifts between manual and MILP-optimized schedule (July-August).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.23457	0.41139	No	-0.07433	0.94221	No
Senior 2	13.20690	0.00072	Yes	0.90575	0.38816	No
Senior 3	2.41200	0.05553	No	-1.17765	0.24885	No
Middle	3.03622	0.01165	Yes	0.00000	1.00000	No
Junior 1	1.09559	0.43847	No	-0.65293	0.52001	No
Junior 2	2.87500	0.11220	No	1.79605	0.09769	No
All levels	1.35539	0.10624	No	0.00000	1.00000	No

Table A.25: F-tests and t-tests of the average night shifts between manual and MILP-optimized schedule (July-August).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	N/A	N/A	N/A	N/A	N/A	N/A
Senior 2	inf	0.00000	Yes	-1.00000	0.34659	No
Senior 3	1.82116	0.13703	No	4.11096	0.00031	Yes
Middle	1.53068	0.18745	No	-0.80628	0.42538	No
Junior 1	1.16221	0.39941	No	-0.95368	0.34975	No
Junior 2	22.00000	0.00077	Yes	-2.74575	0.03066	Yes
All levels	1.09368	0.35652	No	0.02145	0.98291	No

Table A.26: F-tests and t-tests of the average double-shifts days between manual and MILP-optimized schedule (July-August).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.00000	0.50000	No	0.00000	1.00000	No
Senior 2	inf	0.00000	Yes	-1.15470	0.28154	No
Senior 3	11.38462	0.00002	Yes	-4.37880	0.00044	Yes
Middle	11.71429	0.00000	Yes	-6.67799	0.00000	Yes
Junior 1	3.90625	0.01281	Yes	-6.06021	0.00001	Yes
Junior 2	inf	0.00000	Yes	-20.00000	0.00000	Yes
All levels	2.47070	0.00013	Yes	-7.03060	0.00000	Yes

Table A.27: F-tests and t-tests of the average weekends off between manual and MILP-optimized schedule (July-August).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	2.41667	0.17750	No	-0.69843	0.50083	No
Senior 2	inf	0.00000	Yes	1.83533	0.10379	No
Senior 3	12.14286	0.00002	Yes	-1.10335	0.28591	No
Middle	2.45161	0.03242	Yes	0.33489	0.74000	No
Junior 1	1.55556	0.22766	No	1.35133	0.18919	No
Junior 2	N/A	N/A	No	inf	0.00000	Yes
All levels	3.08300	0.00000	Yes	1.10167	0.27305	No

Table A.28: F-tests and t-tests of the average off-bank quota transactions between manual and MILP-optimized schedule (July-August).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	1.77019	0.27302	No	0.84705	0.41678	No
Senior 2	1.70000	0.23474	No	1.15470	0.26517	No
Senior 3	1.30496	0.31261	No	1.52517	0.13843	No
Middle	2.80608	0.01726	Yes	-0.85339	0.40035	No
Junior 1	1.15096	0.40578	No	-0.46659	0.64500	No
Junior 2	inf	0.00000	Yes	-8.00000	0.00020	Yes
All levels	1.37214	0.09732	No	0.02381	0.98104	No

Table A.29: F-tests and t-tests of the average days off after night shifts between manual and MILP-optimized schedule (July-August).

Level	F-test			t-test		
	F-test statistic	p-value	null hypothesis rejection	t-test statistic	p-value	null hypothesis rejection
Senior 1	N/A	N/A	N/A	N/A	N/A	N/A
Senior 2	N/A	N/A	N/A	N/A	N/A	N/A
Senior 3	1.82822	0.13551	No	4.18965	0.00025	Yes
Middle	1.72082	0.12949	No	-0.34801	0.72986	No
Junior 1	1.06019	0.46052	No	-1.39340	0.17627	No
Junior 2	10.40000	0.00589	Yes	-2.52357	0.03896	Yes
All levels	1.17415	0.25490	No	0.25766	0.79706	No

DEPARTMENT OF SOME SUBJECT OR TECHNOLOGY
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY