INTEGRATED NURSE STAFFING AND ASSIGNMENT UNDER UNCERTAINTY

by

PRATTANA PUNNAKITIKASHEM

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

August 2007

To my father Pricha and my mother Narumol Punnakitikashem

ACKNOWLEDGEMENTS

These following acknowledgments cannot describe how grateful I am to my professors, family, and friends. First of all, I would like to express my deep gratitude to my supervising professor Dr. Jay M. Rosenberger for his invaluable advice during my entire course of doctoral studies. Not only his constant motivation and encouragement, but also I am extremely thankful for his patience, support, and consistent help in my research. I appreciate his understanding and taking time answering and guiding me toward my research. I am highly grateful to my supervising committee, Dr. H. W. Corley, Dr. Victoria C. P. Chen, Dr. Seoung B. Kim and Dr. Sharon Judkins for their interest in my research and their helpful advice for my dissertation. I would also like to specially thank Deborah F. B. Behan, my non-voting supervising committee, for always providing me guidance and good input in the nursing area.

I am especially grateful to Dr. H. W. Corley, who first introduced me to the operations research field and to my supervising professor; he convinced me to pursue my doctorate at UTA and join COSMOS. I appreciate Dr. Jay M. Rosenberger, Dr. H. W. Corley, Dr. Victoria C. P. Chen, and Dr. Seoung B. Kim for instructing me in many courses that provided good background in operations research and statistics. Working closely with Dr. Jay M. Rosenberger and Deborah F. B. Behan in applied operations research in the nursing area was very good experience for me. I would also like to thank Terry Clark from Baylor Regional Medical Center at Grapevine TX, for providing data for this research. I would like to extend my appreciation to the Robert Wood Johnson Foundation, Department of Industrial and Manufacturing Systems, and the Graduate School of The University of Texas at Arlington for providing financial support for my doctoral studies.

I highly appreciate to all the teachers who taught me during the years I spent in school at Chulalongkorn University, Thailand, and at The University of Texas at

Arlington, the Unites States. I am grateful to Dr. K. R. Rao for encouraging me to pursue my graduate studies at UTA.

I would like to thank all COSMOS labmates, i.e., Drs. Aihong Wen, Hee-su (Peter) Hwang, Venkata Pilla, Sheela Siddappa, Dachuan (Thomas) Shih, Tai Kuan Sung; Siriwat Visoldilokpun, Prashant Tarun, Huiyuan Fan, Chivalai Temiyasathit, Thuntee Sukchotrat, and Panaya Rattakon for helpful discussions and friendships. I would also like to particularly thank Siriwat Visoldilokpun for his support and discussion in both academic and personal aspects. My special thanks go to my closed friends, Voratima Orawannukul, Kanyanat Ngamlertsirichai, Kornkanda Joowong, Lasa Sophonpanich, and Peetinuch Utaiwan for always listening to me and comforting me during hard times. Many thanks also go to friends at UTA, Martha Gabriela Contreras, Songhee Gardner, and Yodchanan Wongsawat for making my life here pleasurable.

My deep gratitude goes to my parents for their inspiration and encouragement for me to pursue my doctoral degree. I am thankful for their endless love, sacrifice, patience, and always having faith in me. I would also like to thank my brothers for their moral support. Finally, I am indebted to my grandparents, aunts, uncles, and members of my family for their love, blessing, and support given in many ways.

July 18, 2007

ABSTRACT

INTEGRATED NURSE STAFFING AND ASSIGNMENT UNDER UNCERTAINTY

Publication No. ______

PRATTANA PUNNAKITIKASHEM, Ph.D.

The University of Texas at Arlington, 2007

Supervising Professor: Jay M. Rosenberger

One of the major problems in the United States health care system is a shortage of nurses. Baby boomers and an increasing elderly population are the main reasons for a rapidly growing demand for nurses. Besides the decline in enrollments in registered nurse degree programs, many nurses have suffered from work burnout. High workloads and undesirable schedules are two major issues that cause nurses' job dissatisfaction. As a result, nurses plan to leave their jobs causing low retention and low entering rates. One consequence of the shortage is that excessive workload on nurses decreases the quality of patient care. Many states have seriously considered taking actions to cope with the shortage to ensure patient safety, e.g., California has regulated mandatory nurse-to-patient ratios. To satisfy patient care demands, hospital administrations are obligated to employ other expensive staffing resources, such as part-time nurses, agency nurses, overtime nurses, etc. Since nurse staffing costs account for over 50% of hospital expenditures, health care costs are continuously increasing driven by an ongoing severe shortage of nurses. Consequently, the nursing shortage will become more severe and nurse staffing has become one of the most attractive research areas.

We describe four phases of nurse planning, which are nurse budgeting, nurse scheduling, nurse staffing, and nurse assignment. The first part of this dissertation considers

only the last phase of nurse planning, which makes daily decisions on assigning nurses to patients. With nurses from the nurse staffing phase, a charge nurse assigns nurses to patients at the beginning of a shift. To capture uncertainty in patient care, we develop a stochastic integer programming model for nurse assignment with an objective to minimize excess workload on nurses. The problem is solved using a Benders' decomposition approach, whose master problem assigns nurses to patients, and whose recourse subproblems penalize the assignment and determine excess workload on nurses. The recourse subproblems can be considered as network flow problems, in which we develop a new greedy algorithm to solve them. When hospital units have new hires, there is no sufficient data to consider them unique. A symmetry problem may arise when there are identical nurses, which leads us to construct sets of valid inequalities to strengthen the restricted master problem. We develop sets of valid inequalities to prevent symmetric assignments, which eventually reduce the computational effort. In addition, we develop a set of valid inequalities representing the nurse-to-patient ratio to ensure patient safety. These valid inequalities not only enhance the algorithmic performance, but also prevent illegal and impractical assignments.

The second part of this dissertation focuses on an integration between the third and the last phases, which makes short-term decisions (90 minutes before a shift) on staffing and assigning nurses to patients. We present a stochastic integer programming model for integrated nurse staffing and assignment with uncertain patient care with an objective of minimizing excess workload on nurses. We present three decomposition approaches based on the L-shaped method for solving our model, which are (1) Benders' decomposition, (2) Lagrangian relaxation with Benders' decomposition, and (3) nested Benders' decomposition. The Lagrangian relaxation with Benders' decomposition approach can be viewed as a novel search method for bicriteria stochastic integer programs.

Computational results are provided based upon data from two medical-surgical units at Northeast Texas hospital. The focus of this dissertation is to find good solutions within 30 minutes. Results suggest that the hospital can save up to 1588 hours of excess workload each year in each unit by using our stochastic programming for nurse assignment model. The greedy algorithm for the network primal subproblem is 30 times faster than the current commercial network simplex solver (CPLEX 9.1). Moreover, integrated nurse staffing and assignment results indicated the Lagrangian relaxation with Benders' decomposition approach provided the most promising results among the three methods. Considering our model as two-stage stochastic programming results in better nurse schedules and assignments than those from three-stage, meaning that it is more beneficial to perform nurse staffing and assignment simultaneously. The nurse staffs and assignments found by these methods can be used in a nurse staffing decision supporting system, which facilitates a nurse supervisor to select a nurse staff and assignment based on a tradeoff between staffing cost and excess workload on nurses. Moreover, a nurse supervisor can also use our model to evaluate a float assignment. Our model allows decision makers to play important roles in utilizing their judgments to comply the right staffing policy. Furthermore, topics of future research are discussed. Finally, a nurse assignment decision supporting tool based on our underlying model is provided in the appendix.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Overview of Nurse Planning

One of the greatest problems in health care today is a shortage of nurses. The demand for nurses is growing, while fewer young nurses are available to provide care. There were 118,000 vacant positions for Registered Nurses (RNs) in April 2006 [8]. A survey by the American Hospital Association found that 75% of vacant hospital staffing positions are for registered nurses [64]. Baby boomers [8] and an increasing elderly population, who requires a substantial amount of health care, are the main reasons for a rapid growing demand of nurses [100]. The number of citizens over 65 years old is expected to be 70 million in 2030, more than twice that of 1999 [1] and they need more health care services. Despite the current situation, the enrollments in registered nurse degree programs declined by 50,000 nurses from 1993 to 2001 [64]. The number of nurses per capita declined by 2% from 1996 to 2000, while the attrition rate of hospital nursing staff grew from 11.7% in 1998 to 26.2% in 2000 [45]. With fewer new nurses entering the profession, the average age of the working registered nurse is increasing [26]. From 1983 to 1998, the number of nurses under 30 years of age decreased by 41% [64]. Moreover, more than 40 % of nurses in the United States have suffered from work burnout and one-fifth of all nurses planned to leave their job within next year [6]. Due to the current low retention and low entering rates of nurses, the health care industry will need more than 1.2 million new and replacement nurses by 2014 [49]. Consequently, the shortage will become more severe. Buerhaus et al [26] predicted that by 2020 the United States will face a 20% shortage in the number of nurses needed in the nations health care system.

The nursing shortage has a direct effect on patient care. The National Survey on Consumers' Experiences with Patient Safety and Quality Information Consumers showed that the most important factors contributing to medical error are workload, stress, and fatigue of health professionals (74%); not enough time spent with patients (70%); and not enough nurses in health care systems (69%) [97]. A study by the Agency for Healthcare Research and Quality reported that nurses spend insufficient time with patients in hospitals with low staffing levels [94]. Powers [80] observed that excessive workload enhances poor quality of patient care. Patients with insufficient care have higher failure-to-rescue rates and risk adjusted 30-day mortality when they are cared for by nurses with too many assigned patients [5]. The nursing shortage not only affects patients, but it also has a significant impact on the quality of nursing work. Having received too many patients, nurses are more likely to suffer from job dissatisfaction and burnout [5]. One way to ease the burden of nursing shortage and improve the quality of nursing work is to balance the workload of nurses or reduce the excessive workload on nurses.

Nurses work in a variety of environments including hospitals, clinics, private doctors' offices, nursing homes, and individual homes. Nurses have a major responsibility to deliver care to patients in hospital units. Hospitals in the United States employ two types of nurses—*registered nurses (RNs)* and *licensed vocational nurses (LVNs).* We describe the four phases of nurse planning in Section 1.1.1 - 1.1.4.

1.1.1 Nurse Budgeting

The total cost expenditures of health care in the United States were nearly $2 trillion in 2004 [7], and expected to be $4 trillion by 2015 [47]. Nursing accounts for the largest portion of a hospital budgeting (over 50% as a whole) [60]. Consequently, nurse budgeting become an important issue for every hospitals and health care providers. In the nurse budgeting phase, financial planners create an annual budget and determine the number of nurses they will hire as full-time regular nurses, part-time nurses, and nurses from an agency. In general, they makes planing decision to satisfy cost control, predicted patients care, and regulations by nursing union such as weekends, week offs, holidays, sick leaves, vacation packages policies.

Warner [103] implemented a Markovian analysis to forecast nursing personnel for general wards of a hospital. Kao and Tung [58] predicted patient demands over a year by an autoregressive integrated moving average forecasting method. Dieck [39] compared the Box-Jenkins modeling and the Winters' heuristic approach for forecasting patients admission to public health facilities. Trivedi [98] developed a mixed integer goal programming model to optimize an annual budget for nurses. Kao and Queyranne [57] showed that a single-period demand estimate provided good approximation to the nursing budget for a hospital nursing unit. Martel and Ouellet [65] applied stochastic programming to allocate the budget of a nursing unit to different types of nurses.

1.1.2 Nurse Scheduling

The second phase of nurse planning is *nurse scheduling* or *nurse rostering*. A nurse manager forecasts the number of patients that will enter a hospital unit over four to six weeks. Based upon the forecasted number of patients, the manager uses a census matrix to determine the number and level of nurses needed. When the number of nurses of each type is known, a schedule is created that partitions a day into shifts that are typically 8 or 12-hours in length. Typically, the manager posts a schedule two weeks before the beginning of the time horizon. Nurse scheduling can be classified by a time horizon in which decisions are made [28]. We refer to nurse rostering as the mid term (several weeks) allocation of nurses to a working time period. Most nurse rostering literature can be found in Burke et al. [28], Cheang et al. [33], and Sitompul and Randhawa [91] providing nurse rostering survey papers. They summarized the overview of the nurse rostering model and the solution methodologies from the 1960's until 2004. Nurse rostering models and solution approaches included linear and integer programming [2, 42, 52, 55, 66, 67, 68, 84, 99, 104, 105, 106], goal programming/multi-criteria approaches [10, 11, 18, 29, 34, 44, 54, 71, 78, 79], artificial intelligence methods [34, 63, 74, 75, 87], heuristics [9, 23, 61], and metaheuristic, i.e., simulated annealing [24, 53], tabu search [27, 40, 41], genetic algorithms [3, 4, 30, 59]. Because these algorithms only consider the nurse budgeting and scheduling phases, they ignore changes in staff and patient forecasts and assume the schedule will be followed as planned. Anecdotal evidence suggests that changes to the schedule are frequent, so intelligent planning models to reschedule nurses will dramatically improve nurse planning.

1.1.3 Nurse Staffing

The third phase of nurse planning, nurse staffing, involves revising the set of nurses scheduled for a shift. The nurse staffing process occurs 90 minutes before each shift. A nurse supervisor reviews the scheduled nurses based upon the activities of the previous shift, activities of other units, the patients in the emergency room, and either a census matrix or a patient classification system. If there is a shortage of nurses for the upcoming shift, the supervisor tries to recruit additional nurses who work as needed–*PRN nurses*, nurses who work part time–*part-time nurses*, and nurses who are not scheduled for the upcoming shift–*off-duty nurses*. If an insufficient set of nurses agrees to work the shift, the supervisor, upon approval from a nurse manager, hires temporary agency nurses to satisfy the remaining shortage. If there are too many scheduled nurses for the shift than needed, then the supervisor has surplus PRN nurses and part-time nurses take the day off without pay.

Patient classification systems are the most sophisticated technology for *nurse rescheduling*. These systems group patients into one of several categories. They estimate how many times certain tasks will be performed in caring for a patient in each category. Using these estimates and the expected time required to perform each task, the systems determine the amount of time to care for a typical patient. As patients are admitted into the unit, the system classifies these patients, and nurse supervisors use the estimated patient care to determine how many nurses are needed for the shift in nurse rescheduling. As a patient's condition changes, he may be given a new patient classification. Although patient classifications systems provide benchmarks for nurse planning, they have several drawbacks as described in Section 2.3.

Nurse staffing has a direct impact on a nurse-to-patient assignment, nurse workload, and the quality of care for patients [5, 62]. Survey papers on nurse staffing and patient outcomes can be found at Lankshear et al. [62] and Curtin [35]. Lankshear et al. [62] summarized a relationship between nurse staffing and patient outcomes including mortality rates, failure to rescue, and complications. Hall et al. [48] presented nurse staffing models as predictors of patient outcomes and revealed a relationship between the mixture of nursing staffs and patients' self-reported outcomes. Curtin [35] suggested nurses can prevent patient complications by spending more time with patients. Accordingly, many states have seriously considered taking actions to cope with the shortage to ensure patient safety, for instance, Senate Bill 71, Registered Nurse Safe Staffing Act of 2005 requires a minimum number of registered nurses (RNs) on each shift in each unit to ensure the suitable staffing levels to provide patient care [93, 107]. California was the first state to regulate mandatory nurse-to-patient ratios [25, 31]. White [107] summarized minimum nurse-to-patient ratios and nurse staffing plans mandated to many states across the country. However, the relationship between the staffing ratios for registered nurse and patient care has not been investigated in the literature [16]. Cost effectiveness and several nurse-to-patient ratios were examined, and nurse staffing ratios of 1:4 was cost-effective to patient safety [85]. Behan [16] studied the effects of staffing ratio, patient diagnoses, direct and indirect nursing care time, and nurse level of education on weekend versus weekday staffing. Recently, nurse researchers addressed their concerns about nursing staffing in the near future [46]. The success of rescheduling in other industries, and challenges for nurse staffing were mentioned in Gardner and Gemme [46].

According to the optimization literature, most research on nurse staffing has emerged recently. Warner et al. [104] addressed a need for short-term staffing when nurses' unexpected absences occur. Abernathy et al. [2] integrated nurse scheduling and nurse staffing. Siferd and Benton [90] developed a stochastic model based upon the patients in a unit to determine how many nurses are required for the shift. Nevertheless, the different sets of skills among the nurses are ignored. Bard and Purnomo [13, 14, 15] presented deterministic integer programming models for daily nurse rescheduling, one of their papers [14] considered nurse rescheduling with nurse preference and implemented a branch-and-price algorithm to solve the problem. Moz and Pato described a nurse rerostering as a multi-commodity flow problem with an objective to minimize the difference between the original and new schedules [68]. They also constructed a genetic algorithm to cope with this problem [69]. Vericourt and Jennings [38] employed a queuing model to investigate nurse-to-patient ratios mandated by California and they proposed two heuristic staffing policies. They also described that hospitals with no nurse-to-patient ratio policy can provide consistently good quality of care for every units. Wright et al. [110] developed a bicriteria nonlinear integer programming model to evaluate the impact of nurse-to-patient ratios on schedule cost and nurses desirability.

1.1.4 Nurse Assignment

In the final phase of nurse planning, *nurse assignment*, a charge nurse assigns each patient to a nurse at the beginning of a shift. Typically, the nurse assignment has to be performed within 30 minutes before a shift. Although the charge nurse may update an assignment, in many hospital units, such as medical-surgical units, revised assignments only include assigning a nurse to a new admission; rarely is a patient reassigned a new nurse during the middle of a shift. Consequently, the initial assignment can determine the amount of workload given to each nurse during the shift. A nurse's *workload* is the amount of time required to care for her patients over a time period, and *excess workload* is the difference between the workload and the time available for care. In reality, excess workload results in other nurses assisting overworked nurses. One important consideration in nurse assignment is workload balance.

Developing balanced workloads for nurses is difficult because of the variation of patients' conditions [70]. In practice, most nurse assignments are based upon either an intuitive judgment or the caseload method, in which each nurse is assigned the same number of patients [88]. Modern patient classification systems partition the set of patients into groups, and each group is assigned to a nurse [77]. Walts and Kapadia [102] presented a patient classification system and optimization model to determine the level of staffing to meet the required workload level, but they did not use a detailed nurse assignment model. Mullinax and Lawley [70] developed an integer linear programming model that assigns patients to nurses in a neonatal intensive care unit. The nurseries are divided into a number of physical zones. They used a zone-based heuristic that assigns nurses to zones and computes patient assignments within each zone. Rosenberger et al. [83] presented an integer programming to assign nurses to patients. These approaches and patient classification systems ignore uncertainty, which is a major drawback considering the enormous variance in patient care. Punnakitikashem et al. [81] developed a two-stage stochastic integer programming model for a nurse-patient assignment that considered uncertainty in patient care. The model objective was to minimize excess workload for nurses. Punnakitikashem et al. [82] presented the nurse assignment decision supporting tool based on the optimization model and they reported positive feedbacks from users. Sundaramoorthi et al. [96] presented a simulation model from real data to evaluate nurse-patient assignments.

1.2 Overview of Stochastic Programming

Stochastic programming is a mathematical technique that has been widely used for many years in a variety of areas for including uncertainty within decision making models. It was first introduced by Dantzig [36] in 1955. General background can be found at Birge and Louveaux [22] and Kall and Wallace [56]. Stochastic programming has been employed into broad areas of applications, for example, finance, manufacturing, telecommunications, transportation, logistics, airline operations, capacity planning, and many more. Recent references on applications can be found at Birge [21] and Wallace and Ziemba [101].

We present the simplest formulation of a stochastic programming, namely, *a two-stage stochastic linear programs with fixed recourse* in Section 1.2.1. In the first stage, we solve the first-stage linear programming problem to obtain the *first-stage decisions*, which are decisions made without full information on some random events. When we obtain an information on the realization of some random vectors, we solve the *recourse problem* or the *second-stage linear programming problem* to receive the *second-stage decisions* or the *corrective actions*.

1.2.1 Two-Stage Stochastic Linear Programming with Fixed Recourse Formulation

The two-stage stochastic linear programming with fixed recourse can be formulated as:

$$\min c^T x + E_\xi \left[\min q(\omega)^T y(\omega)\right], \tag{1.1}$$

subject to

$$Ax = b, \tag{1.2}$$

$$T(\omega)x + Wy(\omega) = h(\omega), \tag{1.3}$$

$$x \geq 0, \tag{1.4}$$

$$y(\omega) \geq 0. \tag{1.5}$$

where $x \in R^{n_1}$ is the vector of first-stage decision variables. $y \in R^{n_2}$ is the vector of recourse or second-stage decision variables. $c \in R^{n_1}$ is the known objective coefficient vector of x. $A \in R^{m_1 \times n_1}$ is the known first-stage linear constraint matrix with the known right-hand side vector $b \in R^{m_1}$. ω represents random events. With the right-hand side matrix $h(\omega) \in R^{m_2}$, $T(\omega) \in R^{m_2 \times n_1}$ ($W \in R^{m_2 \times n_2}$) is called the technology matrix (recourse matrix), and it is the second stage linear constraint matrix associated with x ($y(\omega)$). $q(\omega) \in R^{n_2}$ is an objective coefficient matrix of vector $y(\omega)$. $\xi \in \Xi$ is a random vector representing each scenario ξ with realizations ω. $\Xi \in R^N$ is a support set of the random vector ξ where $N = n_2 + m_2 + (m_2 \times n_1)$. E_ξ is expectation with respect to the scenarios $\xi \in \Xi$.

To simplify the problem and computation, we assume W is fixed here. The objective of the model is to minimize current cost and the expected value of future corrective actions. The first-stage decision variables x are first made. After random events $\omega \in \Omega$ are realized, problem data $h(\omega), q(\omega)$, and $T(\omega)$ in the second stage problem are known. Then, the second-stage decision variables $y(\omega)$ or recourse actions are taken. Note that $P\{\xi \in \Xi\}$, the probability of all random vectors that have finite support is equal to one. One example of extensions of this above model is a two-stage stochastic integer programming, where the first stage or second stage decision variables are restricted to be integers. Theoretical properties and algorithmic solution approaches for stochastic programming have been extensively studied [22].

Many solution methodologies in literature like decomposition, Lagrangian-based, and other direct methods using a particular structure are applied to solve stochastic programming problems. It is well known that two-stage stochastic programming with recourse can be solved by decomposition, which is described in Section 1.3. Literature survey on stochastic programming computational implementations is included in Birge [21].

1.3 Overview of Decomposition

The difficulty of stochastic programming is the computation of the recourse problem. As the number of realizations increases, the size of the recourse problem becomes larger resulting in computational intractable problems. In general, the recourse problems are linear programming problems. One way to handle the large-scale linear programming problem is to take advantage of problem structures. The large problem with appropriate structure can be separated into one general problem, called the *master problem*, and many small problems, called the *subproblems*. The systematic procedure will solve and pass the information between the master problem and the subproblems until either the optimal solutions are found or the terminating criteria is reached. The decomposition principle helps to solve the large problems efficiently.

The decomposition approaches are decided based upon the appropriate structure of the problems. The general idea can be presented as the following cases:

Case 1: The problems contain the complicating or excessive constraints. We decompose the constraints into the subproblems and solve them separately. The constraint is added to the master problem when it is violated by current solutions.

Case 2: The problems contain the complicating or excessive variables. We decompose the variables into the subproblems and solve them separately. When there is a variable (column) which can improve the objective function value by having negative reduced cost, the variable is generated and added to the master problem.

We briefly describe two famous decomposition methods for large-scale linear programming problems, which are Dantzig-Wolfe decomposition and Benders' decomposition in Section 1.3.1 and 1.3.2, respectively.

1.3.1 Dantzig-Wolfe Decomposition

Dantzig-Wolfe decomposition was introduced in 1960 [37]. The block angular structure problem with complicating or an excessive number of constraints is appropriate for Dantzig-Wolfe decomposition. Figure 1.1 depicts the block angular problem structure for this approach. The constraints of problems can be categorized into easy and hard constraints. We separate a linear programming problem into the master problem and the subproblems. Dantzig-Wolfe decomposition works best when the subproblems can be solved efficiently.

The general form of linear programming can be written as the following:

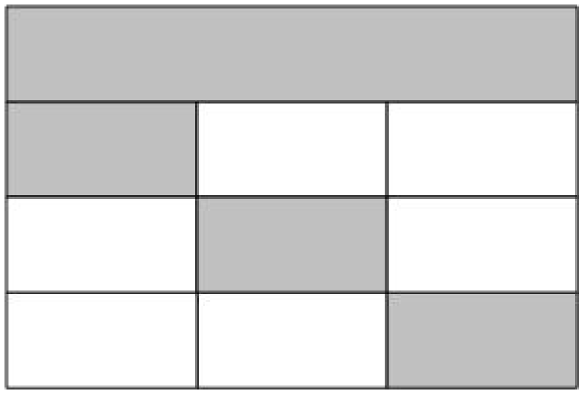


Figure 1.1. The block angular structure for the Dantzig-Wolfe decomposition.

$$\max c^T x, \tag{1.6}$$

subject to

$$A'x = b', \tag{1.7}$$

$$A''x = b'', \tag{1.8}$$

$$x \geq 0. \tag{1.9}$$

where $x \in R^{n_1}$ is a decision vector with the objective function coefficient vector $c \in R^{n_1}$. $A' \in R^{m_1 \times n_1}$ ($A'' \in R^{m_2 \times n_1}$) is a linear constraint matrix with right-hand side vector $b' \in R^{m_1}$ ($b'' \in R^{m_2}$). Let constraints (1.7) and (1.8) be hard and easy constraints, respectively. We maximize the objective function (1.6) subject to constraint sets (1.7)-(1.8). The linear programming problem (1.6)-(1.9) can be rewritten as:

$$\max c^T x, \tag{1.10}$$

subject to

$$A'x = b', \tag{1.11}$$

$$x \in \{x | A''x \leq b'', x \geq 0\}. \tag{1.12}$$

By the Minkowski's theorem, the polyhedron can be represented in term of extreme points and extreme rays. Let K and J be sets of extreme points and extreme rays, respectively. For each $k \in K$, let x^k be an extreme point. For each $j \in J$, let r^j be an extreme ray. For a polyhedron $x \in \{x|A''x \leq b''\}$, let $x = \sum_{k\in K} \lambda_k x^k + \sum_{j\in J} \mu_j r^j, \sum_{k\in K} \lambda_k = 1, \lambda_k \geq 0, \forall k \in K, \mu_j \geq 0, \forall j \in J$. By substitute x into (1.10)-(1.12), the reformulation or the *master problem* can be written as follows:

$$\max \sum_{k\in K} \lambda_k c^T x^k + \sum_{j\in J} \mu_j c^T r^j, \tag{1.13}$$

subject to

$$\sum_{k\in K} \lambda_k A' x^k + \sum_{j\in J} \mu_j A' r^j, \leq b', \tag{1.14}$$

$$\sum_{k\in K} \lambda_k = 1, \tag{1.15}$$

$$\lambda_k \geq 0 \qquad \forall k \in K, \tag{1.16}$$

$$\mu_j \geq 0 \qquad \forall j \in J. \tag{1.17}$$

The *subproblem* can be written as:

$$\max(c^T - yA')x, \tag{1.18}$$

subject to

$$A''x = b'', \tag{1.19}$$

$$x \geq 0. \tag{1.20}$$

In Dantzig-Wolfe decomposition, the number of constraints is reduced at the expense of large number of extreme point and extreme ray variables. Given that the number of column in the master problem is large, we can employ the *delayed column generation* algorithm to handle it. We can solve the linear programming problem (1.6)-(1.9) by using the Dantzig-Wolfe decomposition algorithm described in Algorithm 1. First, we solve the master problem. Its dual multiplier information is passed to the subproblem. The subproblems are then solved for the extreme points and extreme rays. If the extreme point or extreme ray have promising reduced cost, then the column is added to the master problem to improve the objective function. Otherwise, the algorithm terminates and the optimal solution is obtained.

Algorithm 1 Dantzig-Wolfe Decomposition Algorithm.

Consider a subset of extreme points $\overline{K} \subset K$, and extreme rays $\overline{J} \subset J$.

$STOP \leftarrow FALSE$.

while $STOP = FALSE$ **do**

  Solve the master problem (1.13)-(1.17).

  Solve the subproblem (1.18)-(1.20) to find additional extreme points $(\tilde{K})$ and extreme rays $(\tilde{J})$.

  if no more extreme points and rays are found **then**

    $STOP \leftarrow TRUE$.

  end if

  Add new extreme points to $\overline{K} \leftarrow \overline{K} \cup \{\tilde{K}\}$ and extreme rays to $\overline{J} \leftarrow \overline{J} \cup \{\tilde{J}\}$.

end while

1.3.2 Benders' Decomposition

Benders' decomposition was developed in 1962 [17]. It is appropriate for the block angular problem structure with complicating or excessive number of variables. This problem structure is displayed in Figure 1.2. Benders' decomposition can be view as applying the Dantzig-Wolfe decomposition to the dual problem. The general idea is it reduces the number of variable by employing the delayed constraint generation procedure.

In Benders' decomposition, we separate a problem into two simpler problems, namely, the *master problem* and the *subproblems*. The master problem or the first stage problem contains a portion of original variables and their associated constraints. The subproblems or recourse problems (for stochastic programming problem) are the remaining problem with fixed first stage variables. We first solve the master problem. Given the

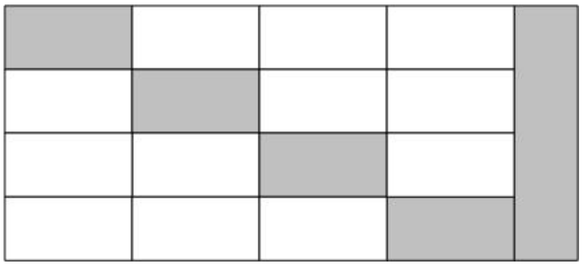


Figure 1.2. The block angular structure for the Benders' decomposition.

master solution, the original problem decomposes into many manageable size subproblems. Benders' decomposition works best when the subproblems are easy to solve. In general, the subproblems decompose by the number of scenarios. Each recourse problem penalizes the solution from the master problem. Benders' decomposition is an iterative procedure, which passes information back and forth between the master problem and the subproblem until an optimal solution is found.

Consider the following linear programming formulation:

$$z = \min c^T x + h^T y, \tag{1.21}$$

subject to

$$Ax + Gy \geq b, \tag{1.22}$$

$$y \in R_+^{n_2}, \tag{1.23}$$

$$x \in X \subseteq Z_+^n. \tag{1.24}$$

where $x \in Z_+^{n_1}$ ($y \in R_+^{n_2}$) is the decision variable with linear objective coefficient vector $c \in R^{n_1}$ ($h \in R_+^{n_2}$). $A \in R^{m_1 \times n_1}$ and $G \in R^{m_1 \times n_2}$ are linear constraint matrices with the known right hand side vector $b \in R^{m_1}$. The problem (1.21)-(1.24) is equivalent to the following problem:

$$z = \min c^T x + \eta(x), \tag{1.25}$$

subject to

$$Ax \geq b, \tag{1.26}$$

$$x \geq 0. \tag{1.27}$$

where $\eta(x)$ is the minimum value of the following:

$$\eta(x) : \min h^T y, \tag{1.28}$$

subject to

$$Gy \geq b - Ax, \tag{1.29}$$

$$y \geq 0. \tag{1.30}$$

Let $u \in R^{m_1}$ be a dual variable associating with constraint (1.29). By the duality theory, the primal and dual problems are interchanged. The dual problem of (1.28)-(1.30) can be presented as follows:

$$\max (b - Ax)^T u, \tag{1.31}$$

subject to

$$Gu \leq h, \tag{1.32}$$

$$u \geq 0. \tag{1.33}$$

Let $F = \{u : Gu \leq h, u \geq 0\}$. Assuming that the dual polyhedron F is nonempty, which implies that a primal subproblem is either infeasible or unbounded. Consequently, the *Benders reformulation* or the *master problem* can be written as:

$$\min c^T x + \eta(x), \tag{1.34}$$

subject to

$$Ax \geq b, \tag{1.35}$$

$$\eta(x) \geq u^k(b - Ax) \qquad k \in K, \tag{1.36}$$

$$0 \geq v^j(b - Ax) \qquad j \in J, \tag{1.37}$$

$$x \in X. \tag{1.38}$$

where $u^k, \forall k \in K$ are extreme points of polyhedron $\{u^k | A^T u^k \leq h, u^k \geq 0\}$ and $v^j, \forall j \in J$ are extreme rays. The drawback of Benders reformulation is that the number of extreme points and extreme rays is large, therefore delayed constraint generation is used. Constraints (1.36) and (1.37) are called *optimality and feasibility cut*, respectively.

We can solve the linear programming problem (1.21)-(1.24) by using the Benders' decomposition algorithm displayed in Algorithm 2. First, we solve the master problem to obtain the solution. Given the solution, we solve the subproblems, and pass information from the dual subproblems to the master problem by adding new constraints. When the dual subproblems are unbound, we add constraints with respect to their extreme rays, these constraints are called *feasibility cuts.* When the dual subproblems produce optimal solutions, we incorporate the constraint associating with their extreme points information to the master problem, called *optimality cuts.* If all constraints in the master problem are satisfied, then we terminate the algorithm and the optimal solution of the original problem is obtained. Otherwise, we iteratively perform these procedures.

More information about the stochastic decomposition can be found in Higle and Sen [51].

Algorithm 2 Benders' Decomposition Algorithm.

$\overline{K} \leftarrow \emptyset$, $\overline{J} \leftarrow \emptyset$, $STOP \leftarrow FALSE$.

while $STOP = FALSE$ **do**

Solve the restricted master problem (1.34), (1.35), (1.36'), (1.37'), (1.38) to obtain the solution $\overline{X}$ and an anticipated objective value $\overline{\eta}$. (On the first iteration, let $\overline{\eta} \leftarrow -\infty$, and let $\overline{X}$ be a feasible solution.)

if the restricted master problem is infeasible **then**

$STOP \leftarrow TRUE$. The problem is infeasible.

end if

Solve the subproblem (1.31)-(1.33).

if the subproblem is unbound **then**

Get an extreme ray $\tilde{v}^j$.

$\overline{J} \leftarrow \overline{J} \cup \{\tilde{v}^j\}$.

Add a feasibility cut $\tilde{v}^j(b - Ax) \leq 0$ to the master problem.

else

Get an extreme point $\tilde{u}$.

if $\overline{\eta} < \tilde{u}^k(b - A\overline{x})$ **then**

$\overline{K} \leftarrow \overline{K} \cup \{\tilde{u}^k\}$.

Add an optimality cut $\eta(x) \geq \tilde{u}^k(b - Ax)$ to the master problem.

else

$STOP \leftarrow TRUE$.

end if

end if

end while

1.4 Research Overview / Contributions

The objectives of my work can be stated as the following:

1. Comprehensive study of nurse staffing and nurse assignment background.
2. Development of a stochastic programming model for nurse assignment.
    - Development of a stochastic programming model formulation for nurse assignment problem.
    - Development of a solution methodology for solving the stochastic programming model for nurse assignment.
        - An algorithm for solving the stochastic programming model for nurse assignment.
        - Valid inequalities to enhance algorithmic performance.
    - Computational study. Comparison of our methodology to current assignment approaches.
    - Investigation of expected value of perfect information and value of the stochastic solution of our algorithmic approach.
3. Development of a stochastic programming model for integrated nurse staffing and assignment.
    - Development of a stochastic programming model formulation for integrated nurse staffing and assignment.
    - Development of algorithms for solving the stochastic programming model for integrated nurse staffing and assignment.
    - Computational study, which includes the following:
        - Parameter tuning.
        - Algorithmic approaches enhancement.
        - Comparison of solution methodologies.
        - An evaluation of float assignment policies.
4. Development of an optimization-based Information Technology (IT) prototype for nurse assignment in hospital units.

- Overall application framework design.
- Identification of input and output data needed for the IT prototype.
- Structure of optimization-based IT prototype design.
- Development of an underlying model of the IT prototype.
- Development of an IT prototype manual and trouble shooting documentation.
- Implementation and training to the potential users.
- Summary of implementation results from surveys and areas of improvement discussions.

CHAPTER 2

NURSE ASSIGNMENT PROBLEM FORMULATIONS AND SOLUTION ALGORITHMS

A nurse-patient assignment is a mandatory routine for all health care units in almost every hospital in the world, and it is performed daily for every shift for the entire year. At the beginning of a nursing shift, a charge nurse assigns each nurse to a set of patients for a shift. Since some patients require more care than others, tending to the needs of a few patients may consume most of a nurse's time, while other patients may receive only minimal care. During the same shift another nurse may have significantly less workload because his (her) patients may require less care. The second nurse should assist the first by taking on some of the first nurse's patients. Similarly, there can also be differences in the skills of the nurses, so assigning the right nurses to the right patients can reduce excess workload.

In this chapter, we focus on the last phase of nurse planning, in which a charge nurse makes daily decisions on assigning nurses to patients. We develop the nurse-patient assignment models with the objective to minimize the excess workload on nurses. We begin the chapter with the model assumptions in Section 2.1. In Section 2.2, the simple deterministic model for nurse assignment is presented. In Section 2.3, we propose a stochastic programming model, which is an extensive model including uncertainty and fluctuation in patient care. In Section 2.4, we present an algorithmic approach to solve a stochastic programming model for nurse assignment.

2.1 Model Assumptions

Prior to the beginning of a shift, a charge nurse assigns each patient to an RN or an LVN. Although patients can usually be nursed by either type of nurse, state regulations can preclude LVNs from performing certain patient care. Furthermore, some states, such as Texas, require that every patient be assessed by an RN within any 24-hour time period. Consequently, a charge nurse will assign RNs to patients who were assigned LVNs in the previous shift. We assume:

Assumption A1. A charge nurse determines which nurses can be assigned to which patients before optimizing nurse assignment.

Because patients enter and leave the hospital unit throughout a shift, nurse assignments are updated dynamically. However, revised nurse assignments often only include assigning a nurse to a new admission. Rarely is a patient reassigned a new nurse during the middle of a shift due to concerns for continuity of care. Hence, we make the following assumption:

Assumption A2. Nurse assignments are not changed, except when there are newly admitted patients.

Nurses distinguish between two types of patient care. *Direct care* is the amount of time nurses spend with patients, while *indirect care* is time spent on other tasks for patients, such as documentation of a patient's condition. In our stochastic programming model, we divide a nurse shift into several smaller *time periods.* The amount of direct and indirect care the patients require in each time period are given as parameters to the model. Nurses often provide indirect care throughout the shift, but direct care is often determined by a patient's condition, which is usually more urgent. Consequently, we make the following assumption:

Assumption A3. Direct care needs to be performed within the given time period, while indirect care can be performed in any time period from the given period until the end of the shift.

In addition to assumption A3, we assume nurses optimally allocate their indirect care to minimize excess workload. In some assignments, a nurse's patients will require more care than the nurse can provide. In such cases, a charge nurse, a nurse aide, or another nurse may assist the overworked nurse. However, an assignment requiring such assistance is undesirable. Implicitly, we presume that nurses receive assistance when absolutely necessary. The penalty of an assignment will be determined by a nondecreasing piecewise-linear convex function. Because the function penalizes assignments with overworked nurses, an assignment will not include overworked nurses if such a solution exists.

During a shift patients may enter the hospital unit by admission from an emergency room, direct admission from a doctor, transferring from another unit, or birth. Patients may leave by discharge, transferring to another unit, or death. After a patient is discharged and his room has been cleaned and sterilized, a charge nurse may assign a newly admitted patient to the original patient's room. The charge nurse will often assign the nurse who cared for the recently discharged patient to the newly admitted patient. She can anticipate some of the patients that will be admitted because they are currently in another hospital unit. However, an *unanticipated patient* may enter a hospital unit during a shift without any warning prior to the shift. Unanticipated patients must be assigned a nurse, so we include them in the set of patients. We can represent an unknown number of patients by increasing the number of patients and randomly allowing their required care to be zero. Similarly, we can model random times for admissions and discharges. In this dissertation, we assume:

Assumption A4. The set of patients to be assigned includes potential unanticipated patients, so the number of patients is fixed.

2.2 Deterministic Model of Nurse Assignment

Rosenberger et al. [83] presented a deterministic model for patient assignment. We propose an alternative deterministic model assigning nurse to patient with an objective to minimize the excess workload on nurses in this section.

Let P and N be the sets of patients and nurses for a shift, respectively. We assume that a charge nurse determines which nurses can be assigned to which patients before optimizing patient assignment. For each patient $p \in P$, let $N(p)$ be the set of nurses which can be assigned to patient p. For each nurse $n \in N$, let $P(n)$ be the set of patients that can be assigned to nurse n; that is, $P(n) = \{p \in P | n \in N(p)\}$. For each patient $p \in P$, and nurse $n \in N(p)$, let *assignment variable*

$$X_{pn} = \begin{cases} 1 & \text{if patient } p \in P \text{ is assigned to nurse } n \in N(p), \\ 0 & \text{otherwise.} \end{cases}$$

A shift is divided into a set of time periods T. As the workload of a nurse increases in a time period $\tau \in T$, her patients receive less care, which is unsafe. We model the penalty for assigning workload to nurses as a monotonically nondecreasing piecewise linear convex function with k pieces.

For each time period $\tau \in T$ and each nurse $n \in N$, let $A_{\tau ni}$ be the amount of workload assigned to nurse n between time durations $m_{\tau ni}$ and $m_{\tau n(i+1)}$. Let $\alpha_{\tau ni}$ be the marginal penalty of $A_{\tau ni}$ for $1 \le i \le k$. Because the penalty is monotonically nondecreasing, $0 = m_{\tau n1} < \ldots < m_{\tau nk}$ and $0 \le \alpha_{\tau n1} < \ldots < \alpha_{\tau nk}$. For notation, let $m_{\tau n(k+1)}$ be ∞. This penalty function is nondecreasing and piecewise linear convex, so the marginal penalty for assigning more patient care to an overworked nurse is greater than that of a nurse with less workload. Consequently, the function naturally balances the workload and allows nurses to provide better care. One special case of the penalty function has $k = 2$, $\alpha_{\tau n1} = 0$, $\alpha_{\tau n2} = 1$, and $m_{\tau n2}$ equal to the duration of the time period τ for each $\tau \in T$ and each $n \in N$. We refer to the value of variable $A_{\tau n2}$ as the *excess workload* on nurse n in time period τ. The objective of our model in the computational results in Section 3 is to minimize the expected excess workload on nurses.

For each patient $p \in P$, and each $t \in T$, let d_{tp} be the amount of direct care required by patient p in time period t. Because patient p may be admitted or discharged during a shift, the patient care may vary dramatically throughout the shift. For each patient $p \in P$, and each time period $t \in T$, let g_{tp} be the amount of indirect care required by patient p at the beginning of time period t until the end of the shift. For each pair of time periods $(t, \tau) \in T \times T$, where $t \le \tau$, and each nurse $n \in N$, let *indirect workload variable* $G_{t\tau n}$ be the total indirect care that can be performed during or after time period

t and is performed in time period τ by nurse n. The amount of direct and indirect care the patients require in each time period under each scenario are given as parameters to the model.

The deterministic programming model for nurse assignment (DNA) is formulated as:

$$\min \sum_{n\in N}\sum_{\tau\in T}\sum_{i=1}^{k} \alpha_{\tau ni} A_{\tau ni} \tag{2.1}$$

$$\sum_{n\in N(p)} X_{pn} = 1 \qquad \forall p \in P, \tag{2.2}$$

$$\sum_{p\in P(n)} g_{tpn} X_{pn} = \sum_{\tau=t}^{|T|} G_{t\tau n} \qquad \forall t \in T, n \in N, \tag{2.3}$$

$$\sum_{p\in P(n)} d_{\tau pn} X_{pn} + \sum_{t=1}^{\tau} G_{t\tau n} = \sum_{i=1}^{k} A_{\tau ni} \qquad \forall \tau \in T, n \in N, \tag{2.4}$$

$$X_{pn} \in \{0, 1\} \qquad \forall p \in P(n), n \in N, \tag{2.5}$$

$$G_{t\tau n} \geq 0 \qquad \forall t, \tau \in T, t \leq \tau, n \in N, \tag{2.6}$$

$$m_{\tau n(i+1)} - m_{\tau ni} \geq A_{\tau ni} \geq 0 \qquad \forall \tau \in T, 1 \leq i \leq k, n \in N. \tag{2.7}$$

Objective (2.8) minimizes the workload penalty on nurses. The first constraint set—the *nurse assignment constraints* (2.9)— ensures that every patient is assigned to a nurse. The *indirect care constraints* in set (2.10) determine the total indirect care performed by nurse n from the beginning of time period t until the end of the shift. For each time period $\tau \in T$, the workload of nurse $n \in N$ consisting of direct care and indirect care is defined by a *workload constraint* in set (2.11). Constraint set (2.12) requires that the assignment variables be binary, and set (2.13) ensures the indirect care variables are nonnegative. Constraints (2.14) give the upper and lower bounds on the marginal workload variables. Observe that for each $\tau \in T, n \in N$, $A_{\tau nk}$ has no upper bound since $m_{\tau n(k+1)} = \infty$.

2.3 Stochastic Model for Nurse Assignment

Let Ξ be a set of random scenarios, and for each $\xi \in \Xi$, let ϕ^ξ be the probability that scenario ξ occurs. For each time period $\tau \in T$ and each nurse $n \in N$, let $A^\xi_{\tau ni}$ be the amount of workload assigned to nurse n between time durations $m_{\tau ni}$ and $m_{\tau n(i+1)}$ in scenario $\xi \in \Xi$.

For each patient $p \in P$, each scenario $\xi \in \Xi$, and each $t \in T$, let d^ξ_{tp} be the amount of direct care required by patient p in time period t. For each patient $p \in P$, each scenario $\xi \in \Xi$, and each time period $t \in T$, let g^ξ_{tp} be the amount of indirect care required by patient p at the beginning of time period t until the end of the shift. For each pair of time periods $(t, \tau) \in T \times T$, where $t \leq \tau$, and each nurse $n \in N$, let *indirect workload variable* $G^\xi_{t\tau n}$ be the total indirect care that can be performed during or after time period t and is performed in time period τ by nurse n.

2.3.1 Extensive Form of the Stochastic Programming Model

The *extensive form* of the stochastic programming model for patient assignment (SPA) is formulated as

$$\min \sum_{\xi \in \Xi} \sum_{n \in N} \sum_{\tau \in T} \sum_{i=1}^{k} \phi^\xi \alpha_{\tau ni} A^\xi_{\tau ni} \tag{2.8}$$

$$\sum_{n \in N(p)} X_{pn} = 1 \qquad \forall p \in P, \tag{2.9}$$

$$\sum_{p \in P(n)} g^\xi_{tpn} X_{pn} = \sum_{\tau=t}^{|T|} G^\xi_{t\tau n} \qquad \forall t \in T, n \in N, \xi \in \Xi, \tag{2.10}$$

$$\sum_{p \in P(n)} d^\xi_{\tau pn} X_{pn} + \sum_{t=1}^{\tau} G^\xi_{t\tau n} = \sum_{i=1}^{k} A^\xi_{\tau ni} \qquad \forall \tau \in T, n \in N, \xi \in \Xi, \tag{2.11}$$

$$X_{pn} \in \{0, 1\} \qquad \forall p \in P(n), n \in N, \tag{2.12}$$

$$G^\xi_{t\tau n} \geq 0 \qquad \forall t, \tau \in T, t \leq \tau, n \in N, \xi \in \Xi, \tag{2.13}$$

$$m_{\tau n(i+1)} - m_{\tau ni} \geq A^\xi_{\tau ni} \geq 0 \qquad \forall \tau \in T, 1 \leq i \leq k, n \in N, \xi \in \Xi. \tag{2.14}$$

2.3.2 Deterministic Equivalent Model of the Stochastic Programming Model

The *deterministic equivalent model* for patient assignment can be written as follows:

$$\min Q(X) \tag{2.15}$$

$$\sum_{n \in N(p)} X_{pn} = 1 \qquad \forall p \in P, \tag{2.16}$$

$$X_{pn} \in \{0,1\} \qquad \forall p \in P(n), n \in N, \tag{2.17}$$

where $Q(X)$ is the *expected second-stage recourse function* defined as:

$$Q(X) = E_\xi Q(X,\xi), \tag{2.18}$$

$$\text{and} \qquad Q(X,\xi) = \min \sum_{n \in N} \sum_{\tau \in T} \sum_{i=1}^{k} \alpha_{\tau ni} A^\xi_{\tau ni} \tag{2.19}$$

$$\sum_{p \in P(n)} g^\xi_{tpn} X_{pn} = \sum_{\tau=t}^{|T|} G^\xi_{t\tau n} \qquad \forall t \in T, n \in N, \tag{2.20}$$

$$\sum_{p \in P(n)} d^\xi_{\tau pn} X_{pn} + \sum_{t=1}^{\tau} G^\xi_{t\tau n} = \sum_{i=1}^{k} A^\xi_{\tau ni} \qquad \forall \tau \in T, n \in N, \tag{2.21}$$

$$G^\xi_{t\tau n} \geq 0 \qquad \forall t, \tau \in T, t \leq \tau, n \in N, \tag{2.22}$$

$$m_{\tau n(i+1)} - m_{\tau ni} \geq A^\xi_{\tau ni} \geq 0 \qquad \forall \tau \in T, 1 \leq i \leq k, n \in N. \tag{2.23}$$

Proposition 1 is obvious. To simplify notation, we ignore $\xi \in \Xi$, $\tau \in T$, $n \in N$, and let $\hat{\mathcal{A}}$ represent $A^\xi_{\tau n}$.

Proposition 1. *Let (X^*, A^*, G^*) be an optimal solution to SPA. Then there exists a positive integer $l \leq k$ such that*

$$\hat{\mathcal{A}}^*_i = \begin{cases} m_{(i+1)} - m_i & 1 \leq i < l, \\ \sum_{j=1}^{i} \hat{\mathcal{A}}^*_j - m_i & i = l, \\ 0 & l < i \leq k. \end{cases} \tag{2.24}$$

Given an assignment $\overline{X}$, the constraints in (2.10), (2.11), (2.13), and (2.14) can be decomposed by nurse and scenario resulting in $|N| \times |\Xi|$ recourse subproblems. In

Section 2.4, we implement Benders' decomposition to solve SPA. Although typical real-world problems cannot be solved to optimality within 30 minutes, the remainder of this dissertation focuses on finding a good solution within the time limit.

2.3.3 Side Constraints

In this section,we present side constraints for nurse assignment model. However, the computational results in Section 3 exclude constraints in this section.

2.3.3.1 Acuity Constraints

Typically, patients in the same unit have different acuity levels. Assignments that balance the number of patients with certain level of acuity are desirable. Let P^4 be a set of patients with a certain level of acuity.

$$\sum_{p \in P^4(n_1)} X_{pn_1} \leq \sum_{p \in P^4(n_2)} X_{pn_2} + 1 \qquad \forall (n_1, n_2) \in N \times N, n_1 \neq n_2. \tag{2.25}$$

For a specific pair of nurses n_1 and n_2, constraints in set (2.25) ensure that one nurse has at least one patient with certain acuity or more than another nurse.

2.3.3.2 Preceptor Constraints

It is common for a hospital unit to have a preceptor. In general, the preceptor receives less patients than other nurses.

$$\sum_{p \in P(n_1)} X_{pn_1} \leq \sum_{p \in P(n_2)} X_{pn_2} + 2 \qquad \forall (n_1, n_2) \in N \times N, n_1 \neq n_2, \tag{2.26}$$

where n_1 represents a preceptor.

Constraints in set (2.26) indicate that every nurse is assigned to at least two patients or more than the preceptor.

2.3.3.3 Charge Nurse Constraints

Anecdotal evidence suggests that a charge nurse usually receives less number of patients than other nurses.

$$\sum_{p \in P(n_1)} X_{pn_1} \leq \sum_{p \in P(n_2)} X_{pn_2} \qquad \forall n_2 \in N \backslash \{n_1\} \tag{2.27}$$

$$\sum_{p \in P(n_1)} X_{pn_1} \leq p^{max}, \tag{2.28}$$

where n_1 represents a charge nurse, and p^{max} represents the maximum number of patients for the charge nurse. In general, p^{max} equals to 3. Constraints in set (2.27) guarantee that every nurse has more patients than the charge nurse. Constraint (2.28) restricts the maximum number of patients assigned to the charge nurse for a shift.

2.3.3.4 Room Exception Constraints

Assigning a nurse to two patients in distant rooms can increase workload on nurses. Assignments that exclude two distant rooms are preferred.

$$X_{p_1 n} + X_{p_2 n} \leq 1 \qquad \forall n \in N. \tag{2.29}$$

For a specific pair of patients p_1 and p_2, constraint set (2.29) ensures that assigning both patients to same a nurse does not occur.

2.4 Algorithmic Approach

In this section, we present a Benders' decomposition approach to solve SPA. Moreover, we develop an optimal greedy algorithm for solving the recourse subproblems, and then we discuss sets of valid inequalities to improve the overall algorithmic performance.

2.4.1 Benders' Decomposition

Solving SPA with many scenarios and many time periods using branch and bound may be time consuming. However, two-stage stochastic programming models, like SPA, have a block angular structure that is appropriate for mathematical decomposition. The standard L-shaped method, based upon Benders' decomposition, is the most common solution approach for two-stage stochastic programming problems [22, 32]. Applying Benders' decomposition to SPA, the master problem assigns nurses to patients, and each recourse problem penalizes the assigned workload. Not only does SPA decompose by scenario like the standard L-shaped method, but it also decomposes by nurse into $|N| \times |\Xi|$ linear programming subproblems. Therefore, the subproblems are even more manageable than the standard L-shaped method, which only decomposes by scenario. Let $\overline{X}$ be a given assignment. For each $t \in T$, let $\overline{g}^{\xi}_{tn} = \sum_{p \in P(n)} g^{\xi}_{tpn} \overline{X}_{pn}$, and let $\overline{d}^{\xi}_{tn} = \sum_{p \in P(n)} d^{\xi}_{tpn} \overline{X}_{pn}$. The *primal subproblem* (PS^{ξ}_{n}) for each nurse $n \in N$ and each scenario $\xi \in \Xi$ is given by

$$\min \sum_{\tau \in T} \sum_{i=1}^{k} \alpha_{\tau n i} A^{\xi}_{\tau n i} \tag{2.30}$$

$$\sum_{\tau=t}^{|T|} G^{\xi}_{t\tau n} = \overline{g}^{\xi}_{tn} \qquad \forall t \in T, \tag{2.31}$$

$$\sum_{i=1}^{k} A^{\xi}_{\tau n i} - \sum_{t=1}^{\tau} G^{\xi}_{t\tau n} = \overline{d}^{\xi}_{\tau n} \qquad \forall \tau \in T, \tag{2.32}$$

$$(A^{\xi}_{n}, G^{\xi}_{n}) \text{ satisfy (2.13) and (2.14).}$$

In the primal subproblem, the workload variables $A^{\xi}_{\tau n i}$ are obtained, and the indirect care variables $G^{\xi}_{t\tau n}$ determine the time periods in which indirect care is performed. This problem always has a feasible solution $(\tilde{A}, \tilde{G})$ given by $\tilde{G}^{\xi}_{ttn} = \overline{g}^{\xi}_{tn}$ and $\tilde{A}^{\xi}_{tnk} = \tilde{G}^{\xi}_{ttn} + \overline{d}^{\xi}_{tn}$ for all $t \in T$, and all other variables are zero.

Each primal subproblem PS^{ξ}_{n} can be formulated as a network flow problem, as depicted in Figure 2.1. Consider a directed network $G = (\mathcal{N}, \mathcal{A})$ with node set $\mathcal{N}$ and arc set $\mathcal{A}$, in which $|\mathcal{N}| = (2+k)|T| + 1$ and $|\mathcal{A}| = |T|(|T|+1)/2 + 2k|T|$. The network

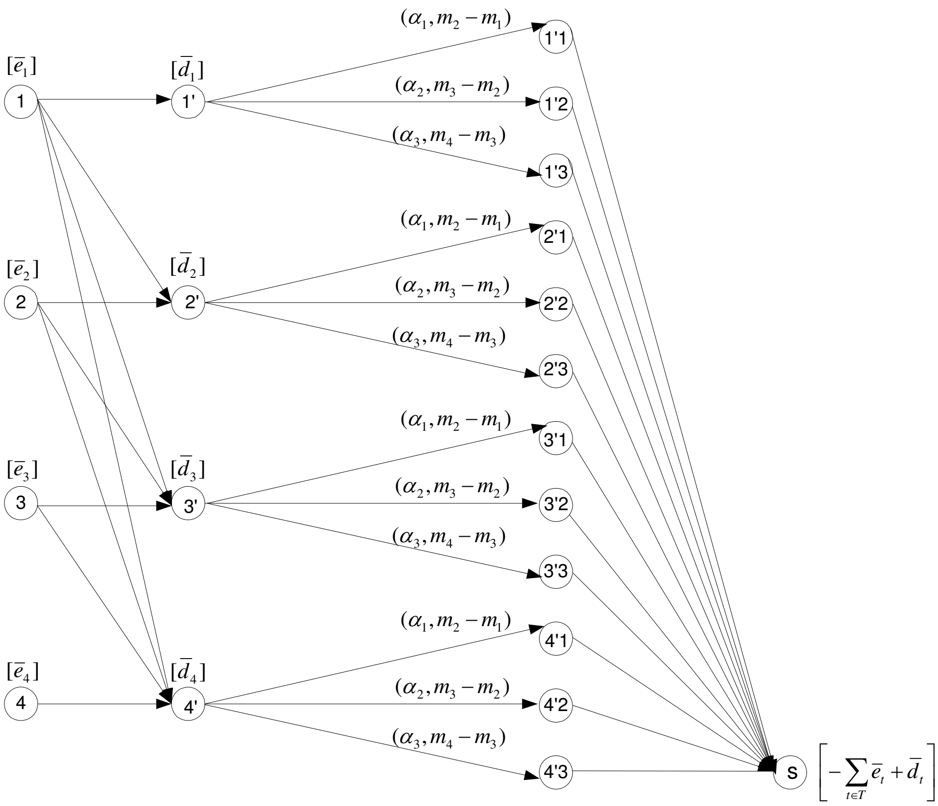


Figure 2.1. The network flow primal subproblem with $|T| = 4$ and $k = 3$.

includes four types of nodes—t nodes (the left nodes in Figure 2.1), t' nodes (the middle-left nodes), $t'i$ nodes (the middle-right nodes), and a sink node (the right node labeled s). For each time period $t \in T$, a t node with supply $\overline{g}^{\xi}_{tn}$ and a t' node with supply $\overline{d}^{\xi}_{tn}$ are in $\mathcal{N}$. An arc between t and t' nodes is in arc set $\mathcal{A}$ whenever $t \leq t'$, and the flow on this arc represents the value of variable $G^{\xi}_{tt'n}$ in the primal subproblem PS^{ξ}_{n}. For each $t \in T$ and each $i = 1, \ldots, k$, a $t'i$ node is added to $\mathcal{N}$, and an arc from the t' node to the $t'i$ node is included in $\mathcal{A}$. The flow on the arc from the t' node to the $t'i$ node is the value of the variable $A^{\xi}_{t'in}$, so it has a per unit cost of $\alpha_{t'ni}$ and an upper bound of $m_{t'n(i+1)} - m_{t'ni}$. A sink node with a demand of $\sum_{t \in T} \overline{d}^{\xi}_{tn} + \overline{g}^{\xi}_{tn}$ is used, and arcs from the $t'i$ nodes to the sink node are in $\mathcal{A}$.

Let π^{ξ}_{tn}, $Y^{\xi}_{\tau n}$, and $\rho^{\xi}_{\tau ni}$ be the dual variables associated with constraint sets (2.31) and (2.32) and the upper bounds in set (2.14), respectively. The *dual subproblem* (DS^{ξ}_{n}) is

$$\max \sum_{t \in T} \left[\sum_{i=1}^{k} (m_{ti} - m_{t(i+1)}) \rho^{\xi}_{tni} \right] + \overline{g}_t \pi^{\xi}_t + \overline{d}_t Y^{\xi}_{tn} \tag{2.33}$$

$$Y^{\xi}_{\tau n} - \rho^{\xi}_{\tau ni} \leq \alpha_{\tau i} \qquad \forall \tau \in T, 1 \leq i \leq k, \tag{2.34}$$

$$\pi^{\xi}_{tn} \leq Y^{\xi}_{\tau n} \qquad \forall t, \tau \in T, t \leq \tau, \tag{2.35}$$

$$\rho^{\xi}_{\tau ni} \geq 0 \qquad \forall \tau \in T, 1 \leq i \leq k, \tag{2.36}$$

$$\pi^{\xi}_{tn}, Y^{\xi}_{\tau n} \text{ free} \qquad \forall t, \tau \in T. \tag{2.37}$$

A dual solution π^{ξ}_{tn} ($Y^{\xi}_{\tau n}$) can be interpreted as the penalty for increasing indirect care (direct care) at time period $t(\tau)$ for each nurse $n \in N(P)$ for each scenario $\xi \in \Xi$. The solution $(\tilde{\pi}^{\xi}_{n}, \tilde{Y}^{\xi}_{n}, \tilde{\rho}^{\xi}_{n}) = 0$ is always feasible, so both the primal and dual subproblems have optimal solutions.

Let DS be the combination of all dual subproblems DS^{ξ}_{n} over all nurses and scenarios. Let Δ be the set of extreme points for the dual subproblem DS. The original SPA problem is reformulated as follows:

$$\min \eta \tag{2.38}$$

$$\eta \geq \sum_{n \in N} \sum_{\xi \in \Xi} \sum_{t \in T} \phi^{\xi} \left[\sum_{p \in P(n)} \left(\tilde{\pi}^{\xi}_{tn} g_{tpn} + \tilde{Y}^{\xi}_{tn} d_{tpn} \right) X_{pn} + \sum_{i=1}^{k} (m_{tni} - m_{tn(i+1)}) \tilde{\rho}^{\xi}_{tni} \right]$$

$$\forall (\tilde{\pi}, \tilde{Y}, \tilde{\rho}) \in \Delta, \tag{2.39}$$

where X_{pn} satisfy (2.9) and (2.12).

The L-shaped method is described as Algorithm 3. Let $\tilde{X}$ be the best assignment found. Let $\tilde{Z}_{UB}$ be the objective value of the best assignment, which is an upper bound on the optimal solution. On each iteration, we consider a subset of dual extreme points $\overline{\Delta} \subseteq \Delta$, and let constraint set (2.39') be the subset of (2.39) over $\overline{\Delta}$. We solve a restricted master problem (2.9), (2.12), (2.38), and (2.39') to find an assignment $\overline{X}$ and an anticipated objective value $\overline{\eta}$. Using the assignment $\overline{X}$, we solve the dual subproblem over all of the nurses and scenarios to obtain $(\tilde{\pi}, \tilde{Y}, \tilde{\rho})$. If the current excess workload for nurses $\sum_{\xi\in\Xi}\sum_{n\in N}\sum_{\tau\in T}\sum_{i=1}^{k}\phi^{\xi}\alpha_{\tau ni}A^{\xi}_{\tau ni}$ is smaller than $\tilde{Z}_{UB}$, then we update the best assignment $\tilde{X}$ and the upper bound $\tilde{Z}_{UB}$. If the anticipated objective value $\overline{\eta}$ is less than the objective value of the dual solution $(\tilde{\pi}, \tilde{Y}, \tilde{\rho})$, then we add a Benders' optimality cut to (2.39'). Otherwise, the algorithm terminates and the assignment $\overline{X}$ is optimal.

2.4.2 Greedy Algorithm

In this section, we present a greedy algorithm to evaluate the recourse function PS^{ξ}_{n}. Properties of solutions by the greedy algorithm are stated, and we prove that the greedy algorithm is a polynomial optimal algorithm. Finally, we describe how to find a complementary optimal dual solution. To simplify notation, we ignore the superscript ξ and the subscript n.

The greedy algorithm solves the subproblems optimally under the following reasonable assumption:

Assumption A5 The nondecreasing piecewise linear convex penalty is the same for each time period; that is, $\alpha_{1i} = \alpha_{2i} = \cdots = \alpha_{|T|i}$ and $m_{1i} = m_{2i} = \cdots = m_{|T|i}$ for all $i = 1, \ldots, k$,

The intuitive explanation for Assumption A5 is that workload is equally penalized throughout a shift.

Consider the greedy algorithm (GAPS) for solving the primal subproblem PS, displayed as Algorithm 2. GAPS uses a solution $(\tilde{A}, \tilde{G})$ that satisfies constraints in (2.13), (2.14), and (2.32), and it increases $\tilde{A}$, $\tilde{G}$, and the objective value as little as possible until constraints in set (2.31) are satisfied. First, GAPS introduces a counter $l(\tau)$ such that a marginal increase in workload for time period τ will increase the objective value by $\alpha_{l(\tau)}$. All direct care is assigned to its given time period, and $\tilde{A}$ is increased appropriately. On every iteration, GAPS considers the time periods in which some indirect care on or prior to these time periods is unassigned. Among these time periods, GAPS examines those with the smallest counter l (equivalently the least marginal penalty α), and it selects the latest such time period τ. Then GAPS finds the latest time period $t \leq \tau$ that has remaining unassigned indirect care. Next $\tilde{A}_{\tau l(\tau)}$ and $\tilde{G}_{t\tau}$ are increased until either $\tilde{A}_{\tau l(\tau)}$ reaches its upper bound (2.14) or all indirect care from time period t is assigned. The counter $l(\tau)$ is incremented if $\tilde{A}_{\tau l(\tau)}$ is increased to its upper bound.

We give some properties of GAPS solutions and prove that the greedy algorithm is optimal in Theorems 8. We also show that the dual solution produced by (2.45) - (2.47) is a complementary optimal dual solution. Let problem $PS(\overline{g}_1, \overline{d}_1, \ldots, \overline{d}_{|T|})$ be a special instance of PS in which $\overline{g}_t = 0$ for all $t = 2, \ldots, |T|$. Now consider two primal problems $PS(\overline{g}_1, \overline{d}_1, \overline{d}^1_2, \ldots, \overline{d}^1_{|T|})$ and $PS(\overline{g}_1, \overline{d}_1, \overline{d}^2_2, \ldots, \overline{d}^2_{|T|})$. Let primal problems $PS(\overline{d}^1_2, \ldots, \overline{d}^1_{|T|})$ and $PS(\overline{d}^2_2, \ldots, \overline{d}^2_{|T|})$ be special instances in which $\overline{g}_1 = \overline{d}_1 = 0$ and $\sum_{t=2}^{|T|} \overline{d}^1_t = \sum_{t=2}^{|T|} \overline{d}^2_t$ and let $z_{PS(\overline{d}^1_2,\ldots,\overline{d}^1_{|T|})}$ and $z_{PS(\overline{d}^2_2,\ldots,\overline{d}^2_{|T|})}$ be their optimal objective values, respectively. Without loss of generality, suppose $z_{PS(\overline{d}^1_2,\ldots,\overline{d}^1_{|T|})} < z_{PS(\overline{d}^2_2,\ldots,\overline{d}^2_{|T|})}$.

Lemma 2. $z_{PS(\overline{g}_1,\overline{d}_1,\overline{d}^1_2,\ldots,\overline{d}^1_{|T|})} \leq z_{PS(\overline{g}_1,\overline{d}_1,\overline{d}^2_2,\ldots,\overline{d}^2_{|T|})}$.

Proof. Let (A^2, G^2) be an optimal solution to $PS(\overline{g}_1, \overline{d}_1, \overline{d}^2_2, \ldots, \overline{d}^2_{|T|})$. Construct the following solution (A^1, G^1). Let the set of time periods $T^1 \subset T$ be such that $\forall t^1 \in T^1$, $\overline{d}^1_{t^1} > \overline{d}^2_{t^1} + G^2_{1t^1}$. For each time period $t \in T \setminus T^1$, increase the value of G^1_{1t} such that $\overline{d}^1_t + G^1_{1t} = \overline{d}^2_t + G^2_{1t}$. Since $\overline{g}_1 + \sum_{t\in T} \overline{d}^1_t = \overline{g}_1 + \sum_{t\in T} \overline{d}^2_t = \sum_{t\in T\setminus T^1} \overline{d}^2_t + G^2_{1t} + \sum_{t\in T^1} \overline{d}^2_t + G^2_{1t} < \sum_{t\in T\setminus T^1} \overline{d}^1_t + G^1_{1t} + \sum_{t\in T^1} \overline{d}^1_t$, then $\sum_{t\in T} G^1_{1t} > \overline{g}_1$. Let t^1 be a time period that has maximum penalty on $\overline{d}^1_t + G^1_{1t}$ and $G^1_{1t} > 0$, let $l^1 = \max\{i = 1, \ldots, k | \overline{d}^1_{t^1} + G^1_{1t^1} > m_i\}$, and reduce $G^1_{1t^1}$ until either $G^1_{1t^1} = 0$, $\overline{d}^1_{t^1} + G^1_{1t^1} = m_{l^1}$, or $\sum_{t\in T} G^1_{1t} = \overline{g}_1$. Repeat the selection of t^1 and reduction of $G^1_{1t^1}$ until $\sum_{t\in T} G^1_{1t} = \overline{g}_1$. Consider the subset of time periods $T^2 \subset T$ for which a time period $t^2 \in T^2$, $\overline{d}^2_{t^2} > \overline{d}^1_{t^2}$. Reducing the most penalized $\overline{d}^2_{t^2}$ in time periods $t^2 \in T^2$ and increasing $\overline{d}^2_{t^1}$ in time periods $t^1 \in T^1$ does not increase the objective penalty because $z_{PS(\overline{d}^1_2,\ldots,\overline{d}^1_{|T|})} < z_{PS(\overline{d}^2_2,\ldots,\overline{d}^2_{|T|})}$. By definition $T^2 \subseteq T \setminus T^1$, so reducing $\overline{d}^1_{t^2} + G^1_{1t^2}$ in the most penalized time periods $t^2 \in T \setminus T^1$ to account for

$\sum_{t^1 \in T^2} \overline{d}^1_{t^1} - \overline{d}^2_{t^1} + G^2_{1t^1}$ will not increase the objective penalty. Thus the objective function value of (A^1, G^1) is less than that of (A^2, G^2), so $z_{PS(\overline{g}_1, \overline{d}_1, \overline{d}^1_2, \ldots, \overline{d}^1_{|T|})} \leq z_{PS(\overline{g}_1, \overline{d}_1, \overline{d}^2_2, \ldots, \overline{d}^2_{|T|})}$. □

Consider the following general greedy algorithm (GGAPS) for PS as given by Algorithm 5.

Lemma 3. *Let $(\tilde{A}, \tilde{G})$ be a solution found by GGAPS. Then $(\tilde{A}, \tilde{G})$ is an optimal solution for PS.*

Proof. By induction and Lemma 2, $(\tilde{A}, \tilde{G})$ is an optimal solution for PS.

□

In GGAPS, we can implement GAPS to solve $PS(\overline{g}_t, \overline{d}_t, \ldots, \overline{d}_{|T|})$ with Assumption A5.

Lemma 4. *Let $(\tilde{A}, \tilde{G})$ be a solution found by GAPS on $PS(\overline{g}_1, \overline{d}_1, \ldots, \overline{d}_{|T|})$. Then $(\tilde{A}, \tilde{G})$ is an optimal solution to $PS(\overline{g}_1, \overline{d}_1, \ldots, \overline{d}_{|T|})$.*

Proof. Suppose to the contrary that $(\tilde{A}, \tilde{G})$ is not an optimal solution. Let $\tilde{l}(\tau), \forall \tau \in T$ be the counters defined in GAPS. Let (A^*, G^*) be an optimal solution to $PS^{\xi}_n(\overline{g}_1, \overline{d}_1, \ldots, \overline{d}_{|T|})$ that minimizes the distance $||G^* - \tilde{G}||$. Let $l^*(\tau)$, $\forall \tau \in T$, be the counters defined in Proposition 1. If $A^*_{\tau i} = m_{i+1} - m_i$ and $A^*_{\tau(i+1)} = 0$, then $l^*(\tau) = i + 1$. Because $(A^*, G^*) \neq (\tilde{A}, \tilde{G})$ and $\sum_{\overline{\tau}=1}^{|T|} G^*_{1\overline{\tau}} = \sum_{\overline{\tau}=1}^{|T|} \tilde{G}_{1\overline{\tau}} = \overline{g}_1$, there exist time periods $\tilde{\tau}, \tau^* \in T$ such that $\overline{d}_{\tilde{\tau}} \leq \sum_{i=1}^{\tilde{l}(\tilde{\tau})} \tilde{A}_{\tilde{\tau} i} = \overline{d}_{\tilde{\tau}} + \tilde{G}_{1\tilde{\tau}} < \sum_{i=1}^{l^*(\tilde{\tau})} A^*_{\tilde{\tau} i} = \overline{d}_{\tilde{\tau}} + G^*_{1\tilde{\tau}}$ and $\overline{d}_{\tau^*} \leq \sum_{i=1}^{l^*(\tau^*)} A^*_{\tau^* i} = \overline{d}_{\tau^*} + G^*_{1\tau^*} < \sum_{i=1}^{\tilde{l}(\tau^*)} \tilde{A}_{\tau^* i} = \overline{d}_{\tau^*} + \tilde{G}_{1\tau^*}$. Now consider the following cases:

Case 1: Suppose $\overline{d}_{\tau^*} + \tilde{G}_{1\tau^*} \leq m_{\tilde{l}(\tilde{\tau})+1}$. Then $\overline{d}_{\tau^*} + G^*_{1\tau^*} < \overline{d}_{\tau^*} + \tilde{G}_{1\tau^*} \leq m_{\tilde{l}(\tilde{\tau})+1}$ and $\overline{d}_{\tilde{\tau}} + G^*_{1\tilde{\tau}} > \overline{d}_{\tilde{\tau}} + \tilde{G}_{1\tilde{\tau}} \geq m_{\tilde{l}(\tilde{\tau})}$. By Assumption A5, increasing $G^*_{1\tau^*}$ and decreasing $G^*_{1\tilde{\tau}}$ does not increase the objective value of (A^*, G^*). Consequently, it is not an optimal solution to $PS(\overline{g}_1, \overline{d}_1, \ldots, \overline{d}_{|T|})$ that minimizes the distance $||G^* - \tilde{G}||$.

Case 2: Suppose $\overline{d}_{\tau^*} + \tilde{G}_{1\tau^*} > m_{\tilde{l}(\tilde{\tau})+1}$. Consider the last iteration of GAPS in which $\tilde{G}_{1\tau^*}$ was increased. By the definition of GAPS, $\tilde{l}(\tau^*) \leq \tilde{l}(\tilde{\tau})$, so $\overline{d}_{\tau^*} + \tilde{G}_{1\tau^*}$ would have increased to at most $m_{\tilde{l}(\tilde{\tau})+1}$, in contradiction to the assumption that $\overline{d}_{\tau^*} + \tilde{G}_{1\tau^*} > m_{\tilde{l}(\tilde{\tau})+1}$.

Thus, $(\tilde{A}, \tilde{G})$ is an optimal solution to $PS(\overline{g}_1, \overline{d}_1, \ldots, \overline{d}_{|T|})$. □

We refer to using GGAPS with GAPS to solve $PS(\overline{g}_1, \overline{d}_1, \ldots, \overline{d}_{|T|})$ as the revised greedy algorithm for the primal subproblem (RGAPS).

Corollary 5. *RGAPS is an optimal algorithm for PS.*

Proof. The proof is immediate from Lemmas 3 and 4. □

Lemma 6. *The complexity of RGAPS is $O(k|T|^2)$ time.*

Proof. We allocate patient care for all $|T|$ time periods. For each time period, the indirect care is assigned to appropriate time periods taking at most $k|T|$ operations. Consequently, the complexity of RGAPS is $O(k|T|^2)$ time. □

Remark. Assumption A5 was need to prove Lemma 4. However, this assumption can be relaxed by using a list of time periods $\hat{T}$ that is sorted by $\alpha_{l(\tau)}$. Algorithm 6 is a GAPS with Sorting algorithm that solves $PS(\overline{g}_1, \overline{d}_1, \ldots, \overline{d}_{|T|})$ without needing Assumption A5. To sort the list $\hat{T}$ uses $O(|T| \log |T|)$ operations, and to insert in the list $\hat{T}$ requires $O(\log |T|)$ computational effort. Consequently, GGAPS using GAPS with Sorting uses $O(k|T|^2 \log |T|)$ operations.

Lemma 7. *Let $(\tilde{A}, \tilde{G})$ be a solution found by GAPS. Let time period $\tau \in T$ be such that there exist time periods $t_1, t_2 \in T$, where $t_1 < t_2 \leq \tau$ and $\tilde{G}_{t_1\tau} > 0$ and $\tilde{G}_{t_2\tau} > 0$. Then GAPS increases $\tilde{G}_{t_2\tau}$ to its final value before it increases $\tilde{G}_{t_1\tau}$.*

Proof. Consider the first iteration in which $\tilde{G}_{t_1\tau}$ was increased. By the definition of GAPS, $\tilde{G}_{t_2\tau}$ would have been selected unless $\sum_{\tilde{\tau}=t_2}^{|T|} \tilde{G}_{t_2\tilde{\tau}} = \overline{g}_{t_2}$. Consequently, $\tilde{G}_{t_2\tau}$ must have been increased its final value before the iteration. □

Theorem 8. *GAPS finds an optimal solution $(\tilde{A}, \tilde{G})$.*

Proof. By Lemma 3, it remains to be proven that RGAPS and GAPS return equivalent solutions. Consider the following induction proof on the number of time periods $|T|$.

(*Base Case*) For $|T| = 1$, RGAPS has one iteration, which uses GAPS, so they are equivalent algorithms. (*Induction Hypothesis*) Suppose RGAPS and GAPS are equivalent algorithms for a problem instance *PS* in which $|T| = \mathbf{T}$. Let $(A^{\mathbf{T}}, G^{\mathbf{T}})$ be the optimal solution given by both algorithms with counters $l^{\mathbf{T}}(t)$, $\forall t \in T$. Consider an instance of *PS* in which $|T| = \mathbf{T} + 1$ and $\overline{d}^{\mathbf{T}+1}_{t+1} = \overline{d}^{\mathbf{T}}_{t}$ and $\overline{g}^{\mathbf{T}+1}_{t+1} = \overline{g}^{\mathbf{T}}_{t}$, $\forall t = 1, \ldots, \mathbf{T}$. Let $(A^{\mathbf{T}+1}, G^{\mathbf{T}+1})$ be the solution given by GAPS. Let $\hat{\tau} \in T$ be such that $G^{\mathbf{T}+1}_{1\hat{\tau}} > 0$, and consider the iteration in which $G^{\mathbf{T}+1}_{1\hat{\tau}}$ was first increased. Prior to the iteration, $G^{\mathbf{T}+1}_{\hat{t}\hat{\tau}}$ had been increased to its final value and $\sum_{\overline{\tau}=\hat{t}}^{|T|} G^{\mathbf{T}+1}_{\hat{t}\overline{\tau}} = \overline{g}^{\mathbf{T}+1}_{\hat{t}}$ for all time periods $\hat{t} = 2, \ldots, \hat{\tau}$ by Lemma 7 and the definition of GAPS. Since $\forall \hat{t} = 2, \ldots, \hat{\tau}$, $\sum_{\overline{\tau}=\hat{t}}^{|T|} G^{\mathbf{T}+1}_{\hat{t}\overline{\tau}} = \overline{g}^{\mathbf{T}+1}_{\hat{t}}$, $G^{\mathbf{T}+1}_{\hat{t}\overline{\tau}}$ must have been its final value, so the value of $G^{\mathbf{T}+1}_{1\hat{\tau}}$ has no effect on the value $G^{\mathbf{T}+1}_{\hat{t}\overline{\tau}}$. The iteration then increases $G^{\mathbf{T}+1}_{1\hat{\tau}}$ and updates $l(\hat{\tau})$ if necessary but makes no changes to $l(\overline{\tau})$ for $\overline{\tau} \neq \hat{\tau}$. Hence the order of the selection of a time period τ in GAPS is not changed for $\overline{\tau} \neq \hat{\tau}$. Thus the value of $G^{\mathbf{T}+1}_{1\hat{\tau}}$ has no effect on the value $G^{\mathbf{T}+1}_{\hat{t}\overline{\tau}}$, $\forall \hat{t} = 2, \ldots, |T|$, and by the induction hypothesis, $G^{\mathbf{T}+1}_{\hat{t}\overline{\tau}}$ must be the same in the solution found using RGAPS. Moreover, prior to the iteration that first increased $G^{\mathbf{T}+1}_{1\hat{\tau}}$, the counter $l(\hat{\tau})$ must be equal to the equivalent counter in RGAPS after the iteration in which $t = \mathbf{T}$. Since the magnitude of an increase in $G^{\mathbf{T}+1}_{1\hat{\tau}}$ uses the same rule in both GAPS and RGAPS, the selection and changes in the counters are the same. Thus GAPS and RGAPS are equivalent algorithms. □

Theorem 9. *The complexity of GAPS is $O(k|T|^2)$ time.*

Proof. By Lemma 6 and the fact that RGAPS and GAPS are equivalent in Theorem 8, the complexity of GAPS is $O(k|T|^2)$ time.

□

We now describe a dual solution $(\tilde{\pi}, \tilde{Y}, \tilde{\rho})$ to DS that is complementary to a solution from GAPS $(\tilde{A}, \tilde{G})$. The complementary slackness conditions of PS and DS are

$$(-A_{\tau i} - m_{\tau i} + m_{\tau(i+1)})\rho_{\tau i} = 0 \qquad \forall i = 1, \dots, k, \forall \tau \in T, \tag{2.40}$$

$$(\alpha_{\tau i} - Y_\tau + \rho_{\tau i})A_{\tau i} = 0 \qquad \forall i = 1, \dots, k, \forall \tau \in T, \tag{2.41}$$

$$(\pi_t - Y_\tau)G_{t\tau} = 0 \qquad \forall t, \tau \in T, t \leq \tau. \tag{2.42}$$

For each time period τ, consider the following two sets of time periods:

$$\mathcal{T}(\tau) = \{\tilde{\tau} \in T | \exists t_1, \dots, t_{q-1}, \tau_1, ..., \tau_q, \tau_1 = \tau, \tau_q = \tilde{\tau}, t_1 \leq \tau_2, t_2 \leq \tau_3, t_{q-1} \leq \tau_q,$$
$$\tilde{G}_{t_1\tau_1}, \tilde{G}_{t_2\tau_2}, ..., \tilde{G}_{t_{q-1}\tau_{q-1}} > 0\} \cup \{\tau\}, \tag{2.43}$$

$$\mathcal{T}^{-1}(\tau) = \{t | G_{t\tilde{\tau}} > 0, \forall \tilde{\tau} \in \mathcal{T}(\tau)\}. \tag{2.44}$$

Let time period $\tilde{\tau} \in \mathcal{T}(\tau)$. Consider the dual solution $(\tilde{\pi}, \tilde{Y}, \tilde{\rho})$ given by

$$\tilde{Y}_\tau = \begin{cases} \min_{\tilde{\tau} \geq \min \mathcal{T}^{-1}(\tau)}\{\alpha_{l(\tilde{\tau})}\} & \text{if } \mathcal{T}^{-1}(\tau) \neq \emptyset \\ \alpha_{l(\tau)} & \text{otherwise} \end{cases} \qquad \forall \tau \in T, \tag{2.45}$$

$$\tilde{\pi}_t = \min_{\tau \geq t} \tilde{Y}_\tau \qquad \forall t \in T, \tag{2.46}$$

$$\tilde{\rho}_{\tau i} = \max\{\tilde{Y}_\tau - \alpha_i, 0\} \qquad \forall i = 1, \dots, k, \forall \tau \in T. \tag{2.47}$$

Theorem 10. *Let $(\tilde{A}, \tilde{G})$ be an optimal solution from GAPS. The dual solution given by (2.45)—(2.47) is a complementary optimal dual solution.*

Lemma 11. *Let $(\tilde{A}, \tilde{G})$ be an optimal solution found by GAPS with objective value z. Let $(\tilde{Y}, \tilde{\pi}, \tilde{\rho})$ be the dual solution given by (2.45)-(2.47). With a sufficiently small $\varepsilon > 0$ increase in $\overline{d}_\tau$ for some $\tau \in T$, there exists a primal feasible solution with an objective function value $z + \varepsilon\tilde{Y}_\tau$.*

Proof. Consider the following two cases:

Case 1: Suppose $\mathcal{T}^{-1}(\tau) = \emptyset$. If $\overline{d}_\tau$ is increased by $\varepsilon \leq m_{l(\tau)+1} - m_{l(\tau)} - \tilde{A}_{\tau l(\tau)}$, then a feasible solution in which $\tilde{A}_{\tau l(\tau)}$ is increased by ε can be constructed. Since the penalty on $\tilde{A}_{\tau l(\tau)}$ is $\alpha_{l(\tau)}$, the increase in the objective value is $\varepsilon\alpha_{l(\tau)} = \varepsilon\tilde{Y}_\tau$.

Case 2: Suppose $\mathcal{T}^{-1}(\tau) \neq \emptyset$. Let $\hat{\tau} \in \arg\min_{\tilde{\tau} \geq \min \mathcal{T}^{-1}(\tau)}\{\alpha_{l(\tilde{\tau})}\}$, and let $\hat{t} = \min \mathcal{T}^{-1}(\tau)$. By the definition of $\mathcal{T}^{-1}(\tau)$, $\exists \tilde{\tau} \in \mathcal{T}(\tau)$ such that $\tilde{G}_{\hat{t}\tilde{\tau}} > 0$. By definition of $\mathcal{T}(\tau)$, $\exists t_1, \ldots, t_{q-1}, \tau_1 = \tau, \ldots, \tau_q = \tilde{\tau}$ such that $t_1 \leq \tau_2, t_2 \leq \tau_3, \ldots, t_{q-1} \leq \tau_q$ and $\tilde{G}_{t_1\tau_1}, \tilde{G}_{t_2\tau_2}, \ldots, \tilde{G}_{t_{q-1}\tau_{q-1}} > 0$. Now suppose

$$\varepsilon \leq \min(\tilde{G}_{t_1\tau_1}, \tilde{G}_{t_2\tau_2}, \ldots, \tilde{G}_{t_{q-1}\tau_{q-1}}, \tilde{G}_{\hat{t}\tilde{\tau}}, m_{l(\hat{\tau})+1} - m_{l(\hat{\tau})} - \tilde{A}_{l(\hat{\tau})}).$$

If $\overline{d}_\tau$ were increased by ε, a feasible solution can be constructed in which both $\tilde{G}_{t_1\tau_1}, \tilde{G}_{t_2\tau_2}, ..., \tilde{G}_{t_{q-1}\tau_{q-1}}$ and $\tilde{G}_{\hat{t}\tilde{\tau}}$ were decreased by ε, and $\tilde{G}_{t_1\tau_2}, \tilde{G}_{t_2\tau_3}, ..., \tilde{G}_{t_{q-1}\tau_q}$, $\tilde{G}_{\hat{t}\hat{\tau}}$, and $\tilde{A}_{l(\hat{\tau})}$ were increased by ε. Since the penalty on $\tilde{A}_{l(\hat{\tau})}$ is $\alpha_{l(\hat{\tau})} = \tilde{Y}_\tau$, the increase in the objective value is $\varepsilon\tilde{Y}_\tau$.

□

Let $(\tilde{A}, \tilde{G})$ be a primal solution found by GAPS, and let $(\tilde{Y}, \tilde{\pi}, \tilde{\rho})$ be the dual solution given by (2.45)-(2.47).

Lemma 12. $(\tilde{Y}, \tilde{\pi}, \tilde{\rho})$ *satisfies the complementary slackness conditions (2.40).*

Proof. Suppose to the contrary, there exists a time period $\tau \in T$ such that $\tilde{A}_{\tau i} < m_{i+1} - m_i$ and $\tilde{\rho}_{\tau i} > 0$. If $\overline{d}_\tau$ is increased by a sufficiently small $\varepsilon > 0$, then a primal feasible solution in which the objective value is increased by $\varepsilon\alpha_i$ can be constructed by Case 1 of Lemma 11. Consequently, $\tilde{Y}_\tau \leq \alpha_i$, in contradiction to the assumption that $\tilde{\rho}_{\tau i} = \tilde{Y}_\tau - \alpha_i > 0$. Hence no such $\tau \in T$ exists. □

Lemma 13. $(\tilde{Y}, \tilde{\pi}, \tilde{\rho})$ *satisfies the complementary slackness conditions (2.41).*

Proof. Suppose to the contrary, there exists a time period $\tau \in T$ such that $\tilde{A}_{\tau i} > 0$ and $\tilde{Y}_\tau - \tilde{\rho}_{\tau i} < \alpha_i$. The index $i \leq l(\tau)$ since $\tilde{A}_{\tau i} = 0, \forall i \geq l(\tau) + 1$. This implies $\tilde{\rho}_{\tau i} = 0$ and $\tilde{Y}_\tau < \alpha_i$ by definition (2.47) and $\tilde{Y}_\tau < \alpha_i \leq \alpha_{l(\tau)}$ by the definition of α. Since $\tilde{Y}_\tau < \alpha_{l(\tau)}$, the set $\mathcal{T}^{-1}(\tau) \neq \emptyset$ by the definition (2.45). For a sufficiently small $\varepsilon > 0$ increase in $\overline{d}_\tau$, a primal feasible solution can be constructed in which the objective value is increased by $\varepsilon\tilde{Y}_\tau$ by case 2 in Lemma 11. Similarly, for a small $\varepsilon' = \min(\varepsilon, \tilde{A}_{\tau i}) > 0$ decrease in $\tilde{A}_{\tau i}$, a primal feasible solution can be constructed in which the objective value is decreased by

$\varepsilon'(\alpha_i - \tilde{Y}_\tau) > 0$. The assumption that $(\tilde{A}, \tilde{G})$ is optimal is contradicted, so no such $\tau \in T$ exists. □

Lemma 14. $(\tilde{Y}, \tilde{\pi}, \tilde{\rho})$ *satisfies the complementary slackness conditions (2.42).*

Proof. Suppose there exist time periods $t \leq \tau$ in which $\tilde{G}_{t\tau} > 0$. By definition (2.46), let $\tilde{\tau} \in \arg\min_{\tilde{\tau} \geq t} \left\{ \tilde{Y}_{\tilde{\tau}} \right\}$, so $\tilde{\pi}_t = \tilde{Y}_{\tilde{\tau}}$ and $\tilde{\tau} \geq t$. By Lemma 11, for a sufficiently small $\varepsilon > 0$ increase in $\overline{d}_{\tilde{\tau}}$, a primal feasible solution in which the objective value is increased by $\varepsilon \tilde{Y}_{\tilde{\tau}}$ can be constructed. Similarly, for a small $\varepsilon' = \min(\varepsilon, \tilde{G}_{t\tau}) > 0$ increase in $\overline{d}_{\tilde{\tau}}$, a primal feasible solution in which the objective value is increased by $\varepsilon \tilde{Y}_{\tilde{\tau}}$ can be constructed by decreasing $\tilde{G}_{t\tau}$, increasing $\tilde{G}_{t\tilde{\tau}}$, and changing the same variables as done for an increase in $\overline{d}_{\tilde{\tau}}$ by ε'. Since case 2 of Lemma 11 includes all such general constructions of primal feasible solutions, the increase in the objective function value $\varepsilon' \tilde{Y}_{\tilde{\tau}}$ is no less than $\varepsilon' \tilde{Y}_\tau$. Hence, $\tilde{Y}_\tau = \tilde{Y}_{\tilde{\tau}} = \tilde{\pi}_t$. □

Theorem 10. *Let* $(\tilde{A}, \tilde{G})$ *be an optimal solution from GAPS. The dual solution given by (2.45) - (2.47) is an complementary optimal dual solution.*

Proof. By definitions of $(\tilde{Y}, \tilde{\pi}, \tilde{\rho})$ in equations (2.45) - (2.47), the dual feasibility constraints (2.34) - (2.37) are satisfied. By Lemmas 12 - 14, $(\tilde{Y}, \tilde{\pi}, \tilde{\rho})$ satisfies the complementary slackness conditions (2.40) - (2.42). □

The greedy algorithm provides a good intuitive explanation corresponding to a nurse's behavior in practice. In constructing the test instances for our computational results, we noticed that direct care was generally less at the end of a shift. In conversations with nurses, we were told that nurses tend to perform indirect care at the end of a shift, which is similar to results from GAPS.

2.4.3 Strengthening the Master Problem

One major drawback of Benders' decomposition is that the first-stage constraints (2.9) and (2.12) in the master problem do not encourage balanced workloads. Consequently, it may spend excessive computational effort generating cuts for extremely imbalanced assignments. In this section, we propose four sets of valid inequalities to tighten the master problem, which are maximum patients constraints, minimum patients constraints, and two sets of symmetry breaking constraints. The results of implementing these valid inequalities to the master problem are demonstrated in Section 3.

2.4.3.1 *Maximum Patients Constraints*

Many states limit the number of patients that can be assigned to a nurse for certain units in a hospital. For instance, California mandates nurse-to-patient ratio regulations that allow no more than six patients assigned to any one nurse for a medical-surgical unit [31]. Typically, the total number of nurses for a shift is obtained from the nurse rescheduling phase. We enforce the following constraint set based upon the number of nurses and the number of patients. To avoid any illegal or unbalanced assignments, we introduce the following *patient-to-nurse ratio constraints* or *maximum patients constraints* (MXPC):

$$\sum_{p \in P(n)} X_{pn} \leq \left\lceil \frac{|P|}{|N|} \right\rceil \qquad \forall n \in N, \tag{2.48}$$

where $\lceil x \rceil$ represents the ceiling of the value x. MXPC prevents assignments with uneven patient loads, which would not be popular with the nurses even if it were balanced in terms of required care. MXPC improves the solvability of the problem because it reduces the feasible region of SPA. Although MXPC can lead to suboptimal solutions, we were unable to construct such a solution in any of our computational experiments.

2.4.3.2 *Minimum Patients Constraints*

Anecdotal evidence suggests that it is common for nurses to receive no fewer than a certain number of patients. We use the following *minimum patient constraints* (MPC):

$$\sum_{p \in P(n)} X_{pn} \geq \left\lfloor \frac{|P|}{|N|} \right\rfloor \qquad \forall n \in N, \tag{2.49}$$

where $\lfloor x \rfloor$ represents the floor of the value x. MPC requires each nurse receive at least a minimum number patients. Incorporating both MXPC and MPC to the master problem enhances algorithmic performance because they eliminate unbalanced and impractical assignments.

2.4.3.3 *Symmetry Breaking Constraints*

A symmetry problem may arise when there are sets of indistinguishable nurses. If we assume that all nurses are identical, given an assignment, there are $|N|! - 1$ equivalent alternative assignments that occur from rearranging the identical nurses with the sets of patients. In other words, symmetry could potentially lead to adding an unnecessary set of Benders' cuts to the master problem. Sherali and Smith [89] investigated symmetry issues in discrete optimization problems, and they developed a method to reduce the number of symmetric solutions. Smith et al. [92] illustrated that incorporating hierarchy constraints to a Synchronous Optical Network Ring Design problem resulted in significant improvement in algorithm efficiency.

We propose two sets of constraints to reduce the symmetry problem. We assume that all nurses are identical for the remainder of this section. Let Θ be an assignment for a shift in which a set of patients P_i is assigned to each nurse $n_i, i = 1, ..., |N|$; that is, $\Theta = ((n_1, P_1), (n_2, P_2), ..., (n_{|N|}, P_{|N|}))$. To eliminate an assignment Θ, we introduce an *assignment symmetry breaking constraints* (ASBC) as follows:

$$\sum_{i=1}^{|N|} \sum_{p \in P_i} X_{pn_i} - \sum_{i=1}^{|N|} \sum_{p \notin P_i} X_{pn_i} \leq |P| - 1. \tag{2.50}$$

Given any assignment, the master problem requires $|N|! - 1$ constraints to eliminate each symmetric assignment individually. Alternatively, we can replace the constraint set (2.50) with the constraint set (2.52). With the same principle as Sherali and Smith [89], we denote an assignment vector $(X_{1n}, X_{2n}, ..., X_{|P|n})$ to be lexicographically greater than or equal to ($\geq^L$) an assignment vector $(X_{1,n+1}, X_{2,n+1}, ..., X_{|P|,n+1})$ for each $n = 1, ..., |N|-1$. The *hierarchy constraints* are given by:

$$\sum_{p \in P} |P|^{p-1} X_{pn} \geq \sum_{p \in P} |P|^{p-1} X_{p,n+1} \qquad \forall n = 1, ..., |N| - 1. \tag{2.51}$$

We also consider *pairwise symmetry breaking constraints* (PSBC), which can be written as:

$$\sum_{p \in P_2} X_{pn_1} - \sum_{p \notin P_2} X_{pn_1} + \sum_{p \in P_1} X_{pn_2} - \sum_{p \notin P_1} X_{pn_2} \leq |P_1| + |P_2| - 1 \qquad \forall n_1, n_2 \in N, n_1 > n_2,$$
$$\forall P_1, P_2 \subseteq P, P_1 >^L P_2. \tag{2.52}$$

PSBC ensures that identical nurses are assigned to sets of patients in lexicographic order. The number of PSBC added to each iteration of the restricted master problem is $\binom{|N|}{2}^2 - 1$. ASBC and PSBC reduce the number of symmetric solutions, and because all nonzero coefficients of constraints are equal to positive or negative one, they encourage integrality of solutions more than the hierarchy constraints in set (2.51). In theory, as the number of nurses scales up, the number of ASBC increases exponentially while PSBC increases polynomially with the number of identical nurses. In practice, we do not often encounter the symmetry problems because nurses do not have identical skills. However, we may not have sufficient data on temporary agency nurses or new hires to consider them unique.

Algorithm 3 Nurse Assignment Benders' Decomposition Algorithm (SPA-BA).

$\overline{\Delta} \leftarrow \emptyset$, the best assignment $\tilde{X} \leftarrow \emptyset$, objective value of the best assignment $\tilde{Z}_{UB} \leftarrow \infty$, $STOP \leftarrow FALSE$.

while $STOP = FALSE$ **do**

Solve the restricted master problem (2.9), (2.12), (2.38), and (2.39') to obtain an assignment $\overline{X}$ and an anticipated objective value $\overline{\eta}$. (On the first iteration, let $\overline{\eta} \leftarrow -\infty$, and let $\overline{X}$ be a feasible assignment.)

for all $n \in N$, $\xi \in \Xi$ **do**

Solve the dual subproblem (DS_n^ξ) to obtain extreme point $(\tilde{\pi}_n^\xi, \tilde{Y}_n^\xi, \tilde{\rho}_n^\xi)$.

end for

if $\sum_{\xi\in\Xi}\sum_{n\in N}\sum_{\tau\in T}\sum_{i=1}^{k}\phi^\xi\alpha_{\tau ni}A_{\tau ni}^\xi < \tilde{Z}_{UB}$ **then**

$\tilde{X} \leftarrow \overline{X}$.

$\tilde{Z}_{UB} \leftarrow \sum_{\xi\in\Xi}\sum_{n\in N}\sum_{\tau\in T}\sum_{i=1}^{k}\phi^\xi\alpha_{\tau ni}A_{\tau ni}^\xi$.

end if

if $\overline{\eta} < \sum_{p\in P}\sum_{n\in N(P)}\sum_{\xi\in\Xi}\sum_{t\in T}\phi^\xi[\left(\tilde{\pi}_{tn}^\xi g_{tpn} + \tilde{Y}_{tn}^\xi d_{tpn}\right)\overline{X}_{pn} + \sum_{i=1}^{k}(m_{tni} - m_{tn(i+1)})\tilde{\rho}_{tni}^\xi]$ **then**

$\overline{\Delta} \leftarrow \overline{\Delta} \cup \left\{(\tilde{\pi}, \tilde{Y}, \tilde{\rho})\right\}$, where $(\tilde{\pi}, \tilde{Y}, \tilde{\rho})$ is the combination of the vectors $(\tilde{\pi}_n^\xi, \tilde{Y}_n^\xi, \tilde{\rho}_n^\xi)$.

else

$STOP \leftarrow TRUE$.

end if

end while

return the best assignment $\tilde{X}$.

Algorithm 4 Greedy Algorithm for the Primal Subproblem (GAPS).

for all $\tau \in T$ **do**

Let the counter $l(\tau)$ be such that $m_{\tau l(\tau)} \leq \overline{d}_\tau < m_{\tau l(\tau)+1}$.

$$\tilde{A}_{\tau i} \leftarrow \begin{cases} m_{\tau(i+1)} - m_{\tau i} & 1 \leq i < l(\tau), \\ \overline{d}_\tau - m_{\tau i} & i = l(\tau), \\ 0 & l(\tau) < i \leq k. \end{cases}$$

$\tilde{G}_{t\tau} \leftarrow 0, \forall t \leq \tau$

end for

while $\sum_{\tilde{\tau}=t}^{|T|} \tilde{G}_{t\tilde{\tau}} < \overline{g}_t, \forall t \in T$ **do**

$\tau \leftarrow \max\left\{\arg\min_{\hat{\tau}\in T}\left\{l(\hat{\tau})\,\middle|\,\exists\hat{t} \leq \hat{\tau}, \sum_{\overline{\tau}=\hat{t}}^{|T|} \tilde{G}_{\hat{t}\overline{\tau}} < \overline{g}_{\hat{t}}\right\}\right\}$.

$t \leftarrow \max\left\{\hat{t} \in T \,\middle|\, \hat{t} \leq \tau, \sum_{\hat{\tau}=\hat{t}}^{|T|} G_{\hat{t}\hat{\tau}} < \overline{g}_{\hat{t}}\right\}$.

$\delta \leftarrow \min\left\{\overline{g}_t - \sum_{\tilde{\tau}=t}^{|T|} \tilde{G}_{t\tilde{\tau}}, m_{\tau(l(\tau)+1)} - \tilde{A}_{\tau l(\tau)}\right\}$.

$\tilde{A}_{\tau l(\tau)} \leftarrow \delta + \tilde{A}_{\tau l(\tau)}$

$\tilde{G}_{t\tau} \leftarrow \delta + \tilde{G}_{t\tau}$

if $\tilde{A}_{\tau l(\tau)} = m_{\tau(l(\tau)+1)}$ **then**

$l(\tau) \leftarrow l(\tau) + 1$.

end if

end while

Algorithm 5 General Greedy Algorithm for the Primal Subproblem (GGAPS)

$t \leftarrow |T|$.

while $t \geq 1$ **do**

Solve $PS(\overline{g}_t, \overline{d}_t, \ldots, \overline{d}_{|T|})$.

$\overline{d}_{\hat{t}} \leftarrow \overline{d}_{\hat{t}} + G_{t\hat{t}}, \forall \hat{t} = t, \ldots, |T|$.

$t \leftarrow t - 1$.

end while

Algorithm 6 GAPS with Sorting

for all $\tau \in T$ **do**

Let the counter $l(\tau)$ be such that $m_{\tau l(\tau)} \leq \overline{d}_\tau < m_{\tau l(\tau)+1}$.

$$\tilde{A}_{\tau i} \leftarrow \begin{cases} m_{\tau(i+1)} - m_{\tau i} & 1 \leq i < l(\tau), \\ \overline{d}_\tau - m_{\tau i} & i = l(\tau), \\ 0 & l(\tau) < i \leq k. \end{cases}$$

$\tilde{G}_{t\tau} \leftarrow 0, \forall t \leq \tau$.

end for

Sort a list of time periods $\hat{T}$ by $\alpha_{l(\tau)}$.

while $\sum_{\tilde{\tau}=t}^{|T|} \tilde{G}_{t\tilde{\tau}} < \overline{g}_t$ **do**

Let $\overline{\tau}$ be a time period in $\hat{T}$ with minimum $\alpha_{l(\tau)}$.

$\delta \leftarrow \min \left\{ \overline{g}_t - \sum_{\tilde{\tau}=t}^{|T|} \tilde{G}_{t\tilde{\tau}}, m_{\overline{\tau}(l(\overline{\tau})+1)} - \tilde{A}_{\overline{\tau}l(\overline{\tau})} \right\}$.

$\tilde{A}_{\overline{\tau}l(\overline{\tau})} \leftarrow \delta + \tilde{A}_{\overline{\tau}l(\overline{\tau})}$

$\tilde{G}_{t\overline{\tau}} \leftarrow \delta + \tilde{G}_{t\overline{\tau}}$

if $m_{\overline{\tau}(l(\overline{\tau})+1)} - \tilde{A}_{\overline{\tau}l(\overline{\tau})}$ **then**

$l(\overline{\tau}) \leftarrow l(\overline{\tau}) + 1$.

end if

Reinsert $\overline{\tau}$ into $\hat{T}$ according to $\alpha_{l(\overline{\tau})}$.

end while

CHAPTER 3

NURSE ASSIGNMENT COMPUTATIONAL RESULTS

In this chapter, we provide a computational study on nurse assignment described in Chapter 2. Problem instances were generated based upon data from a Northeast Texas hospital as described in Section 3.1. These instances, however, cannot be solved exactly within 30 minutes. Consequently, the focus of the computational study is to find good solutions within the time limit. We describe several alternative assignment methods in Section 3.2. In Section 3.3, we compare the solutions from these methods with those from executing the Benders' approach for 30 minutes. We examine the performance of solving the second-stage recourse subproblems by using the greedy algorithm versus the network simplex method. Moreover, we investigate the effects of imposing sets of valid inequalities to strengthen the relaxed master problem. We discuss the number of occurrences of symmetric solutions in the problem instances with identical nurses. Finally, the expected value of perfect information and the value of the stochastic solution of the problem instances are presented in Section 3.4.

3.1 Problem Instances

Each nurse at the Northeast Texas hospital wears a badge that locates the nurse in the hospital unit. The purpose of the locator is so a charge nurse can inform a nurse immediately when one of her patients calls the nurses' station. The locator system stores data on the location of the nurses for one month. In addition to these data, the Northeast Texas hospital provided encrypted patient data for a medical-surgical unit to study for this research from March 2004 - December 2004.

We generated four random instances based upon these data. The first two instances were day shifts from 7:00 AM to 3:00 PM, while instances 3 and 4 were evening and night

Table 3.1. Instances generated from the Northeast Texas hospital data

Instance	Shift	Pat	RN	LVN
1	Day	23	2	1
2	Day	18	4	0
3	Evening	18	2	1
4	Night	13	1	1

shifts from 3:00 PM to 11:00 PM and 11:00 PM to 7:00 AM, respectively. Sundaramoorthi et al. [95] noted that patients' diagnoses and locations are the most significant factors affecting the amount of time nurses spend with patients. For each instance, we sampled a random set of patients from an empirical distribution of patients with similar diagnoses and patient rooms. We used a census matrix from a medical-surgical unit to determine the number and type of nurses for the shift. Table 3.1 displays characteristics of the four instances. The column labeled "Instance" is the random instance, "Shift" is the time of the shift, "Pat" is the number of patients, and "RN" and "LVN" are the number of registered and licensed vocational nurses on duty, respectively.

We partitioned the shift into eight one-hour time periods for T, and we randomly generated 100, 200, 500, 700, 3000, 5000, and 7000 scenarios for Ξ. The probability of each scenario is equally likely. We assumed that the admit and discharge processes were Poisson with mean equal to the number of patients in a shift divided by the average length of stay. Our data indicated that the average length of stay of patients in a medical-surgical unit was 2.725 days per patient. For each time period $\tau \in T$, each patient $p \in P$, and each scenario $\xi \in \Xi$, the direct care $d^{\xi}_{\tau p}$, was sampled from a gamma distribution. Each gamma distribution was fitted by the moment estimator method [108] from the amount of time during time period τ that nurses were in the rooms of patients with diagnoses and rooms similar to those of patient p. We distinguished each nurse by using her badge number. For each time period $\tau \in T$, let $\bar{d}_{\tau\bullet}$ be the mean of the direct care performed by all nurses in time period τ, and for each nurse $n \in N$, let $\bar{d}_{\tau n}$ be the mean of the direct care performed by nurse n in time period τ. For each time period $\tau \in T$, each patient $p \in P$, each nurse $n \in N$, and each scenario $\xi \in \Xi$, the amount of direct care was obtained by $d^{\xi}_{\tau np} = \frac{d^{\xi}_{\tau p} \times \bar{d}_{\tau n}}{\bar{d}_{\tau \bullet}}$. In our computational experiment in Section 3.3.3, we assumed that nurses were identical for the symmetry study. Because indirect care can be performed in several locations, it cannot be estimated from the data from the Northeast Texas hospital. However, in some patient classification systems for similar medical-surgical units, total indirect care is 32% of direct care. Consequently, we estimated indirect care $g^{\xi}_{\tau np} = 0.32 \times d^{\xi}_{\tau np}$, $\forall \tau \in T$, $\forall p \in P, \forall n \in N$, and $\forall \xi \in \Xi$. In addition, we implicitly assumed that direct care engenders indirect care.

3.2 Alternative Assignments

In this section, we describe several alternative approaches to find an assignment. With many scenarios, stochastic integer programming problems are often computationally intractable, but the Mean Value Problem (MVP) often provides a good solution [22]. For each of the four instances from Section 3.1, we replaced the direct and indirect care random variables with their mean, and we solved the deterministic integer programming problem. In all four instances, solving MVP required less than one minute of CPU time, so finding a good solution is computationally tractable.

In addition to MVP, we also used a heuristic that balanced workload based upon the expected total required care of the patients. When the number of nurses divides the number of patients evenly, the heuristic assigns the patients with the greatest and least required care time to the same nurse. Otherwise, the heuristic assigns the patients with greatest required care to the nurses who are assigned to fewer patients. Finally, we randomly divided the patients evenly among the nurses without considering workload.

In practice, charge nurses often intuitively assign patients to nurses. More sophisticated hospitals use patient classification systems that only consider the expected total care and ignore the fluctuations and uncertainty of care. Consequently, assignments in practice are often similar to those of the heuristic or random assignment.

3.3 Computational Results

In this section, we begin with determining a number of scenarios that gives the best SPA results, and we compare the excess workload assignments from the five different assignment methods in Section 3.3.1. In Section 3.3.2, we discuss the efficiency of GAPS versus the network simplex method to solve the second-stage recourse subproblems. In Section 3.3.3, we study the effects of implementing sets of valid inequalities to strengthen the master problem. Finally, we discuss the occurrences of symmetric solutions in our problem.

We solved SPA with and without using the Benders' approach, denoted as SPA-BA and SPA-IP, respectively. If a method required more than 30 minutes to solve, we considered the best solution found within the time limit. MVP, SPA-IP, and SPA-BA were implemented in ANSI C and processed by a Dual 3.06-GHz Intel Xeon Workstation using CPLEX 9.1 callable library. To find an initial solution for SPA-BA, we used the MVP for less than one minute and then used SPA-BA for the remaining time. Solving the MVP for SPA-BA also served as an initial upperbound for the problem.

Before selecting the number of scenarios providing the best SPA results, we conducted an experiment to select an appropriate number of scenarios to evaluate the recourse function. Estimated excess workload was calculated by evaluating assignments with 3000, 5000, and 7000 scenarios. Because the difference between excess workload estimated under 5000 and 7000 scenarios was within one minute of each other, we concluded that the recourse function converged by evaluating it with 5000 scenarios. Consequently, after having obtained solutions from each approach, GAPS calculated the excess workload of each assignment with 5000 scenarios.

We examined the number of scenarios that gave the best SPA results. We obtained assignments by optimizing based upon the four patient instances with 100, 200, 500, and 700 scenarios and evaluated those assignments with 5000 scenarios with GAPS. Table 3.2 compares the average excess workload of optimizing SPA-IP and SPA-BA with different numbers of scenarios. Results indicated that as the number of scenarios increases, SPA-

Table 3.2. The computational results comparing average excess workload from solving SPA-IP and SPA-BA with different numbers of scenarios

Instance	Algorithm	Expected Patient Workload	100 scenarios		200 scenarios	
			Avg Excess Workload	%	Avg Excess Workload	%
1	SPA-IP	1103	89.47	8.11	84.94	7.70
1	SPA-BA	1103	89.36	8.10	88.23	8.00
2	SPA-IP	759	4.38	0.58	5.50	0.72
2	SPA-BA	759	4.48	0.59	4.53	0.60
3	SPA-IP	939	47.35	5.04	46.67	4.97
3	SPA-BA	939	49.67	5.29	46.67	4.97
4	SPA-IP	327	2.12	0.65	2.61	0.80
4	SPA-BA	327	2.38	0.73	1.97	0.60

Instance	Algorithm	Expected Patient Workload	500 scenarios		700 scenarios	
			Avg Excess Workload	%	Avg Excess Workload	%
1	SPA-IP	1103	91.17	8.27	111.35	10.10
1	SPA-BA	1103	86.91	7.88	87.47	7.93
2	SPA-IP	759	4.00	0.53	3.85	0.51
2	SPA-BA	759	4.21	0.55	3.69	0.49
3	SPA-IP	939	46.67	4.97	47.35	5.04
3	SPA-BA	939	48.00	5.11	46.67	4.97
4	SPA-IP	327	1.87	0.57	2.46	0.75
4	SPA-BA	327	1.78	0.55	1.81	0.55

BA was preferable to SPA-IP in most problem instances, and SPA-BA outperformed SPA-IP in all instances with 700 scenarios. The reason SPA-BA performed better with many scenarios is the amount of CPU time that GAPS spent to solve the recourse subproblem is small. In general, more scenarios are necessary for problems with large variance. However, as the number of scenarios increases, more computational time is required to solve problems. The tradeoff between solution quality and the computational effort to solve problems should be considered. To make SPA-BA and SPA-IP comparable, we optimized SPA-BA and SPA-IP with 500 scenarios in the remainder of this computational study.

3.3.1 Comparison to Other Assignments

We evaluated the performance of five different nurse assignment methods–the random assignment method, the heuristic, MVP, SPA-IP, and SPA-BA. Table 3.3 displays the expected total workload in minutes, the average excess workload, and the average excess workload as a percentage of the expected total workload minutes for assignments given five different assignment approaches. All assignments from SPA-IP and SPA-BA were obtained by optimizing the four patient instances with 500 scenarios, and they were evaluated with 5000 scenarios. On each iteration of SPA-BA, we solved the restricted master problem optimally and a single Benders' optimality cut was added.

In all four instances, SPA-IP and SPA-BA competed to find the best solution within the time limit. Assignments from SPA-BA reduced the average excess workload for nurses between 1 minute and 87 minutes over the random assignment, between 1 minute and 85 minutes over the heuristic assignment, and upto 52 minutes over MVP. Considering there are 1095 8-hour shifts per year, SPA-BA could save up to 1588 hours of excess workload each year in each unit of a hospital. Thus, a nurse-assignment decision-support system that used SPA-BA would reduce the burden of the nursing shortage. Note that in the last instance, MVP provided large average excess workload compared to those from other assignments because we did not enforce any strengthening constraints in the problem, which resulted in one nurse assigned to 11 patients while another received only two.

3.3.2 Greedy Algorithm versus Network Simplex

We compared the computational efficiencies of GAPS and the network simplex method to solve the recourse subproblems. Assignments were obtained by optimizing SPA-BA with the four patient instances using both GAPS and the network simplex optimizer in CPLEX 9.1 to solve the linear subproblems. Table 3.4 displays the average CPU time in seconds that GAPS and the network simplex used to update data and solve one subproblem and the total number of Benders' optimality cuts added to the restricted master problem. GAPS is about 30 times faster than the network simplex.

Table 3.3. The computational results comparing solutions from 5 methods on instances 1, 2, 3, and 4

Instance	Algorithm	Expected Total Workload	Average Excess Workload	Percent
1	Random	1103	174.02	15.78
1	Heuristic	1103	153.22	13.89
1	MVP	1103	109.48	9.93
1	SPA-IP	1103	91.17	8.27
1	SPA-BA	1103	86.91	7.88
2	Random	759	13.95	1.84
2	Heuristic	759	15.32	2.02
2	MVP	759	23.91	3.15
2	SPA-IP	759	4.00	0.53
2	SPA-BA	759	4.21	0.55
3	Random	939	114.33	12.17
3	Heuristic	939	133.50	14.21
3	MVP	939	100.71	10.72
3	SPA-IP	939	46.67	4.97
3	SPA-BA	939	48.00	5.11
4	Random	327	3.74	1.14
4	Heuristic	327	3.68	1.13
4	MVP	327	14.75	4.51
4	SPA-IP	327	1.87	0.57
4	SPA-BA	327	1.78	0.55

Table 3.4. The computational comparison between solving SPA-BA with GAPS versus the network simplex

Instance	Greedy Algorithm		Network Simplex	
	CPU time/cut	No. of cuts	CPU time/cut	No. of cuts
1	0.0329	387	0.9180	364
2	0.0351	857	1.1368	860
3	0.0311	580	0.9040	525
4	0.0213	38	0.7342	38

3.3.3 Strengthening Constraints

The computational effects of applying sets of valid inequalities to the master problem are discussed in this section. We assumed that all nurses were identical. Assignments were obtained by solving SPA-BA with and without sets of valid inequalities, which are MXPC, MPC, maximum patients and minimum patients constraints (MX-MPC), ASBC, and PSBC. Table 3.5 shows the average excess workload of assignments from SPA-BA with and without sets of valid inequalities. The column labeled "Iter No." is the total number of iterations or the number of single Benders' optimality cuts added to the relaxed master problem within 30 minutes. The last column labaled "Cut No./Iter" is the number of strengthening cuts added to the relaxed master problem along with a single Benders' cut on each iteration.

In general, implementing MX-MPC tended to be the best option to improve the overall algorithmic performance for these problem instances. Although enforcing MX-MPC did not provide a significant improvement, it enhanced the quality of solutions in most instances. Neither of the symmetry breaking constraints significantly reduced average excess workload.

Moreover, we conducted an analysis on the occurrences of symmetric solutions within 30 minutes. Ignoring sets of valid inequalities, we solved SPA-BA for all identical nurses. Although we expected to obtain many symmetric assignments by reshuffling identical nurses to sets of patients, only five pairs of symmetric solutions occurred in one out of four problem instances. The explanation is that the coefficients of the Benders' cuts are often similar when the total workload in each time period is balanced. In other words, a single Benders' cut often accounts for the symmetric solutions. Since there were few symmetric solutions, neither ASBC nor PSBC significantly improved the solvability of the problem.

3.4 Expected Value of Perfect Information and Value of the Stochastic Solution

Table 3.5. The computational results comparing average excess workload from solving SPA-BA with and without sets of valid inequalities to strengthen the master problem

Instance	Strengthening cut	Iter No.	Total Workload	Excess Workload	%	Cuts No./ Iter
1	None	278	1103	74.07	6.71	0
1	MXPC	277	1103	74.30	6.74	0
1	MPC	277	1103	74.30	6.74	0
1	MX-MPC	266	1103	72.98	6.62	0
1	ASBC	277	1103	73.59	6.67	5
1	PSBC	124	1103	73.57	6.67	9
2	None	305	759	8.42	1.11	0
2	MXPC	282	759	8.20	1.08	0
2	MPC	282	759	8.20	1.08	0
2	MX-MPC	285	759	7.64	1.01	0
2	ASBC	295	759	8.42	1.11	23
2	PSBC	83	759	8.45	1.11	36
3	None	297	939	58.95	6.28	0
3	MXPC	297	939	58.62	6.24	0
3	MPC	297	939	58.62	6.24	0
3	MX-MPC	311	939	59.24	6.31	0
3	ASBC	297	939	58.95	6.28	5
3	PSBC	139	939	59.78	6.37	9
4	None	18	327	1.10	0.34	0
4	MXPC	16	327	1.12	0.34	0
4	MPC	16	327	1.12	0.34	0
4	MX-MPC	16	327	1.12	0.34	0
4	ASBC	18	327	1.10	0.34	1
4	PSBC	57	327	1.04	0.32	1

In this section, we discuss the value of information and the benefit of applying stochastic programming to our problem instances. The *wait-and-see solution* (WS) is the expected value of the optimal solution when we have perfect information. The *expected value of perfect information* (EVPI) is the loss of objective value due to the presence of uncertainty, which can be defined as the difference between the optimal value of SPA and WS [22]. The *value of the stochastic solution* (VSS) describes the loss of ignoring uncertainty in our problem instances when we solve the deterministic programming model, which all random variables are replaced by their mean, instead of the stochastic one. We obtained WS by solving deterministic problems in which each problem corresponded with one scenario from each of the 5000 scenarios. From Table 3.3, the average excess workload from SPA-BA is an upper bound on the optimal value of SPA. Consequently, we can only calculate upper bounds (lower bounds) for EVPI (VSS). Table 3.6 displays WS, EVPI upper bounds, and VSS lower bounds of 4 problem instances. The forth and sixth column represent the EVPI upper bound percentages and VSS lower bound percentages of total objective of SPA-BA, which optimized with 500 scenarios and evaluated with 5000 scenarios, respectively.

The EVPI upper bounds of instances 2 and 4 were relatively low because the sets of nurses scheduled for the shifts had sufficient time and skills to care for the patients, which resulted in small excess workload for nurses in most solutions in Table 3.3. In practice, good nurse budgeting, scheduling, and rescheduling yield low EVPI values. In contrast, instances 1 and 3 had high EVPI upper bounds meaning that perfect information would be helpful to substantially improve the objective function. For example, we reduced excess workload by more than half of an hour with perfect information for instance 1. Instances 2 and 4 had extremely high VSS lower bound percentages. The explanation is that MVP had multiple optimal solutions, and our MVP algorithm arbitrarily selected solutions that did not performed well in these instances. Especially with problem instances that had enormous variance in patient care, the solutions given by the deterministic models would not be able to yield minimal excess workload. Instances 1 and 3 had relatively high VSS lower bound values because these instances contained patients who required a high fluctuation of care. SPA-BA were able to manage variation in patient care better than MVP, which did not consider it. For instance, solutions from SPA-BA provided much less excess workload than MVP, random, and heuristic approaches in Instances 1 and 3 in Table 3.3. Overall, we had considerably large lower bounds for VSS for all instances indicating that using MVP was not as beneficial as SPA.

Table 3.6. An upper bound on the expected value of perfect information and a lower bound on the value of the stochastic solution of 4 problem instances

Instance	WS	EVPI UB	% EVPI UB	VSS LB	% VSS LB
1	52.36	34.55	39.75	22.57	25.97
2	0.25	3.96	94.06	19.70	467.93
3	9.70	38.30	79.79	52.71	109.81
4	0.50	1.28	71.91	12.97	728.65

CHAPTER 4

INTEGRATED NURSE STAFFING AND ASSIGNMENT PROBLEM FORMULATIONS AND SOLUTION ALGORITHMS

Nurse staffing is an important issue of hospitals for several reasons. It is a routine performed by all units in every shift everyday throughout a year. Given that nursing consumes the largest portion of hospital budget, the health care cost is driven by nursing cost. Without an efficient decision planning, hospitals might spend unnecessarily for hiring costly agency nurses or excess permanent staffing. Moreover, the shortage of nurses, which is one of the greatest problem in health care system, is becoming more severe. In addition to the nursing shortage and financial issues, staffing has been a conflict between hospital administration and nurses. Hospital administrations prefer to provide care with minimal cost whereas nurses require minimal excess workload with sufficient staffing. One way to deal with the staffing problem is to determine an efficient staffing decision that optimizes resource allocation to satisfy the demand and to benefit all parties. The goals are to improve the quality of nursing work by balancing nurse workload, to provide an amount of sufficient care to patients, and to reduce the overall cost of hospital. Ultimately, the long-term goal is to reduce the burden of nursing shortage.

The nurse staffing problem involves creating a schedule for nurses determining nursing staffs in charge of working for a shift on a given day. We refer to *nurse scheduling* or *nurse rostering* as a mid-term scheduling, which occurs a couple weeks before a shift. Much research has been done on nurse rostering problems. Recent literature surveys include Cheang et al. [33] and Burke et al. [28]. We refer to *nurse staffing* as a short-term nurse scheduling that occurs 90 minutes before the upcoming shift.

In nurse staffing, we consider scheduled nurses, float nurses, PRN nurses, overtime nurses, and agency nurses in nurse staffing. *Scheduled nurses* are nurses who scheduled to work for a particular unit in a given shift on a given day. *Float nurses* are those trained for a particular floor or specialty. They report to work not knowing which unit will need help. When help is needed, they go to the floor that is short for a shift. A *PRN nurse* is one that works primarily on one unit. They are called as needed to come in and work when there is a shortage in a shift. They also can schedule themselves to work after the schedule is made when they see the holes that need to be filled. *On-call nurses* cannot get further than about a 30 mile radius from the hospital. They must carry a beeper, and be ready to go to the hospital when they are called. If they are called to go in, they will get paid the full amount for their services. Given that on-call nurses usually work for gastrointestinal laboratories, operating rooms, or other special areas, and they are not assigned to the patients, we exclude on-call nurses in our model. *Agency nurses* are also available to call in from agency nurse services. *Overtime nurses* are nurses who worked the previous shift and will work the next shift consecutively.

Based upon the mid-term schedule, a nurse supervisor reevaluates the schedule 90 minutes prior to a shift. If there are more nurses than needed, she lets voluntary surplus scheduled nurses take that day off without pay. When there is a shortage of nurses, she recruits nurses from the following priorities:

1. excess nurses from other units,
2. float nurses,
3. PRN nurses,
4. overtime nurse,
5. agency nurses.

With nurses from the nurse staffing phase, a charge nurse assigns nurses to patients at the beginning of a shift. Staffing has a direct effect on a nurse-patient assignment, nurse workload, and the quality of care for patients [5, 62, 81]. Incorporating staffing decision within a nurse-patient assignment would likely provide better care for patients as well as balance workload for nurses. Hospitals also benefit from having better budget control, providing quality care to patient, and reducing liability cost. Nevertheless, there is little literature on nurse staffing, and no one has ever integrated the staffing problem with assignment.

The focus of this dissertation is on integrating nurse staffing with the nurse-patient assignment decision. Our model focuses on short-term staffing, therefore it is unnecessary to account for nurse preference because it was included in mid-term scheduling. The goal of our model is to determine nurse staffing and their assignment, which minimize excess workload on nurses while meeting uncertain patient care and satisfying a controlled budget. Our model can be viewed as either a general resource allocation model or a general complex personnel scheduling model, therefore it can also be applied to other organizations.

In Section 4.1.1, we present assumptions made for our model. In Section 4.1.2, we describe decision variables and parameters, which are used in the model. The integrated staffing and assignment model is discussed in Section 4.1.3. In Section 4.2, we propose the algorithmic approaches to solve the problem.

4.1 The Nurse Staffing Problem Formulation

4.1.1 Model Assumptions

We made the following reasonable assumptions:

Assumption 1: The number of nurses in each type who are available to provide services to each hospital unit is known.

Assumption 2: The qualifications and specialties of nurses in each type are known. A list of qualified nurses who can provide care to each patient is known prior to when staffing occurs.

Assumption 3: The cost function is linear.

4.1.2 Decision Variables and Parameters

In addition to decision variables and parameters described in Chapter 2, we use the following notation in our model.

Let J be the set of units. Based on a mid-term schedule, let N denote the set scheduled nurses, including full-time nurses and float nurses, assigned to work for a shift. Let R, O, and A be the sets of PRN nurses, overtime nurses, and agency nurses for a shift, respectively. Let $\mathbf{N}$ be the set all nurses including full-time nurses, float nurses, PRN nurses, overtime nurses, and agency nurses for a shift; that is, $\mathbf{N} = N \cup R \cup O \cup A$. For each unit $j \in J$, let $P(j)$ be the set of patients who stay in unit j. For each unit $j \in J$, let $N(j)$ be set of full-time nurses and float nurses who are scheduled to work for a shift in unit j. Let $R(j)$, $O(j)$, and $A(j)$ be the sets of PRN nurses, overtime nurses, and agency nurses qualified to work in unit j, respectively. For each patient $p \in P$, let $N(p)$, $R(p)$, $O(p)$, and $A(p)$ be the sets of scheduled nurses, PRN nurses, overtime nurses, and agency nurses who can be assigned to patient p, respectively. For each patient $p \in P$, let $\mathbf{N}(p)$ be the set of nurses which can be assigned to patient p. For each nurse $n \in \mathbf{N}$, let $P(n)$ be the set of patients that can be assigned to nurse n; that is, $P(n) = \{p \in P | n \in \mathbf{N}(p)\}$.

For each unit $j \in J$, and nurse $n \in N(j)$, let *scheduled nurse variable*

$$Y_{nj} = \begin{cases} 1 & \text{if a scheduled nurse } n \in N(j) \text{ is assigned to work for a shift} \\ & \text{for unit } j \in J, \\ 0 & \text{otherwise.} \end{cases}$$

For each nurse $n \in N$, let *cancellation variable*

$$Y_n^c = \begin{cases} 1 & \text{if a scheduled nurse } n \in N \text{ is canceled for her shift,} \\ 0 & \text{otherwise.} \end{cases}$$

For each unit $j \in J$, and nurse $n \in R(j)$, let *PRN staffing variable*

$$Y_{nj}^r = \begin{cases} 1 & \text{if a PRN nurse } n \in R(j) \text{ is assigned to work for a shift in unit } j \in J, \\ 0 & \text{otherwise.} \end{cases}$$

For each unit $j \in J$, and nurse $n \in O(j)$, let *overtime staffing variable*

$$Y^o_{nj} = \begin{cases} 1 & \text{if an overtime nurse } n \in O(j) \text{ is assigned to work for a shift in unit } j \in J, \\ 0 & \text{otherwise.} \end{cases}$$

For each unit $j \in J$, and nurse $n \in A(j)$, let *agency staffing variable*

$$Y^a_{nj} = \begin{cases} 1 & \text{if an agency nurse } n \in A(j) \text{ is assigned to work for a shift in unit } j \in J, \\ 0 & \text{otherwise.} \end{cases}$$

There are costs associated with hiring nurses. For each unit $j \in J$ and each nurse $n \in N(J)$, let c^s_{nj} be the cost associated with hiring full-time nurse or float nurse n scheduled to work for unit j. For each nurse $n \in N$, let c^c_n be the cost of canceling scheduled nurse n. For each unit $j \in J$ and each nurse $n \in R(j)$, $n \in O(j)$, and $n \in A(j)$, let c^r_{nj}, c^o_{nj}, and c^a_{nj} be the cost of hiring PRN nurses, overtime nurses, and agency nurses n for unit j, respectively. Let B be a budget for hiring nurses for all units on a particular shift. All costs and the budget are given as parameters to the model.

4.1.3 Integrated Nurse Staffing and Assignment Model

In this section, we introduce an extension of the SPA model from Section 2.3 by incorporating the staffing decision into the assignment model. The Stochastic Integrated Nurse Staffing and Assignment Model (SINSA) can be formulated as:

$$\min \sum_{\xi \in \Xi} \sum_{n \in \mathbf{N}} \sum_{\tau \in T} \sum_{i=1}^{k} \phi^{\xi} \alpha_{\tau n i} A^{\xi}_{\tau n i}, \tag{4.1}$$

subject to

$$\sum_{j \in J} \sum_{n \in N(j)} c^s_{nj} Y_{nj} + \sum_{n \in N} c^c_n Y^c_n +$$

$$\sum_{j \in J} \sum_{n \in R(j)} c^r_{nj} Y^r_{nj} + \sum_{j \in J} \sum_{n \in O(j)} c^o_{nj} Y^o_{nj} +$$

$$\sum_{j \in J} \sum_{n \in A(j)} c^a_{nj} Y^a_{nj} \leq B, \tag{4.2}$$

$$\sum_{j \in J(n)} Y_{nj} + Y^c_n = 1 \qquad \forall n \in N, \tag{4.3}$$

$$Y_{nj} \geq X_{pn} \qquad \forall n \in N(p), p \in P(j), j \in J, \tag{4.4}$$

$$Y^r_{nj} \geq X_{pn} \qquad \forall n \in R(p), p \in P(j), j \in J, \tag{4.5}$$

$$Y^o_{nj} \geq X_{pn} \qquad \forall n \in O(p), p \in P(j), j \in J, \tag{4.6}$$

$$Y^a_{nj} \geq X_{pn} \qquad \forall n \in A(p), p \in P(j), j \in J, \tag{4.7}$$

$$Y_{nj} \in \{0,1\} \quad \forall n \in N(j), j \in J, \tag{4.8}$$

$$Y^c_n \in \{0,1\} \quad \forall n \in N, \tag{4.9}$$

$$Y^r_{nj} \in \{0,1\} \quad \forall n \in R(j), j \in J, \tag{4.10}$$

$$Y^o_{nj} \in \{0,1\} \quad \forall n \in O(j), j \in J, \tag{4.11}$$

$$Y^a_{nj} \in \{0,1\} \quad \forall n \in A(j), j \in J, \tag{4.12}$$

$$\sum_{n \in \mathbf{N}(p)} X_{pn} = 1 \qquad \forall p \in P, \tag{4.13}$$

$$\sum_{p\in P(n)} g^{\xi}_{tpn}X_{pn} = \sum_{\tau=t}^{|T|} G^{\xi}_{t\tau n} \qquad \forall t\in T, n\in \mathbf{N}, \xi\in\Xi, \tag{4.14}$$

$$\sum_{p\in P(n)} d^{\xi}_{\tau pn}X_{pn} + \sum_{t=1}^{\tau} G^{\xi}_{t\tau n} = \sum_{i=1}^{k} A^{\xi}_{\tau ni} \qquad \forall \tau\in T, n\in \mathbf{N}, \xi\in\Xi, \tag{4.15}$$

$$X_{pn}\in\{0,1\} \qquad \forall p\in P(n), n\in\mathbf{N}, \tag{4.16}$$

$$G^{\xi}_{t\tau n}\geq 0 \qquad \forall t,\tau\in T, t\leq\tau, n\in\mathbf{N}, \xi\in\Xi, \tag{4.17}$$

$$m_{\tau n(i+1)} - m_{\tau ni} \geq A^{\xi}_{\tau ni}\geq 0 \qquad \forall \tau\in T, 1\leq i\leq k, n\in\mathbf{N}, \xi\in\Xi. \tag{4.18}$$

Objective function (4.1) is to minimize expected excess workload on nurses. Constraint (4.2) is the *budget constraint*, which ensures that the cost of hiring and canceling nurses does not exceed the budget. For each scheduled nurse $n \in N$, the *cancelation constraints* in set (4.3) indicate that either she is assigned to work or her shift is canceled. Constraints (4.4)-(4.7) are linking constraints between staffing and assignment decision variables. If a nurse is assigned to a patient, then she must be scheduled to work for a shift. Constraints (4.8)-(4.12) require the staffing variables be binary. The *nurse assignment constraints* in set (4.13) ensure that every patient is assigned to a nurse. For each nurse $n \in \mathbf{N}$, the *indirect care constraints* in set (4.14) determine the total indirect care performed from the beginning of time period t until the end of the shift. For each time period $\tau \in T$, the *total workload constraints* in set (4.15) define the total workload of nurse $n \in \mathbf{N}$ containing both direct care and indirect care. Constraint set (4.16) is the *binary constraints* for the assignment variables. The nonnegativity constraints in set (4.17) require the indirect care variables be nonnegative. The upper and lower bounds on the marginal workload variables are provided by constraints (4.18). For each $\tau \in T, n \in \mathbf{N}, \xi \in \Xi$, the total workload variable $A^{\xi}_{\tau nk}$ has no upper bound because $m_{\tau n(k+1)} = \infty$. For constraints (4.13)-(4.18) related to nurse assignment, the unit index j can be neglected because the unit is embedded in the patient information.

The *deterministic equivalent model* for integrated nurse staffing and assignment can be written as follows:

$$\min Q(X) \tag{4.19}$$

subject to

$$\sum_{j\in J}\sum_{n\in N(j)} c^s_{nj}Y_{nj} + \sum_{n\in N} c^c_n Y^c_n + \sum_{j\in J}\sum_{n\in R(j)} c^r_{nj}Y^r_{nj} + \sum_{j\in J}\sum_{n\in O(j)} c^o_{nj}Y^o_{nj} + \sum_{j\in J}\sum_{n\in A(j)} c^a_{nj}Y^a_{nj} \le B, \tag{4.20}$$

$$\sum_{j\in J(n)} Y_{nj} + Y^c_n = 1 \qquad \forall n \in N, \tag{4.21}$$

$$Y_{nj} \ge X_{pn} \qquad \forall n \in N(p), p \in P(j), j \in J, \tag{4.22}$$

$$Y^r_{nj} \ge X_{pn} \qquad \forall n \in R(p), p \in P(j), j \in J, \tag{4.23}$$

$$Y^o_{nj} \ge X_{pn} \qquad \forall n \in O(p), p \in P(j), j \in J, \tag{4.24}$$

$$Y^a_{nj} \ge X_{pn} \qquad \forall n \in A(p), p \in P(j), j \in J, \tag{4.25}$$

$$Y_{nj} \in \{0,1\} \qquad \forall n \in N(j), j \in J, \tag{4.26}$$

$$Y^c_n \in \{0,1\} \qquad \forall n \in N, \tag{4.27}$$

$$Y^r_{nj} \in \{0,1\} \qquad \forall n \in R(j), j \in J, \tag{4.28}$$

$$Y^o_{nj} \in \{0,1\} \qquad \forall n \in O(j), j \in J, \tag{4.29}$$

$$Y^a_{nj} \in \{0,1\} \qquad \forall n \in A(j), j \in J, \tag{4.30}$$

$$\sum_{n\in \mathbf{N}(p)} X_{pn} = 1 \qquad \forall p \in P, \tag{4.31}$$

$$X_{pn} \in \{0,1\} \qquad \forall p \in P(n), n \in \mathbf{N}, \tag{4.32}$$

where $Q(X)$ is the *expected second-stage recourse function* defined as:

$$Q(X) = E_{\xi} Q(X, \xi), \tag{4.33}$$

and

$$Q(X, \xi) = \min \sum_{n \in \mathbf{N}} \sum_{\tau \in T} \sum_{i=1}^{k} \alpha_{\tau n i} A^{\xi}_{\tau n i} \tag{4.34}$$

subject to

$$\sum_{p \in P(n)} g^{\xi}_{tpn} X_{pn} = \sum_{\tau=t}^{|T|} G^{\xi}_{t\tau n} \qquad \forall t \in T, n \in \mathbf{N}, \tag{4.35}$$

$$\sum_{p \in P(n)} d^{\xi}_{\tau pn} X_{pn} + \sum_{t=1}^{\tau} G^{\xi}_{t\tau n} = \sum_{i=1}^{k} A^{\xi}_{\tau n i} \qquad \forall \tau \in T, n \in \mathbf{N}, \tag{4.36}$$

$$G^{\xi}_{t\tau n} \geq 0 \qquad \forall t, \tau \in T, t \leq \tau, n \in \mathbf{N}, \tag{4.37}$$

$$m_{\tau n(i+1)} - m_{\tau n i} \geq A^{\xi}_{\tau n i} \geq 0 \qquad \forall \tau \in T, 1 \leq i \leq k, n \in \mathbf{N}. \tag{4.38}$$

4.2 Algorithmic Approaches

In this section, we present decomposition approaches for solving SINSA, which are Benders' decomposition, Lagrangian relaxation with Benders' decomposition, and nested Benders' decomposition. SINSA is viewed as a two-stage stochastic programming problem for the first two approaches. We solve SINSA with Benders' decomposition, which is a common method to solve two-stage stochastic programming problems. In the second approach, we apply the Lagrangian relaxation with Benders' decomposition to solve SINSA, in which we relax a budget constraint (4.2). We describe how the Lagrangian relaxation with Benders' decomposition can be applied as a search method for bicriteria programming problems. Lastly, SINSA is alternatively considered as a multistage stochastic programming problem for which its nested Benders' decomposition is demonstrated.

4.2.1 Benders' Decomposition

In this section, we consider SINSA as a two-stage stochastic integer programming problem. We solve SINSA using the L-shaped method based on Benders' decomposition with integer first-stage variables [17, 22]. The Benders' decomposition separates the original problem given by (4.1)-(4.18) into the master problem and the subproblems. The master problem determines scheduled nurses, PRN nurses, over-time nurses, and agency nurses working for a shift and it assigns those nurses to patients with an objective of minimizing excess workload on nurses. Given the nurse schedule and their assignments, the recourse problems penalize the excess workload from the assignment. The subproblems decompose by the number of nurses and the number of scenarios into $|\mathbf{N}| \times |\xi|$ linear programming subproblems.

Let $(\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj})$ be a given nurse schedule and let $\overline{X}$ be a given assignment. For each $t \in T$, let $\overline{g}^\xi_{tn} = \sum_{p \in P(n)} g^\xi_{tpn} \overline{X}_{pn}$, and let $\overline{d}^\xi_{tn} = \sum_{p \in P(n)} d^\xi_{tpn} \overline{X}_{pn}$. Our recourse subproblems are reduced to ones similar to those primal subproblems and dual subproblems of SPA. For each nurse $n \in \mathbf{N}$ and each scenario $\xi \in \Xi$, the *primal subproblem* is the following linear program (PS^ξ_n):

$$\min \sum_{\tau \in T} \sum_{i=1}^{k} \alpha_{\tau ni} A^\xi_{\tau ni} \tag{4.39}$$

$$\sum_{\tau=t}^{|T|} G^\xi_{t\tau n} = \overline{g}^\xi_{tn} \qquad \forall t \in T, \tag{4.40}$$

$$\sum_{i=1}^{k} A^\xi_{\tau ni} - \sum_{t=1}^{\tau} G^\xi_{t\tau n} = \overline{d}^\xi_{\tau n} \qquad \forall \tau \in T, \tag{4.41}$$

$$(A^\xi_n, G^\xi_n) \text{ satisfy (4.17), and (4.18).}$$

GAPS determines the total workload variables $A^\xi_{\tau ni}$ and total indirect care variables $G^\xi_{t\tau n}$ performed by nurse n from the beginning of time period t until the end of the shift. This problem always has a feasible solution $(\tilde{A}, \tilde{G})$ that is $\tilde{G}^\xi_{ttn} = \overline{g}^\xi_{tn}$ and $\tilde{A}^\xi_{tnk} = \tilde{G}^\xi_{ttn} + \overline{d}^\xi_{tn}$

for all $t \in T$, and all other variables are zero. For each nurse $n \in \mathbf{N}$ and each scenario $\xi \in \Xi$, the dual subproblem (DS_n^ξ) is given by:

$$\max \sum_{t \in T} \left[\sum_{i=1}^{k} (m_{ti} - m_{t(i+1)}) \rho_{tni}^\xi \right] + \overline{g}_t \pi_t^\xi + \overline{d}_t Y_{tn}^\xi \tag{4.42}$$

$$Y_{\tau n}^\xi - \rho_{\tau ni}^\xi \leq \alpha_{\tau i} \qquad \forall \tau \in T, 1 \leq i \leq k, \tag{4.43}$$

$$\pi_{tn}^\xi \leq Y_{\tau n}^\xi \qquad \forall t, \tau \in T, t \leq \tau, \tag{4.44}$$

$$\rho_{\tau ni}^\xi \geq 0 \qquad \forall \tau \in T, 1 \leq i \leq k, \tag{4.45}$$

$$\pi_{tn}^\xi, Y_{\tau n}^\xi \text{ free} \qquad \forall t, \tau \in T. \tag{4.46}$$

The dual subproblem has a feasible solution given by all variables are zero. Consequently, the primal and dual subproblems have optimal solutions. Let DS denote the combination of all dual subproblems DS_n^ξ over all nurses and scenarios. Let Δ denote set of extreme points for the dual subproblem DS. The SINSA reformulation problem (SINSAR) can be written as follows:

$$\min \eta \tag{4.47}$$

subject to

$$\eta \geq \sum_{n \in \mathbf{N}} \sum_{\xi \in \Xi} \sum_{t \in T} \phi^\xi \left[\sum_{p \in P(n)} \left(\tilde{\pi}_{tn}^\xi g_{tpn} + \tilde{Y}_{tn}^\xi d_{tpn} \right) X_{pn} + \sum_{i=1}^{k} (m_{tni} - m_{tn(i+1)}) \tilde{\rho}_{tni}^\xi \right] \tag{4.48}$$

$$\forall (\tilde{\pi}, \tilde{Y}, \tilde{\rho}) \in \Delta,$$

where $(Y_{nj}, Y_n^c, Y_{nj}^r, Y_{nj}^o, Y_{nj}^a)$ satisfy (4.2)-(4.12),

X_{pn} satisfy (4.13) and (4.16).

Constraints in set (4.48) associated with the extreme points of the optimal dual solutions are termed *optimality cuts*.

The L-shaped method is described as Algorithm 7. Let $(\tilde{Y}_{nj}, \tilde{Y}_n^c, \tilde{Y}_{nj}^r, \tilde{Y}_{nj}^o, \tilde{Y}_{nj}^a)$ and $\tilde{X}$ be the best nurse staffing schedule and the best assignment found, respectively. Let $\tilde{Z}_{UB}$ be the objective value of the best staffing and assignment, which is an upper bound on the optimal solution. On each iteration, we consider a subset of dual extreme points $\overline{\Delta} \subseteq \Delta$, and let constraint set (4.48') be the subset of (4.48) over $\overline{\Delta}$. We solve a restricted master problem (4.2)-(4.13), (4.16), (4.47), and (4.48') to find a nurse schedule $(\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj})$, an assignment $\overline{X}$, and an anticipated objective value $\overline{\eta}$. Using the assignment $\overline{X}$, we solve the dual subproblems over all of the nurses and scenarios to obtain $(\tilde{\pi}, \tilde{Y}, \tilde{\rho})$. If the current excess workload for nurses $\sum_{\xi \in \Xi} \sum_{n \in \mathbf{N}} \sum_{\tau \in T} \sum_{i=1}^{k} \phi^{\xi} \alpha_{\tau n i} A^{\xi}_{\tau n i}$ is smaller than $\tilde{Z}_{UB}$, then we update the best nurse schedule $(\tilde{Y}_{nj}, \tilde{Y}^c_n, \tilde{Y}^r_{nj}, \tilde{Y}^o_{nj}, \tilde{Y}^a_{nj})$, the best assignment $\tilde{X}$, and the upper bound $\tilde{Z}_{UB}$. If the anticipated objective value $\overline{\eta}$ is less than the objective value of the dual solution $(\tilde{\pi}, \tilde{Y}, \tilde{\rho})$, then we add a Benders' optimality cut to (4.48'). Otherwise, the algorithm terminates and a nurse schedule $(\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj})$ and the assignment $\overline{X}$ are optimal.

4.2.2 Lagrangian Relaxation with Benders' Decomposition

In this section, we describe the Lagrangian relaxation with Benders' decomposition for solving SINSA, and follow by the subgradient method to determine the value of the Lagrange multiplier. Then, we describe how the Lagrangian relaxation with Benders' decomposition can be viewed as a search method for bicriteria programming problems.

Lagrangian relaxation methods have been widely used to solve integer programming problems that contain sets of hard constraints and easy constraints. By dualizing hard constraints, we construct a Lagrangian problem that is relative easy to solve compared to the original problem. An optimal value of the Lagrangian problem is a lower bound on the optimal value of the original problem. Lagrangian relaxation efficiently solves integer programming problems since it provides better lower bounds than those from linear programming relaxation in a branch and bound algorithm. A review paper for Lagrangian relaxation for solving integer programming problems can be found in Fisher [43]. The *subgradient method* is a common technique to solve the dual of Lagrangian problem, and it usually provides promising results. More information about subgradient method is included in Bertsekas [19] and Nemhauser and Wolsey [72]. Held et al. [50] described

Algorithm 7 Stochastic Integrated Nurse Staffing and Assignment Benders' Decomposition Algorithm (SINSA-BD).

$\overline{\Delta} \leftarrow \emptyset$, the best nurse schedule $(\tilde{Y}_{nj}, \tilde{Y}^c_n, \tilde{Y}^r_{nj}, \tilde{Y}^o_{nj}, \tilde{Y}^a_{nj}) \leftarrow \emptyset$, the best assignment $\tilde{X} \leftarrow \emptyset$, objective value of the best staffing and assignment $\tilde{Z}_{UB} \leftarrow \infty$, $STOP \leftarrow FALSE$.

while $STOP = FALSE$ **do**

Solve the restricted master problem (4.2)-(4.13), (4.16), (4.47), and (4.48') to obtain a nurse schedule $(\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj})$, an assignment $\overline{X}$ and an anticipated objective value $\overline{\eta}$. (On the first iteration, let $\overline{\eta} \leftarrow -\infty$, and let $(\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj})$ be a feasible nurse schedule, $\overline{X}$ be a feasible assignment.)

for all $n \in \mathbf{N}$, $\xi \in \Xi$ **do**

Solve the dual subproblem (DS^ξ_n) to obtain extreme point $(\tilde{\pi}^\xi_n, \tilde{Y}^\xi_n, \tilde{\rho}^\xi_n)$.

end for

if $\sum_{\xi \in \Xi} \sum_{n \in \mathbf{N}} \sum_{\tau \in T} \sum_{i=1}^k \phi^\xi \alpha_{\tau n i} A^\xi_{\tau n i} < \tilde{Z}_{UB}$ **then**

$\tilde{X} \leftarrow \overline{X}$.

$(\tilde{Y}_{nj}, \tilde{Y}^c_n, \tilde{Y}^r_{nj}, \tilde{Y}^o_{nj}, \tilde{Y}^a_{nj}) \leftarrow (\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj})$.

$\tilde{Z}_{UB} \leftarrow \sum_{\xi \in \Xi} \sum_{n \in \mathbf{N}} \sum_{\tau \in T} \sum_{i=1}^k \phi^\xi \alpha_{\tau n i} A^\xi_{\tau n i}$.

end if

if $\overline{\eta} < \sum_{p \in P} \sum_{n \in \mathbf{N}(p)} \sum_{\xi \in \Xi} \sum_{t \in T} \phi^\xi [\left(\tilde{\pi}^\xi_{tn} g_{tpn} + \tilde{Y}^\xi_{tn} d_{tpn}\right) \overline{X}_{pn} +$ $\sum_{i=1}^k (m_{tni} - m_{tn(i+1)}) \tilde{\rho}^\xi_{tni}]$ **then**

$\overline{\Delta} \leftarrow \overline{\Delta} \cup \left\{(\tilde{\pi}, \tilde{Y}, \tilde{\rho})\right\}$, where $(\tilde{\pi}, \tilde{Y}, \tilde{\rho})$ is the combination of the vectors $(\tilde{\pi}^\xi_n, \tilde{Y}^\xi_n, \tilde{\rho}^\xi_n)$.

else

$STOP \leftarrow TRUE$.

end if

end while

return the best nurse schedule $(\tilde{Y}_{nj}, \tilde{Y}^c_n, \tilde{Y}^r_{nj}, \tilde{Y}^o_{nj}, \tilde{Y}^a_{nj})$, and assignment $\tilde{X}$.

theoretical convergence properties and computational performance of the subgradient optimization.

We propose to solve the SINSA by using Benders' decomposition, in which Lagrangian relaxation is employed to relax the budget constraint. We dualize the budget constraint (4.2) to the objective function and obtain the following *Lagrangian problem*:

$$L(\lambda) = \min \sum_{\xi \in \Xi} \sum_{n \in \mathbf{N}} \sum_{\tau \in T} \sum_{i=1}^{k} \phi^{\xi} \alpha_{\tau n i} A^{\xi}_{\tau n i} + \lambda (\sum_{j \in J} \sum_{n \in N(j)} c^{s}_{nj} Y_{nj} + \sum_{n \in N} c^{c}_{n} Y^{c}_{n} + \sum_{j \in J} \sum_{n \in R(j)} c^{r}_{nj} Y^{r}_{nj} + \sum_{j \in J} \sum_{n \in O(j)} c^{o}_{nj} Y^{o}_{nj} + \sum_{j \in J} \sum_{n \in A(j)} c^{a}_{nj} Y^{a}_{nj} - B) \tag{4.49}$$

$$\text{subject to} \qquad (4.3) - (4.18),$$

where λ is a Lagrange multiplier. $L(\lambda)$ is a piecewise linear function. For any $\lambda \geq 0$, $L(\lambda)$ forms a lower bound on SINSA problem, as $\lambda(\sum_{j \in J} \sum_{n \in N(j)} c^{s}_{nj} Y_{nj} + \sum_{n \in N} c^{c}_{n} Y^{c}_{n} + \sum_{j \in J} \sum_{n \in R(j)} c^{r}_{nj} Y^{r}_{nj} + \sum_{j \in J} \sum_{n \in O(j)} c^{o}_{nj} Y^{o}_{nj} + \sum_{j \in J} \sum_{n \in A(j)} c^{a}_{nj} Y^{a}_{nj} - B) < 0$. The Lagrangian problem (4.49),(4.3)-(4.18) can be alternatively viewed as a bicriteria stochastic integer programming problem with objectives that minimize average excess workload on nurses and total nurse staffing cost. Given a nurse schedule $(\overline{Y}_{nj}, \overline{Y}^{c}_{n}, \overline{Y}^{r}_{nj}, \overline{Y}^{o}_{nj}, \overline{Y}^{a}_{nj})$ and an assignment $\overline{X}$, the subproblems is separated by the total number of nurses and the total number of scenarios into $|N| \times |\xi|$ linear programming subproblems. With the same fashion as the Benders' decomposition approach, for each nurse $n \in \mathbf{N}$ and each scenario $\xi \in \Xi$, the *primal subproblem* (PS^{ξ}_{n}) can be written by (4.17), (4.18), and (4.39)-(4.41), and the *dual subproblem* (DS^{ξ}_{n}) is given by (4.42)-(4.46).

We use the subgradient method to determine the Lagrange multiplier λ. The subgradient method for SINSA is described as Algorithm 8. Let r and α denote an iteration number and a step-size, respectively. For each iteration r, let θ_r be a parameter for the subgradient algorithm. On each iteration, we solve the Lagrangian problem by (4.3)-(4.18), (4.49) using the Benders' decomposition described in Algorithm 7 to obtain the nurse schedule $(\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj})$, the assignment $\overline{X}$. According to a given schedule and assignment, we update the step-size α and the Lagrange multiplier λ. If an absolute difference between the previous and current Lagrange multipliers are less than a positively small number ϵ, then we terminate the algorithm and obtain an optimal solution. Otherwise, we update the parameter θ if the objective value does not improve. The iteration number is also updated. We repeat this iterative procedure until the termination criteria is met.

Typically, scheduling problems are difficult because of large solution search spaces. Many search methods have been developed for nurse scheduling problems, for instance, Tabu search, simulated annealing, genetic algorithm, etc. In this dissertation, we provided a novel search approach for a bicriteria stochastic integer program by using Lagrangian relaxation as a framework. A Lagrange multiplier plays a role as a penalty for violating the second objective and Benders' decomposition handles stochasticity in the model. According to our model, we penalize a schedule that violates the budget for a shift. We find the nurse staff and assignment which minimize both excess workload on nurses and budget violation in the Lagrangian problem. During the searching process, the Lagrangian relaxation with Benders' decomposition searches for solutions with different weights between average excess workload on nurses and budget violation.

4.2.3 Nested Benders' Decomposition

The nested Benders' decomposition method is one of the common solution methods for multistage stochastic programming problems [22]. It is appropriate to use when the subproblems have block angular structure and involve further decomposition. In addition to multistage stochastic programs, the decomposition has been used successfully to solve the multistage convex programs [76]. More details about the nested Benders' decomposition can be found at Birge [20], and Birge and Louveaux [22].

We develop a solution approach based on the nested Benders' decomposition of SINSA. SINSA can be considered as a three-stage stochastic programming problem. The first stage problem proposes a nurse schedule that determines nurses who work for the shift. Given a nurse staff, the second stage subproblem assigns nurses to a set of patients. Based upon an assignment, the third stage problems are decomposed into subproblems associated with each nurse at each scenario, and they evaluate assignments.

Given a nurse schedule ($\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj}$) and assignment $\overline{X}$, the subproblems can be reduced to ones similar to those from the two-stage stochastic program with Benders' decomposition. For each nurse $n \in N$ and each scenario $\xi \in \Xi$, the *third stage primal subproblem* ($PN^{3\xi}_n$) is given by (4.17), (4.18), (4.39)-(4.41) and the *third stage dual subproblem* ($DN^{3\xi}_n$) is given by (4.42)-(4.46). Let (DN^3) be the combination of all dual subproblems ($DN^{3\xi}_n$) over all nurses and scenarios. Let Λ be the set of extreme points for the dual subproblem (DN^3). Given a schedule of nurses, the *second stage restricted master problem* (RMP^2) can be formulated as

$$\min \eta^2 \tag{4.50}$$

subject to

$$\eta^2 \geq \sum_{n\in\mathbf{N}} \sum_{\xi\in\Xi} \sum_{t\in T} \phi^\xi \{ \sum_{p\in P(n)} (\tilde{\pi}^\xi_{tn} g_{tpn} + \tilde{Y}^\xi_{tn} d_{tpn}) X_{pn} + \sum_{i=1}^{k} (m_{tni} - m_{tn(i+1)}) \tilde{\rho}^\xi_{tni} \} \quad \forall (\tilde{\pi}, \tilde{Y}, \tilde{\rho}) \in \Lambda, \tag{4.51}$$

$$X_{pn} \leq \overline{Y}_{nj} \quad \forall n \in N(p), p \in P(j), j \in J, \tag{4.52}$$

$$X_{pn} \leq \overline{Y}^r_{nj} \quad \forall n \in R(p), p \in P(j), j \in J, \tag{4.53}$$

$$X_{pn} \leq \overline{Y}^o_{nj} \quad \forall n \in O(p), p \in P(j), j \in J, \tag{4.54}$$

$$X_{pn} \leq \overline{Y}^a_{nj} \quad \forall n \in A(p), p \in P(j), j \in J, \tag{4.55}$$

$$0 \leq X_{pn} \leq 1 \quad \forall p \in P, n \in \mathbf{N}(p), \tag{4.56}$$

where X_{pn} satisfy (4.13).

Constraints (4.51) are optimality cuts, which represent a successive linear approximation of the third stage problem. Note that the binary constraint (4.16) can be relaxed to the upper bound constraint (4.56). Let $(\psi_{(\tilde{\pi},\tilde{Y},\tilde{\rho})}, \sigma_{npj}, \beta_{npj}, \gamma_{npj}, \nu_{npj}, \chi_{pn}, \omega_p)$ be the dual variables associating with constraints (4.51), (4.52), (4.53), (4.54), (4.55), (4.56), and (4.13) respectively. Let $(\psi^r_{(\tilde{\pi},\tilde{Y},\tilde{\rho})}, \sigma^r_{npj}, \beta^r_{npj}, \gamma^r_{npj}, \nu^r_{npj}, \chi^r_{pn}, \omega^r_p)$ be extreme rays of dual polyhedron. The *second stage dual problem* $(DRMP^2)$ can be written as the following:

$$
\begin{aligned}
\max \sum_{(\tilde{\pi},\tilde{Y},\tilde{\rho})\in\Lambda} & (\sum_{n\in\mathbf{N}} \sum_{\xi\in\Xi} \sum_{t\in T} \phi^\xi \sum_{i=1}^{k} (m_{tni} - m_{tn(i+1)})\tilde{\rho}^\xi_{tni}) \psi_{(\tilde{\pi},\tilde{Y},\tilde{\rho})} \\
& - \sum_{j\in J} \sum_{p\in P(j)} \{ \sum_{n\in N(p)} \overline{Y}_{nj}\sigma_{npj} + \sum_{n\in R(p)} \overline{Y}^r_{nj}\beta_{npj} \\
& + \sum_{n\in O(p)} \overline{Y}^o_{nj}\gamma_{npj} + \sum_{n\in A(p)} \overline{Y}^a_{nj}\nu_{npj}\} - \sum_{p\in P} \sum_{n\in\mathbf{N}(p)} \chi_{pn} + \sum_{p\in P} \omega_p
\end{aligned}
\tag{4.57}
$$

subject to

$$-\sum_{(\tilde{\pi},\tilde{Y},\tilde{\rho})\in\Lambda}\sum_{\xi\in\Xi}\sum_{t\in T}\phi^{\xi}(\tilde{\pi}^{\xi}_{tn}g_{tpn}+\tilde{Y}^{\xi}_{tn}d_{tpn})\psi_{(\tilde{\pi},\tilde{Y},\tilde{\rho})}-\sigma_{npj}-\chi_{pn}+\omega_p\leq 0$$

$$\forall j\in J, p\in P(j), n\in N(p), \qquad (4.58)$$

$$-\sum_{(\tilde{\pi},\tilde{Y},\tilde{\rho})\in\Lambda}\sum_{\xi\in\Xi}\sum_{t\in T}\phi^{\xi}(\tilde{\pi}^{\xi}_{tn}g_{tpn}+\tilde{Y}^{\xi}_{tn}d_{tpn})\psi_{(\tilde{\pi},\tilde{Y},\tilde{\rho})}-\beta_{npj}-\chi_{pn}+\omega_p\leq 0$$

$$\forall j\in J, p\in P(j), n\in R(p), \qquad (4.59)$$

$$-\sum_{(\tilde{\pi},\tilde{Y},\tilde{\rho})\in\Lambda}\sum_{\xi\in\Xi}\sum_{t\in T}\phi^{\xi}(\tilde{\pi}^{\xi}_{tn}g_{tpn}+\tilde{Y}^{\xi}_{tn}d_{tpn})\psi_{(\tilde{\pi},\tilde{Y},\tilde{\rho})}-\gamma_{npj}-\chi_{pn}+\omega_p\leq 0$$

$$\forall j\in J, p\in P(j), n\in O(p), \qquad (4.60)$$

$$-\sum_{(\tilde{\pi},\tilde{Y},\tilde{\rho})\in\Lambda}\sum_{\xi\in\Xi}\sum_{t\in T}\phi^{\xi}(\tilde{\pi}^{\xi}_{tn}g_{tpn}+\tilde{Y}^{\xi}_{tn}d_{tpn})\psi_{(\tilde{\pi},\tilde{Y},\tilde{\rho})}-\nu_{npj}-\chi_{pn}+\omega_p\leq 0$$

$$\forall j\in J, p\in P(j), n\in A(p), \qquad (4.61)$$

$$\sum_{(\tilde{\pi},\tilde{Y},\tilde{\rho})\in\Lambda}\psi_{(\tilde{\pi},\tilde{Y},\tilde{\rho})}=1, \qquad (4.62)$$

$$\psi_{(\tilde{\pi},\tilde{Y},\tilde{\rho})}\geq 0 \qquad \forall(\tilde{\pi},\tilde{Y},\tilde{\rho})\in\Lambda, \qquad (4.63)$$

$$\sigma_{npj}\geq 0 \qquad \forall j\in J, p\in P(j), n\in N(p), \qquad (4.64)$$

$$\beta_{npj}\geq 0 \qquad \forall j\in J, p\in P(j), n\in R(p), \qquad (4.65)$$

$$\gamma_{npj}\geq 0 \qquad \forall j\in J, p\in P(j), n\in O(p), \qquad (4.66)$$

$$\nu_{npj}\geq 0 \qquad \forall j\in J, p\in P(j), n\in A(p), \qquad (4.67)$$

$$\chi_{pn}\geq 0 \qquad \forall p\in P, n\in \mathbf{N}(p), \qquad (4.68)$$

$$\omega_p \quad \text{free} \qquad \forall p\in P. \qquad (4.69)$$

Let Ψ (Γ) be the set of extreme points (extreme rays) of the second stage dual problem (4.57)-(4.69). The *restricted master problem* (RMP^1) for the nested Benders' decomposition is reformulated as follows:

$$\min \eta^1 \tag{4.70}$$

subject to

$$\eta^1 \geq -\sum_{j\in J}\sum_{p\in P(j)}\{\sum_{n\in N(p)}\tilde{\sigma}_{npj}Y_{nj}+\sum_{n\in R(p)}\tilde{\beta}_{npj}Y^r_{nj}$$

$$+\sum_{n\in O(p)}\tilde{\gamma}_{npj}Y^o_{nj}+\sum_{n\in A(p)}\tilde{\nu}_{npj}Y^a_{nj}\}+\overline{\psi}-\overline{\chi}+\overline{\omega} \qquad \forall(\tilde{\psi},\tilde{\sigma},\tilde{\beta},\tilde{\gamma},\tilde{\nu},\tilde{\chi},\tilde{\omega})\in\Psi, \tag{4.71}$$

$$-\sum_{j\in J}\sum_{p\in P(j)}\{\sum_{n\in N(p)}\tilde{\sigma}^r_{npj}Y_{nj}+\sum_{n\in R(p)}\tilde{\beta}^r_{npj}Y^r_{nj}$$

$$+\sum_{n\in O(p)}\tilde{\gamma}^r_{npj}Y^o_{nj}+\sum_{n\in A(p)}\tilde{\nu}^r_{npj}Y^a_{nj}\}+\overline{\psi}^r-\overline{\chi}^r+\overline{\omega}^r\leq 0 \quad \forall(\tilde{\psi}^r,\tilde{\sigma}^r,\tilde{\beta}^r,\tilde{\gamma}^r,\tilde{\nu}^r,\tilde{\chi}^r,\tilde{\omega}^r)\in\Gamma, \tag{4.72}$$

where $(Y_{nj}, Y^c_n, Y^r_{nj}, Y^o_{nj}, Y^a_{nj})$ satisfy (4.2), (4.3), (4.8)-(4.12).

Where
$$\overline{\psi}=\sum_{(\tilde{\pi},\tilde{Y},\tilde{\rho})\in\Lambda}(\sum_{n\in\mathbf{N}}\sum_{\xi\in\Xi}\sum_{t\in T}\phi^\xi\sum_{i=1}^{k}(m_{tni}-m_{tn(i+1)})\tilde{\rho}^\xi_{tni})\tilde{\psi}_{(\tilde{\pi},\tilde{Y},\tilde{\rho})},$$

$$\overline{\omega}=\sum_{p\in P}\tilde{\omega}_p,$$

$$\overline{\chi}=\sum_{p\in P}\sum_{n\in\mathbf{N}(p)}\tilde{\chi}_{pn},$$

$$\overline{\psi}^r=\sum_{(\tilde{\pi},\tilde{Y},\tilde{\rho})\in\Lambda}(\sum_{n\in\mathbf{N}}\sum_{\xi\in\Xi}\sum_{t\in T}\phi^\xi\sum_{i=1}^{k}(m_{tni}-m_{tn(i+1)})\tilde{\rho}^\xi_{tni})\tilde{\psi}^r_{(\tilde{\pi},\tilde{Y},\tilde{\rho})},$$

$$\overline{\omega}^r=\sum_{p\in P}\tilde{\omega}^r_p,$$

$$\overline{\chi}^r=\sum_{p\in P}\sum_{n\in\mathbf{N}(p)}\tilde{\chi}^r_{pn}$$

Constraints (4.71) are the optimality cuts passing information from the second stage dual problem ($DRMP^2$) to the restricted master problem (RMP^1). When the second stage problem (RMP^2) is infeasible, the *feasibility cuts* in constraints (4.72) are added to the restricted master problem (RMP^1) to induce a feasible solution.

Figure 4.1 illustrates the flow chart of nested Benders' decomposition method for SINSA. Note that our third stage subproblems ($PN_n^{3\xi}$) for all of the nurses and scenarios are always feasible. The nested Benders' decomposition algorithm for the three-stage integrated nurse staffing and assignment problem (SINSA-NBD) is described as Algorithm 9.

The nested L-shaped method proceeds as follows. Let $SolveRMP1$ denote a boolean variable which is *true* when we need to solve the restricted master problem (RMP^1). Let $CheckRMP1$ and $CheckRMP2$ be boolean variables. If $CheckRMP1$ is *true*, then the current restricted master problem (RMP^1) is checked whether it is optimal with respect to the first stage optimality cut. If $CheckRMP2$ is *true*, then the current second stage restricted master problem (RMP^2) is checked whether it is optimal with respect to the second stage optimality cut. Let $(\tilde{Y}_{nj}, \tilde{Y}_n^c, \tilde{Y}_{nj}^r, \tilde{Y}_{nj}^o, \tilde{Y}_{nj}^a)$ and $\tilde{X}$ be the best nurse staffing schedule and the best assignment found, respectively. Let $\tilde{Z}_{UB}^1$ and $\tilde{Z}_{UB}^2$ be the objective value of the best staffing and best assignment, respectively. On each iteration, we consider a subset of dual extreme points of (DS^3) $\overline{\Lambda} \subseteq \Lambda$, and let constraint set (4.51') be the subset of (4.51) over $\overline{\Lambda}$. We consider a subsets of dual extreme points of ($DRMP^2$) $\overline{\Psi} \subseteq \Psi$, and let constraint set (4.71') be the subset of (4.71) over $\overline{\Psi}$. We also consider a subsets of dual extreme rays of ($DRMP^2$) $\overline{\Gamma} \subseteq \Gamma$, and let constraint set (4.72') be the subset of (4.72) over $\overline{\Gamma}$. We solve a restricted master problem (4.2), (4.3), (4.8)-(4.12), (4.70), (4.71'), and (4.72') to find a nurse schedule $(\overline{Y}_{nj}, \overline{Y}_n^c, \overline{Y}_{nj}^r, \overline{Y}_{nj}^o, \overline{Y}_{nj}^a)$. Give a nurse schedule $(\overline{Y}_{nj}, \overline{Y}_n^c, \overline{Y}_{nj}^r, \overline{Y}_{nj}^o, \overline{Y}_{nj}^a)$, we solve the second stage restricted master problem (4.13), (4.50), (4.51'), (4.52)-(4.56) and obtain a nurse assignment $\overline{X}$ and anticipated penalty $\overline{\eta}^2$. If a current assignment is infeasible, then we add a feasibility cut to (4.72') and update the variable $solveRMP1$ to resolve the restricted master problem (RMP^1) until a feasible assignment is obtained. Otherwise, we update the variables $CheckRMP1$ and $CheckRMP2$ to check whether the restricted master problem (RMP^1) and the second stage restricted master problem (RMP^2) are optimal with respect to optimality cuts. If the variable $CheckRMP2$ is true, we solve the dual subproblems (DS^3) over all of the nurses $n \in \mathbf{N}$ and scenarios $\xi \in \Xi$ to obtain the optimal dual solutions $(\tilde{\pi}, \tilde{Y}, \tilde{\rho})$. If the current excess workload for nurses $\sum_{\xi\in\Xi}\sum_{n\in\mathbf{N}}\sum_{\tau\in T}\sum_{i=1}^{k}\phi^{\xi}\alpha_{\tau ni}A^{\xi}_{\tau ni}$ is smaller than $\tilde{Z}^2_{UB}$, then we update the best assignment $\tilde{X}$, and the upper bound $\tilde{Z}^2_{UB}$. If the constraints (4.51) with these dual solutions $(\tilde{\pi}, \tilde{Y}, \tilde{\rho})$ are violated by the current assignment $(\overline{X})$ and anticipated objective value $\overline{\eta}^2$, then we add a Benders' optimality cut to (4.51') and update the variable $CheckRMP2$ to resolve and check the second stage restricted master problem (RMP^2). Otherwise, we adjust the variable $CheckRMP1$ to check an optimality cut for the restricted master problem (RMP^1). If the $CheckRMP1$ is $true$, then we examine the following condition. If the anticipated objective value $\overline{\eta}^2$ is less than the best upper bound $\tilde{Z}^1_{UB}$, then we update the nurse schedule $(\tilde{Y}_{nj}, \tilde{Y}^c_n, \tilde{Y}^r_{nj}, \tilde{Y}^o_{nj}, \tilde{Y}^a_{nj})$ and the best upper bound $\tilde{Z}^1_{UB}$. If the anticipated objective value $\overline{\eta}^1$ is less than the objective value of the dual solution $(\tilde{\psi}_{(\tilde{\pi},\tilde{Y},\tilde{\rho})}, \tilde{\sigma}_{npj}, \tilde{\beta}_{npj}, \tilde{\gamma}_{npj}, \tilde{\nu}_{npj}, \tilde{\chi}_{pn}, \tilde{\omega}_p)$, then we add a Benders' optimality cut to (4.71') and all boolean variables are adjusted to resolve and check both restricted master problem (RMP^1) and second stage restricted master problem (RMP^2). Then, another iteration is performed. Otherwise, the algorithm terminates and a nurse schedule $(\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj})$ and the assignment $\overline{X}$ are optimal.

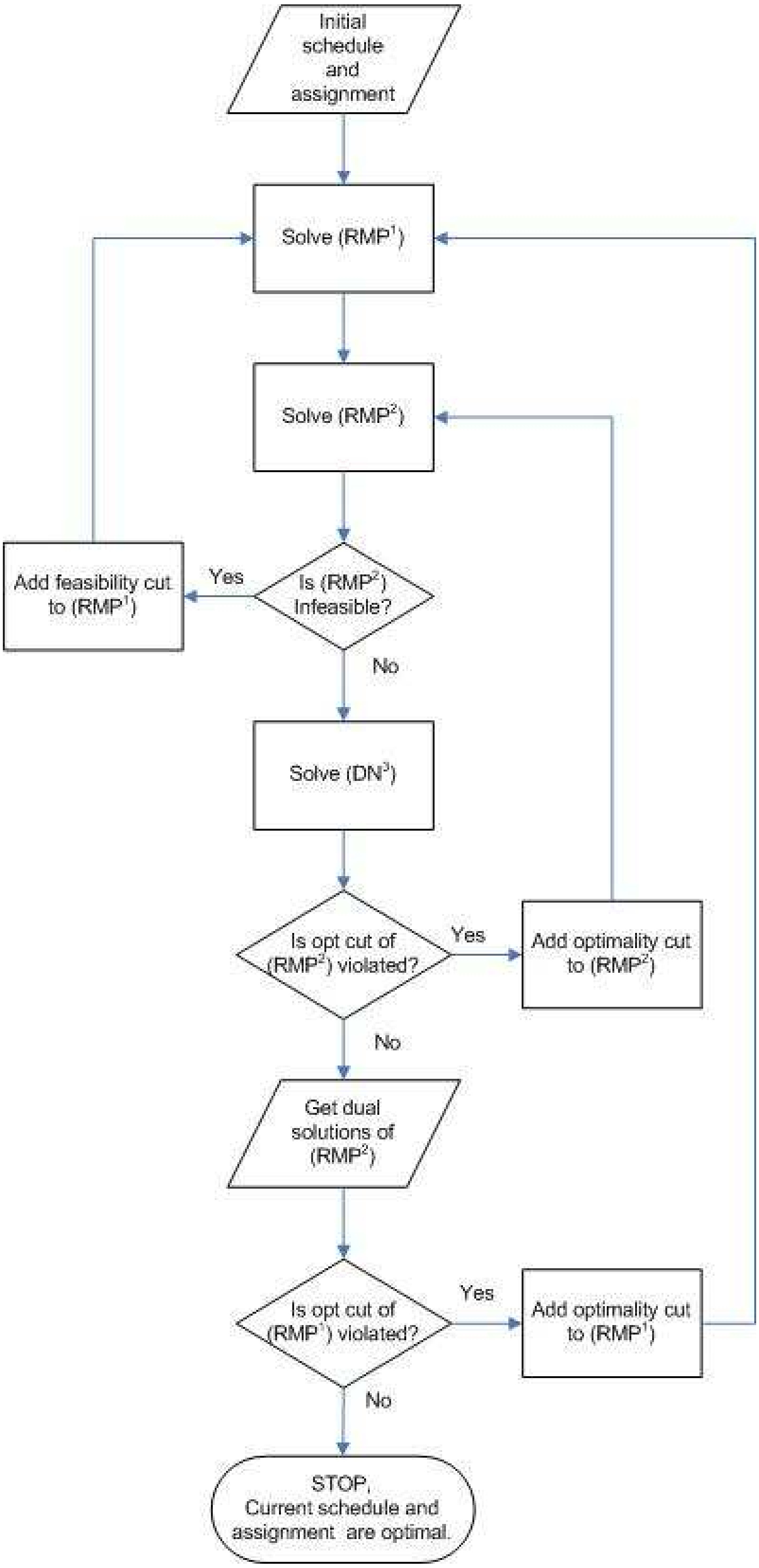


Figure 4.1. The nested Benders' decomposition algorithm for the three-stage integrated nurse scheduling and assignment problem flow chart.

Algorithm 8 Stochastic Integrated Nurse Staffing and Assignment Subgradient Algorithm (SINSA-SA).

Let r be an iteration number, α be step size, $\theta_0 \leftarrow 2$, initial Lagrange multiplier $\lambda^0 \geq 0$, $STOP \leftarrow FALSE$.

while $STOP = FALSE$ **do**

Solve the Lagrangian problem (4.3)-(4.18),(4.49) by the SINSA-BD described in Algorithm 7 to obtain the lower bound $\tilde{Z}_{LB}$.

$\alpha \leftarrow \theta(\tilde{Z}_{UB} - \tilde{Z}_{LB}) / \, (\sum_{j \in J} \sum_{n \in N(j)} c^s_{nj} Y_{nj} + \sum_{n \in N} c^c_n Y^c_n + \sum_{j \in J} \sum_{n \in R(j)} c^r_{nj} Y^r_{nj} + \sum_{j \in J} \sum_{n \in O(j)} c^o_{nj} Y^o_{nj} + \sum_{j \in J} \sum_{n \in A(j)} c^a_{nj} Y^a_{nj} - B)^2$.

$\lambda^{r+1} \leftarrow \max\{0, \lambda^r + \alpha(\sum_{j \in J} \sum_{n \in N(j)} c^s_{nj} Y_{nj} + \sum_{n \in N} c^c_n Y^c_n + \sum_{j \in J} \sum_{n \in R(j)} c^r_{nj} Y^r_{nj} + \sum_{j \in J} \sum_{n \in O(j)} c^o_{nj} Y^o_{nj} + \sum_{j \in J} \sum_{n \in A(j)} c^a_{nj} Y^a_{nj} - B)$.

if $|\lambda^{r+1} - \lambda^r| < \epsilon$ **then**

$STOP \leftarrow TRUE$

end if

if $|\tilde{Z}^{r+1}_{LB} - \tilde{Z}^r_{LB}| < \epsilon$ **then**

$\theta_{r+1} \leftarrow \theta_r / 2$

else

$\theta_{r+1} \leftarrow \theta_r$

end if

$r \leftarrow r + 1$

end while

Algorithm 9 Nested Benders' Decomposition Algorithm for the Three-Stage Integrated Nurse Staffing and Assignment Problem (SINSA-NBD)

Initialization: $\overline{\Theta} \leftarrow \emptyset$, $\overline{\Lambda} \leftarrow \emptyset$, $\overline{\Psi} \leftarrow \emptyset$, $\overline{\Gamma} \leftarrow \emptyset$, the best nurse schedule $(\tilde{Y}_{nj}, \tilde{Y}^c_n, \tilde{Y}^r_{nj}, \tilde{Y}^o_{nj}, \tilde{Y}^a_{nj}) \leftarrow \emptyset$, the best assignment $\tilde{X} \leftarrow \emptyset$, objective value of best scheduling $\tilde{Z}^1_{UB} \leftarrow \infty$, objective value of best assignment $\tilde{Z}^2_{UB} \leftarrow \infty$, *solveRMP1* $\leftarrow$ *TRUE*, *CheckRMP1* $\leftarrow$ *TRUE*, *CheckRMP2* $\leftarrow$ *TRUE*.

while *solveRMP1 = TRUE||CheckRMP1 = TRUE||CheckRMP2 = TRUE* **do**
  if *solveRMP1 = TRUE* **then**
    Solve the restricted master problem (RMP^1) to obtain a nurse schedule $(\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj})$ and an anticipated objective value $\overline{\eta}^1$. (On the first iteration, let $\overline{\eta}^1 \leftarrow -\infty$).
  end if
  Solve the second stage restricted master problem (RMP^2) to obtain a nurse assignment $\overline{X}$ and an anticipated objective value $\overline{\eta}^2$. (On the first iteration, let $\overline{\eta}^2 \leftarrow -\infty$).
  if the second stage restricted master problem (RMP^2) is infeasible **then**
    $\overline{\Gamma} \leftarrow \overline{\Gamma} \cup \left\{(\tilde{\psi}^r, \tilde{\sigma}^r, \tilde{\beta}^r, \tilde{\gamma}^r, \tilde{\nu}^r, \tilde{\chi}^r, \tilde{\omega}^r)\right\}$, where $(\tilde{\psi}^r, \tilde{\sigma}^r, \tilde{\beta}^r, \tilde{\gamma}^r, \tilde{\nu}^r, \tilde{\chi}^r, \tilde{\omega}^r)$ is the combination of the extreme rays $(\tilde{\psi}^r_{(\tilde{\pi},\tilde{Y},\tilde{\rho})}, \tilde{\sigma}^r_{npj}, \tilde{\beta}^r_{npj}, \tilde{\gamma}^r_{npj}, \tilde{\nu}^r_{npj}, \tilde{\chi}^r_{pn}, \tilde{\omega}^r_p)$.
    solveRMP1 $\leftarrow$ *TRUE*, *CheckRMP1* $\leftarrow$ *FALSE*, and *CheckRMP2* $\leftarrow$ *FALSE*.
  else
    solveRMP1 $\leftarrow$ *FALSE*, *CheckRMP1* $\leftarrow$ *TRUE*, and *CheckRMP2* $\leftarrow$ *TRUE*.
  end if
  if *CheckRMP2 = TRUE* **then**
    for all $n \in \mathbf{N}$, $\xi \in \Xi$ **do**
      Solve the third stage dual subproblem $(DN^{3\xi}_n)$ to obtain the extreme points $(\tilde{\pi}^\xi_n, \tilde{Y}^\xi_n, \tilde{\rho}^\xi_n)$.
    end for
    if $\sum_{\xi\in\Xi}\sum_{n\in\mathbf{N}}\sum_{\tau\in T}\sum_{i=1}^{k}\phi^\xi\alpha_{\tau ni}\tilde{A}^\xi_{\tau ni} < \tilde{Z}^2_{UB}$ **then**
      $\tilde{X} \leftarrow \overline{X}$.
      $\tilde{Z}^2_{UB} \leftarrow \sum_{\xi\in\Xi}\sum_{n\in\mathbf{N}}\sum_{\tau\in T}\sum_{i=1}^{k}\phi^\xi\alpha_{\tau ni}\tilde{A}^\xi_{\tau ni}$.
    end if
    if $\overline{\eta}^2 < \sum_{p\in P}\sum_{n\in\mathbf{N}(p)}\sum_{\xi\in\Xi}\sum_{t\in T}\phi^\xi\left[\left(\tilde{\pi}^\xi_{tn}g_{tpn} + \tilde{Y}^\xi_{tn}d_{tpn}\right)\overline{X}_{pn} + \sum_{i=1}^{k}(m_{tni} - m_{tn(i+1)})\tilde{\rho}^\xi_{tni}\right]$ **then**
      $\overline{\Lambda} \leftarrow \overline{\Lambda} \cup \left\{(\tilde{\pi}, \tilde{Y}, \tilde{\rho})\right\}$, where $(\tilde{\pi}, \tilde{Y}, \tilde{\rho})$ is the combination of the vectors $(\tilde{\pi}^\xi_n, \tilde{Y}^\xi_n, \tilde{\rho}^\xi_n)$.
      solveRMP1 $\leftarrow$ *FALSE*, *CheckRMP1* $\leftarrow$ *FALSE*, and *CheckRMP2* $\leftarrow$ *TRUE*.
    else
      solveRMP1 $\leftarrow$ *TRUE*, *CheckRMP1* $\leftarrow$ *TRUE*, and *CheckRMP2* $\leftarrow$ *FALSE*.
    end if
  end if
  if *CheckRMP1 = TRUE* **then**
    if $\overline{\eta}^2 < \tilde{Z}^1_{UB}$ **then**
      $(\tilde{Y}_{nj}, \tilde{Y}^c_n, \tilde{Y}^r_{nj}, \tilde{Y}^o_{nj}, \tilde{Y}^a_{nj}) \leftarrow (\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj})$.
      $\tilde{Z}^1_{UB} \leftarrow \overline{\eta}^2$.
    end if
    if $\overline{\eta}^1 < \sum_{(\tilde{\pi},\tilde{Y},\tilde{\rho})\in\Lambda}(\sum_{n\in\mathbf{N}}\sum_{\xi\in\Xi}\sum_{t\in T}\phi^\xi\sum_{i=1}^{k}(m_{tni} - m_{tn(i+1)})\tilde{\rho}^\xi_{tni})\tilde{\psi}_{(\tilde{\pi},\tilde{Y},\tilde{\rho})} - \sum_{j\in J}\sum_{p\in P(j)}\left\{\sum_{n\in N(p)}\overline{Y}_{nj}\tilde{\sigma}_{npj} + \sum_{n\in R(p)}\overline{Y}^r_{nj}\tilde{\beta}_{npj} + \sum_{n\in O(p)}\overline{Y}^o_{nj}\tilde{\gamma}_{npj} + \sum_{n\in A(p)}\overline{Y}^a_{nj}\tilde{\nu}_{npj}\right\} + \sum_{p\in P}\tilde{\omega}_p - \sum_{p\in P}\sum_{n\in\mathbf{N}(p)}\tilde{\chi}_{pn}$ **then**
      $\overline{\Psi} \leftarrow \overline{\Psi} \cup \left\{(\tilde{\psi}, \tilde{\sigma}, \tilde{\beta}, \tilde{\gamma}, \tilde{\nu}, \tilde{\chi}, \tilde{\omega})\right\}$, where $(\tilde{\psi}, \tilde{\sigma}, \tilde{\beta}, \tilde{\gamma}, \tilde{\nu}, \tilde{\chi}, \tilde{\omega})$ is the combination of the vectors $(\tilde{\psi}_{(\tilde{\pi},\tilde{Y},\tilde{\rho})}, \tilde{\sigma}_{npj}, \tilde{\beta}_{npj}, \tilde{\gamma}_{npj}, \tilde{\nu}_{npj}, \tilde{\chi}_{pn}, \tilde{\omega}_p)$.
      solveRMP1 $\leftarrow$ *TRUE*, *CheckRMP1* $\leftarrow$ *TRUE*, and *CheckRMP2* $\leftarrow$ *TRUE*.
    else
      solveRMP1 $\leftarrow$ *FALSE*, *CheckRMP1* $\leftarrow$ *FALSE*, and *CheckRMP2* $\leftarrow$ *FALSE*.
    end if
  end if
end while
return the best nurse schedule $(\tilde{Y}_{nj}, \tilde{Y}^c_n, \tilde{Y}^r_{nj}, \tilde{Y}^o_{nj}, \tilde{Y}^a_{nj})$, and assignment $\tilde{X}$.

CHAPTER 5

INTEGRATED NURSE STAFFING AND ASSIGNMENT COMPUTATIONAL RESULTS

We report a computational study on integrated nurse staffing and assignment in this section. We tested three solution approaches on four problem instances generated from data from a Northeast Texas hospital. The problem instances are described in Section 5.1. Because of the complexity of the model, these problem instances cannot be solved optimally within 30 minutes. However, finding the optimal solutions may be meaningless since the nurse supervisor wants to quickly obtain high quality schedule and assignment that satisfy all requirements. Accordingly, the focus of the computational study is to find good solutions within a 30-minute time limit. In Section 5.2, we select the appropriate parameters for the solution approaches, and then the algorithmic efficiencies of three approaches are compared. In Section 5.2.3, we study the effects of imposing nurses to work on their primary work units versus allowing nurses to float to other units. Results are stated in the same section.

5.1 Problem Instances

The Northeast Texas hospital provided us encrypted data from two medical-surgical units, namely, Med-Surg1 and Med-Surg2, for this study, and the data was from March 2004-December 2004. We obtained encrypted patient data including patient's primary diagnosis, room location, admission date, discharge date, units a patient stayed. In addition to patient data, the Northeast Texas hospital gave nurses data as well. Each nurse at the hospital wears a badge that locates nurses in the hospital unit, so that a charge nurse can reach a nurse immediately when her patient calls the nurses' station. Given that the Northeast Texas hospital has a nurse locator device with RFID technology,

Table 5.1. Instances generated from the Northeast Texas hospital data

		Med-Surg1 unit		Med-Surg2 unit	
Instance	Shift	No. of patients	No. of nurses	No. of patients	No. of nurses
1	Day	23	2-1	23	2-1
2	Day	18	4-0	18	4-0
3	Evening	18	2-1	18	2-1
4	Night	13	1-1	13	1-1

they can track nurses' location from her badge and collect location data for nurses over months.

Problem instances were generated based upon encrypted patient data and nurse data. We used the Med-Surg1 unit instances described in Section 3.1. The Med-Surg2 unit instances were generated in the same fashion as those from Med-Surg1. We randomly generated 500 and 5000 scenarios for Ξ. The probability of each scenario is equally likely. We estimated patient admission and discharge to a unit as Poisson processes with mean equal to the number of patients in a shift divided by the average length of stay. Our data indicates that the average length of stay of patient in Med-Surg1 and Med-Surg2 unit were 2.725 and 1.936 days per patient, respectively. Besides patient information, individual nurse skills were also taken into consideration. Table 5.1 represents characteristics of the four problem instances from two medical-surgical units. The column labeled "Instance" is the random instance, "Shift" is the time of the shift, and "No. of Pat" is the number of patients in the instance. The column labeled "No. of Nurses" is in the format of a-b, where a and b represent the number of registered nurses and licensed vocational nurses on duty, respectively. Table 5.2 displays salary for each type of nurses. The column labeled "Nurse Type" represents types of nurses. The column labeled "\$/shift" displays the salary of a nurse per shift, and "Total No. of Nurses" shows the total number of nurses. The row labeled "Regular Nurse" is in the format of c-d-e-f, where c, d, e, and f represent the number of regular nurses in Instances 1-4, respectively. The last row labeled "Budget" is the total budget for both units for a shift.

Table 5.2. Salary of different types of nurses

Nurse Type	\$/Shift	Total no. of nurses
Regular Nurse	160	6-8-6-4
Overtime Nurse	240	4
PRN Nurse	256	4
Agency Nurse	320	4
Budget	\$2000	

5.2 Computational Results

In this section, we present computational results based upon instances created from real data from the Northeast Texas hospital described in Section 5.1. We begin with determining appropriate parameters for the Lagrangian relaxation with Benders' decomposition and the nested Benders' decomposition methods in Section 5.2.1. In Section 5.2.2, we compare expected excess workload and staffing cost from three different solution methods. The tradeoff between average excess workload and staffing cost are shown. Finally, we perform a computational comparison between two policies, which are with and without a reasonable assumption that nurses should be restricted to work on their primary work units in Section 5.2.3.

We implemented three different solution approaches in the C programming language on a Dell Precision Workstation with dual 3.06-Gz Intel Xeon processors using CPLEX 9.1 callable library. We solved SINSA with Benders' decomposition, Lagrangian relaxation with Benders' decomposition, and nested Benders' decomposition approaches, denoted as SINSA-BD, SINSA-LRBD, SINSA-NBD, respectively. We solved the mean value problem, which is a deterministic integer programming replaced direct care and indirect care random variables with their mean, for less than one minutes to find an initial solution for all methods. Then, the problem was solved by each solution method for the remaining time. We optimized SINSA-BD, SINSA-LRBD, and SINSA-NBD with 500 scenarios and evaluated the recourse subproblems by using the greedy algorithm (GAPS) with 5000 scenarios to obtain the excess workload of each schedule and assignment.

According to the staffing policy from the Northeast Texas hospital, the staffing happens per unit with different managers. Med-Surg1 and Med-Surg2 nurses are assigned to their primary work units. Nevertheless, two medical-surgical units use the same nurse pool of PRN nurses, overtime nurses, and agency nurses. Therefore, we apply the following assumption:

Assumption 4: Scheduled nurses must be assigned to their primary work units. PRN nurse, overtime nurse, and agency nurses can be scheduled to any units needing help.

We refer to *nondominated solutions* as nurse schedules and assignments that are not dominated by any other schedules and assignments found, either they require less excess workload or less staffing cost than the other solutions found. Algorithm 10 describes the algorithm for the nondominated solutions. Let L be a list of nondominated solutions. Every iteration, we solve SINSA by one of three proposed approaches to obtain a solution, we compare the current solution with those in the list L. If the current solution produces higher staffing cost or more excess workload than those in the list L, we discard the current solution. Otherwise, we add the current solution into the list L. Then, we update the list of nondominated solutions by deleting solutions dominated by the current solution. Each solution represents schedules and assignments for nurses in an upcoming shift. We refer to the *efficient frontier* as a tradeoff curve between excess workload and staffing cost of the set of nondominated solutions found within 30 minutes. The focus of this dissertation is to find many nondominated solutions to form the efficient frontier.

5.2.1 Parameter Tuning

In this section, we determine appropriate parameters for solving SINSA-LRBD and SINSA-NBD.

5.2.1.1 Parameters for SINSA-LRBD

We find proper parameters for the SINSA-LRBD approach in this section. We solve SINSA-LRBD by employing the subgradient algorithm for stochastic integrated

Algorithm 10 Nondominated Solution Algorithm

Let L be a list of nondominated solutions.

Solve the SINSA to obtain a nurse schedule $(\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj})$, an assignment $\overline{X}$, and their objective function $Z_{(\overline{Y}_{nj},\overline{Y}^c_n,\overline{Y}^r_{nj},\overline{Y}^o_{nj},\overline{Y}^a_{nj},\overline{X})}$.

if $\exists(Y_{nj}, Y^c_n, Y^r_{nj}, Y^o_{nj}, Y^a_{nj}, X) \in L$ in which $Z_{(Y_{nj},Y^c_n,Y^r_{nj},Y^o_{nj},Y^a_{nj},X)} < Z_{(\overline{Y}_{nj},\overline{Y}^c_n,\overline{Y}^r_{nj},\overline{Y}^o_{nj},\overline{Y}^a_{nj},\overline{X})}$ **then**

Delete the current solution $(\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj}, \overline{X})$.

else

Add the current solution $(\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj}, \overline{X})$ to a list L, $L \leftarrow L \cup (\overline{Y}_{nj}, \overline{Y}^c_n, \overline{Y}^r_{nj}, \overline{Y}^o_{nj}, \overline{Y}^a_{nj}, \overline{X})$.

for all $(Y_{nj}, Y^c_n, Y^r_{nj}, Y^o_{nj}, Y^a_{nj}, X) \in L$ **do**

if $Z_{(\overline{Y}_{nj},\overline{Y}^c_n,\overline{Y}^r_{nj},\overline{Y}^o_{nj},\overline{Y}^a_{nj},\overline{X})} < Z_{(Y_{nj},Y^c_n,Y^r_{nj},Y^o_{nj},Y^a_{nj},X)}$ **then**

Delete a solution $(Y_{nj}, Y^c_n, Y^r_{nj}, Y^o_{nj}, Y^a_{nj}, X)$ in a list L, $L \leftarrow L \backslash (Y_{nj}, Y^c_n, Y^r_{nj}, Y^o_{nj}, Y^a_{nj}, X)$.

end if

end for

end if

nurse staffing and assignment (SINSA-SA) in Algorithm 8. Table 5.3 depicts parameters for SINSA-SA, which are initial step-size α, initial Lagrange multiplier λ, parameter for algorithm θ, small positive number ϵ, and number of iteration limit. The values of initial parameters are selected because they are common for the subgradient algorithm.

The termination criteria for SINSA-SA algorithm are the following:

1. The Lagrange multiplier converges within the small positive number ϵ. The different between the previous and the current Lagrange multiplier is less than the small positive number ϵ.
2. The time limit is met. We use 30 minutes time limit in our computational results.

Table 5.3. Parameters for the SINSA-LRBD approach

Parameter	Value
Initial step size α	2.0
Initial Lagrange multiplier λ	0.0
Initial LRBD parameter θ	2
ϵ	0.00005
Iteration number	2000

3. The iteration number limit is reached. The limitation for solving SINSA-SA is 2000 iterations.

SINSA-SA terminates when one of the above termination criteria is satisfied.

According to SINSA-SA, we solve SINSA-BD described in Algorithm 7 and update the Lagrange multiplier and the step-size. Then, we perform another iteration until the termination criteria is met. One problem with SINSA-SA is that we cannot optimally solve SINSA-BD within 30 minutes, therefore not enough solutions are generated to form the efficient frontier. One way to overcome this problem is to set time limit for solving SINSA-BD. Accordingly, we examined time duration for solving SINSA-BD embedded in SINSA-SA that yields the best results. We solved SINSA-BD for 30, 60, 120, and 300 seconds, and then updated the step-size, Lagrange multiplier, and parameter θ. Tables 5.4-5.7 show excess workload and staffing cost of solving SINSA-BD with different time limits embedded in SINSA-SA for all four instances, respectively. Figure 5.1 displays the efficient frontiers of solving SINSA-BD with different time limits embedded in SINSA-SA, and they indicated that solving SINSA-BD with 300 seconds within SINSA-SA provided the best results. Thus, we solved SINSA-BD for 300 seconds before updating the sub-gradient parameters in SINSA-SA in the remainder of this computational study.

5.2.1.2 SINSA-NBD Algorithm Enhancement

SINSA-NBD confronted the similar situation as SINSA-LRBD that not many solutions were produced when we tried to solve the restricted master problem (RMP^1)

Table 5.4. Instance 1: solving SINSA-BD with different time limits within the SINSA-LRBD approach

Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
30 sec.			60 sec.		
58.90	3152	14	49.17	3168	14
102.11	3088	14	67.66	2032	9
185.30	2768	13	93.05	1440	7
309.24	928	4	104.78	1392	7
311.24	800	4	117.56	1360	6
545.67	720	4	126.23	1200	6
659.46	656	3	195.67	1120	6
688.15	560	3	221.31	1072	5
890.41	480	3	308.52	912	4
1094.86	400	2	336.74	816	4
1273.56	320	2	375.79	720	4
			761.68	560	3
			1127.59	400	2
			1273.56	320	2
120 sec.			300 sec.		
37.05	3904	17	32.83	3968	17
38.94	3728	16	35.48	3728	16
39.99	3648	16	42.12	3648	16
42.41	3232	14	45.09	3328	15
75.96	3088	14	54.60	2912	13
117.67	1200	6	125.21	2768	13
133.87	1120	6	168.12	2576	12
218.26	1056	6	172.93	1152	6
255.22	976	5	311.06	976	5
326.56	896	4	337.68	880	4
359.56	720	4	366.12	736	4
1048.90	560	3	372.15	720	4
1094.86	400	2	701.54	640	4
1273.56	320	2	1065.57	560	3
			1127.59	400	2
			1273.56	320	2

Table 5.5. Instance 2: solving SINSA-BD with different time limits within the SINSA-LRBD approach

Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
	30 sec.			60 sec.	
234.34	2400	12	18.40	480	3
607.06	2240	11	105.81	320	2
730.01	1520	9			
	120 sec.			300 sec.	
34.12	3536	16	16.51	1040	5
44.77	3216	15	24.35	720	4
50.16	2976	14	26.92	576	3
59.06	2656	13	42.25	560	3
71.59	2640	13	90.17	480	3
91.40	2480	12	481.57	320	2
107.13	320	2			

optimally within 30 minutes. Hence, we enforced a different time limit and solved the restricted master problem (RMP^1) in SINSA-NBD described in Algorithm 4.1. Only one nondominated solution was obtained leading us to incorporate the following *minimum nurses constraint* to the restricted master problem (RMP^1) along with time limit to enhance the algorithmic performance.

$$\sum_{j \in J} \left\{ \sum_{n \in N(j)} Y_{nj} + \sum_{n \in R(j)} Y^r_{nj} + \sum_{n \in O(j)} Y^o_{nj} + \sum_{n \in A(j)} Y^a_{nj} \right\} \geq lb + \lfloor rand * (ub - lb + 1) \rfloor . \tag{5.1}$$

where

$$rand = \text{random number}, \tag{5.2}$$

$$lb = 1, \tag{5.3}$$

$$ub = \left\lfloor \frac{|\mathbf{N}|}{|J|} \right\rfloor - \{|R| + |O| + |A|\} / |J|. \tag{5.4}$$

Table 5.6. Instance 3: solving SINSA-BD with different time limits within the SINSA-LRBD approach

Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
30 sec.			60 sec.		
39.79	3152	14	28.46	3168	14
73.98	896	5	44.80	2272	10
94.04	880	5	56.13	1952	9
152.28	720	4	60.97	1472	7
259.58	576	3	76.47	1456	7
310.57	560	3	79.55	1280	7
335.70	480	3	86.05	1200	6
818.26	416	2	117.92	1136	6
941.84	400	2	146.25	720	4
			251.79	560	3
			391.33	480	3
			818.26	400	2
			1886.86	320	2
120 sec.			300 sec.		
33.54	2912	13	20.42	3984	17
54.22	2608	12	22.57	3968	17
57.53	2592	12	25.22	3648	16
85.76	2336	11	27.07	3488	15
286.35	2096	10	28.19	3472	15
366.15	1696	9	30.81	3408	15
941.84	416	2	32.82	3392	15
			33.54	2912	13
			34.25	2848	13
			38.23	2832	13
			38.72	1216	6
			58.43	1136	6
			59.73	992	5
			61.97	960	5
			77.76	880	5
			132.52	800	4
			133.69	720	4
			242.50	640	4
			577.38	560	3

Table 5.7. Instance 4: solving SINSA-BD with different time limits within the SINSA-LRBD approach

Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
	30 sec.			60 sec.	
19.82	2128	10	17.39	2096	9
32.77	1136	6	19.80	1936	9
74.81	960	5	32.77	1136	6
134.09	896	5	74.81	960	5
159.01	880	5	134.09	896	5
			159.01	880	5
	120 sec.			300 sec.	
11.63	3072	13	8.82	3664	15
13.12	1712	8	8.93	3648	15
21.34	1632	8	9.26	3584	15
32.77	1136	6	9.45	3072	13
74.81	960	5	9.89	1552	7
134.09	896	5	14.02	1056	5
159.01	880	5	19.04	816	4
			28.46	560	3
			56.79	480	3
			159.01	400	2

The minimum nurses constraint (5.1) randomly changes the number of nurses required to be staffed for a shift. As the algorithm was forced to explore many nurse staffs with different numbers of nurses, more quality nondominated solutions were generated.

With constraint (5.1), we investigated the time limit for solving the restricted master problem (RMP^1) within SINSA-NBD that gave the best staffing and assignments. We solved the restricted master problem (RMP^1) with 30, 60, 120, and 300 seconds. Tables (5.8)-(5.11) display excess workload and staffing cost for solving the restricted master problem (RMP^1) with different time limits in SINSA-NBD for all four instances, respectively, and Figure 5.2 displays their efficient frontiers. Results illustrated that solving the restricted master problem (RMP^1) with 300 seconds gave minimum excess workload and staffing cost, therefore we included the minimum nurses constraint (5.1)

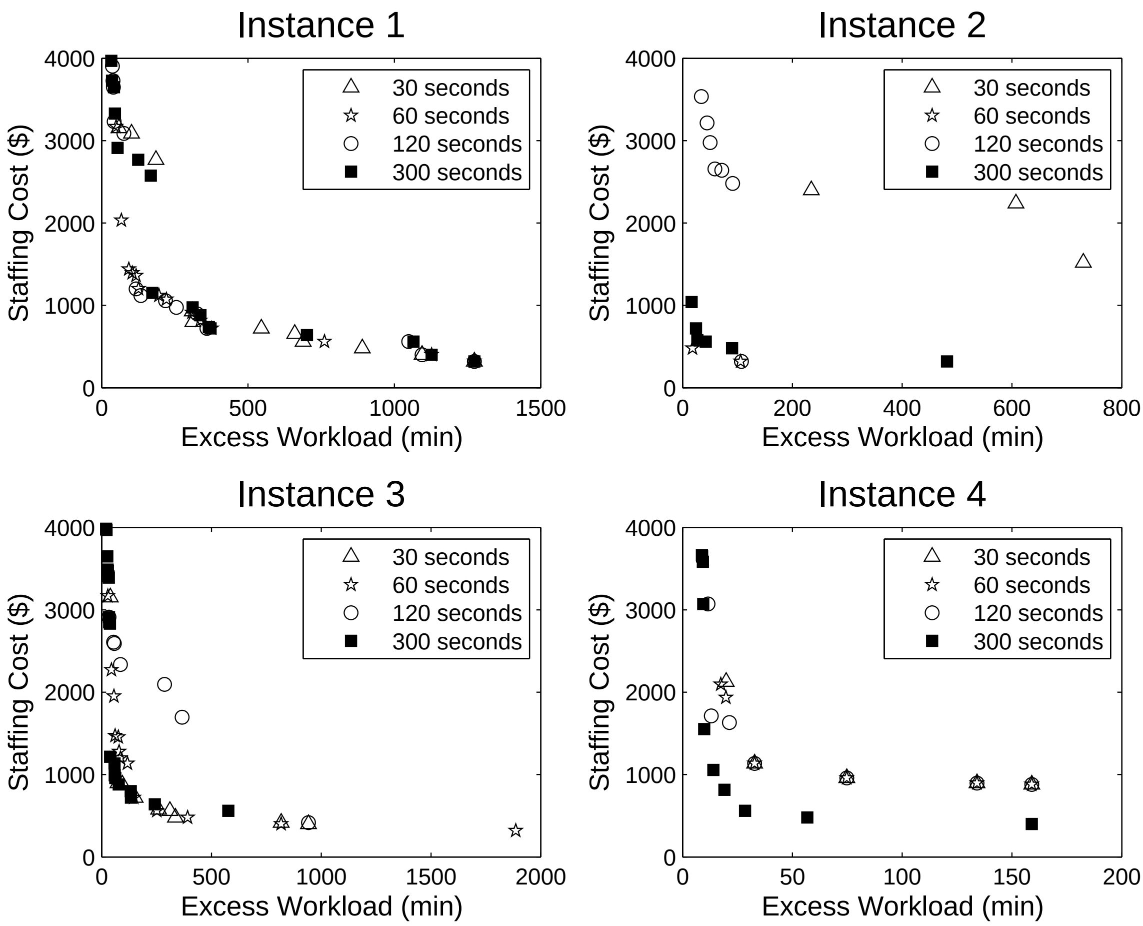


Figure 5.1. Selecting time limit (in seconds) for the SINSA-LRBD approach.

and solved the restricted master problem (RMP^1) with 300 seconds in SINSA-NBD in the remainder of this computational study.

5.2.2 Algorithmic Approaches Comparison

In this section, we evaluated the algorithmic performance of three solution methods, which were SINSA-BD, SINSA-LRBD, and SINSA-NBD. Tables 5.12-5.15 summarize the average excess workload in minutes, the staffing costs, and the total number of nurses scheduled for a shift with different solution methods for all four instances, respectively. The breakdown to the number of each type of nurses, i.e., PRN nurses, overtime nurses, and agency nurses, scheduled to work for a shift is displayed in Table B.1-B.4 in the Appendix. Figure 5.3 shows the efficient frontiers comparing average excess workload and

Table 5.8. Instance 1: solving the restricted master problem (RMP^1) with different time limits within the SINSA-NBD approach

Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
30 sec.			60 sec.		
81.18	1936	9	81.18	1936	9
824.20	480	3	781.85	720	4
			815.98	480	3
120 sec.			300 sec.		
81.18	1936	9	49.10	1936	9
628.29	896	4	618.68	896	4
679.76	640	3	675.97	640	3

Table 5.9. Instance 2: solving the restricted master problem (RMP^1) with different time limits within the SINSA-NBD approach

Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
30 sec.			60 sec.		
12.40	1968	10	12.40	1968	10
70.05	1376	6	33.62	1136	6
1195.89	960	4	1195.82	816	4
120 sec.			300 sec.		
6.81	1136	6	12.40	1968	10
474.17	736	4	45.41	1392	7

Table 5.10. Instance 3: solving the restricted master problem (RMP^1) with different time limits within the SINSA-NBD approach

Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
30 sec.			60 sec.		
57.40	1952	9	37.34	1952	9
526.90	800	4	530.81	800	4
606.33	560	3	599.26	560	3
120 sec.			300 sec.		
37.34	1952	9	22.60	1952	9
624.84	896	4	505.59	800	4
969.65	656	3	586.56	560	3

Table 5.11. Instance 4: solving the restricted master problem (RMP^1) with different time limits within the SINSA-NBD approach

Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
30 sec.			60 sec.		
19.19	1936	8	11.50	1936	8
128.82	640	3	128.76	640	3
120 sec.			300 sec.		
5.52	1936	8	3.34	1936	8
128.88	720	3	128.81	640	3

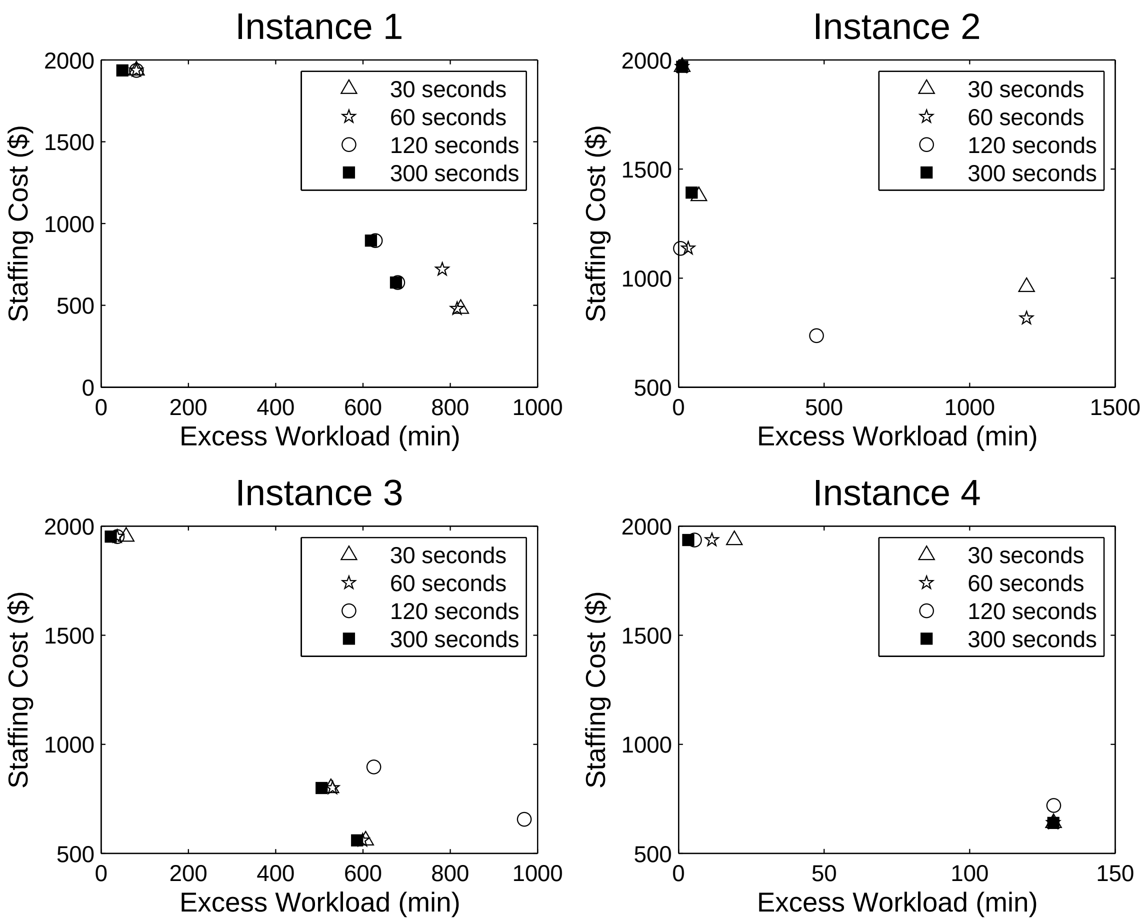


Figure 5.2. Selecting time limit (in seconds) for the SINSA-NBD approach.

Table 5.12. Instance 1 results comparing average excess workload and staffing cost from solving SINSA with three different approaches

SINSA-BD			SINSA-LRBD			SINSA-NBD		
Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
59.19	1856	8	32.83	3968	17	49.10	1936	9
294.79	1792	7	35.48	3728	16	618.68	896	4
302.61	1712	7	42.12	3648	16	675.97	640	3
470.36	1472	6	45.09	3328	15			
1291.76	1056	4	54.60	2912	13			
1339.31	896	3	125.21	2768	13			
			168.12	2576	12			
			172.93	1152	6			
			311.06	976	5			
			337.68	880	4			
			366.12	736	4			
			372.15	720	4			
			701.54	640	4			
			1065.57	560	3			
			1127.59	400	2			
			1273.56	320	2			

staffing cost from solving SINSA with three different approaches. In general, SINSA-LRBD generated more and better nondominated solutions than SINSA-BD and SINSA-NBD. The explanation is that SINSA-LRBD takes both excess workload on nurses and staffing cost into consideration resulting in favorable solutions. Results also suggested that simultaneously staffing and assigning nurses (as in two-stage) provided better solutions than sequentially considering them (as in three-stage). In addition, the nurse schedule and assignments found by these methods can be used in a nurse staffing decision supporting system for a nurse supervisor. Not only can the nurse supervisor make a revised nurse schedule based on the tradeoff between staffing cost and excess workload on nurses, but she also obtains assignments of nurses to patients. Table 5.16 lists the average CPU time in seconds that each method used to solve SINSA. The Lagrange multipliers converged in two instances causing less CPU time.

Table 5.13. Instance 2 results comparing average excess workload and staffing cost from solving SINSA with three different approaches

SINSA-BD			SINSA-LRBD			SINSA-NBD		
Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
19.18	1936	9	16.51	1040	5	12.40	1968	10
19.54	1872	8	24.35	720	4	45.41	1392	7
23.06	1856	8	26.92	576	3			
28.18	1152	6	42.25	560	3			
151.36	1072	5	90.17	480	3			
726.81	480	2	481.57	320	2			
751.49	400	2						

Table 5.14. Instance 3 results comparing average excess workload and staffing cost from solving SINSA with three different approaches

SINSA-BD			SINSA-LRBD			SINSA-NBD		
Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
25.68	1952	9	20.42	3984	17	22.60	1952	9
50.46	1936	9	22.57	3968	17	505.59	800	4
55.02	1920	9	25.22	3648	16	586.56	560	3
68.32	1856	8	27.07	3488	15			
92.18	1776	8	28.19	3472	15			
117.20	1632	7	30.81	3408	15			
431.96	1616	7	32.82	3392	15			
941.84	416	2	33.54	2912	13			
			34.25	2848	13			
			38.23	2832	13			
			38.72	1216	6			
			58.43	1136	6			
			59.73	992	5			
			61.97	960	5			
			77.76	880	5			
			132.52	800	4			
			133.69	720	4			
			242.50	640	4			
			577.38	560	3			

Table 5.15. Instance 4 results comparing average excess workload and staffing cost from solving SINSA with three different approaches

SINSA-BD			SINSA-LRBD			SINSA-NBD		
Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
8.44	1872	8	8.82	3664	15	3.34	1936	8
9.14	1808	8	8.93	3648	15	128.81	640	3
10.03	1792	8	9.26	3584	15			
10.25	1776	7	9.45	3072	13			
10.75	1696	7	9.89	1552	7			
12.87	1680	7	14.02	1056	5			
14.49	1616	7	19.04	816	4			
15.68	1600	7	28.46	560	3			
18.78	1552	7	56.79	480	3			
19.88	1440	6	159.01	400	2			
31.67	672	3						
162.13	640	3						

Table 5.16. Comparison of CPU time (seconds)

	CPU time (seconds)		
Instance	SINSA-BD	SINSA-LRBD	SINSA-NBD
1	1839.75	1831.86	938.72
2	1807.14	585.24	613.20
3	1816.30	1821.62	916.93
4	1805.43	541.13	351.63

5.2.3 Working in Primary Work Units Only vs. Floating to Other Units

In this section, we performed a computational experiment to evaluate two float assignment policies:

Policy 1: Regular nurses are restricted to work in their primary work units only.

Policy 2: Regular nurses are allowed to float to other units in which they are qualified.

In both policies, PRN nurses, overtime nurses, and agency nurses can be staffed to any units in which they are qualified to work. Given that SINSA-LRBD provided the best results among three approaches, we employed SINSA-LRBD to compare these two policies. In the Policy 1, we imposed regular nurses to work in their originally scheduled

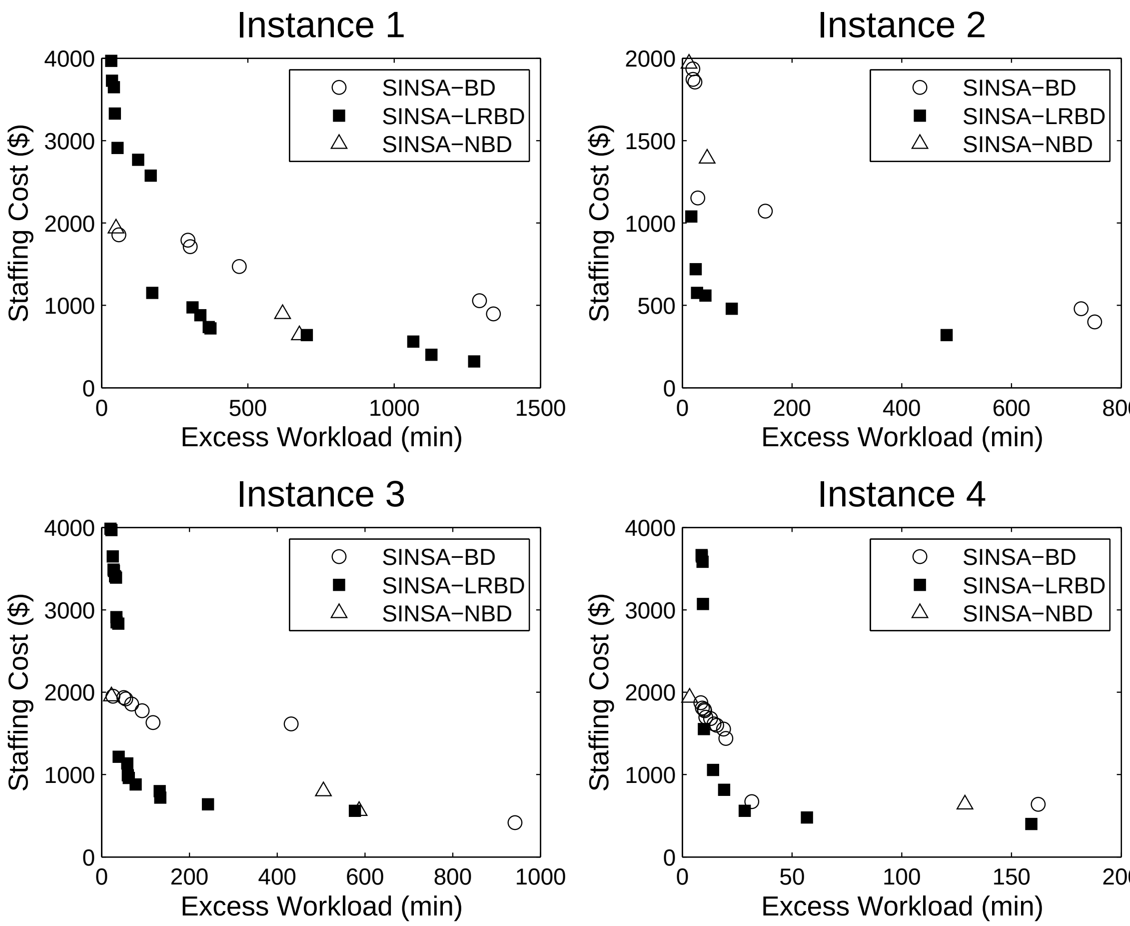


Figure 5.3. Efficient frontiers comparing average excess workload and staffing cost from solving SINSA with three different approaches for all four instances.

units (or their primary work units) only where PRN nurses, overtime nurses, and agency nurses were allowed to be staffed in Med-Surg1 and MedSurg2 units. In the Policy 2, all nurses including regular nurses, PRN nurses, overtime nurses, and agency nurses were free to float to both units. Tables 5.17-5.20 show the average excess workload and staffing cost of the nondominated schedules and assignments from two policies for all four instances. Figure 5.4 illustrates the efficient frontiers between average excess workload on nurses and staffing cost from two policies. Results indicated that working only in primary work units provided less excess workload with less staffing cost in Instances 2 and 3. The explanation is that regular nurses are more familiar with facilities and other procedure in their primary work units causing smaller excess workload for nurses. These

Table 5.17. Instance 1 comparison of Policy 1 vs. Policy 2

Policy 1			Policy 2		
Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
32.83	3968	17	31.45	3008	13
35.48	3728	16	68.22	2656	11
42.12	3648	16	105.65	1696	8
45.09	3328	15	123.73	1296	6
54.60	2912	13	136.78	1216	6
125.21	2768	13	176.14	1136	6
168.12	2576	12	245.37	736	4
172.93	1152	6	253.15	320	2
311.06	976	5			
337.68	880	4			
366.12	736	4			
372.15	720	4			
701.54	640	4			
1065.57	560	3			
1127.59	400	2			
1273.56	320	2			

results are consistent with academic literature [86]; nurses spend more time working in non-originally assigned units since they spend much time performing unit routines, searching for medical supplies, and caring for patients with unfamiliar diagnosis. A unit orientation, including unit routine introduction, patient care documentation overview, and assistants' phone numbers, can help float nurses to reduce nervous tension, time, and workload as well as increase quality of patient care [73, 86]. Moreover, the solution space was reduced by enforcing regular nurses to work in the primary units, resulting in finding quality solutions quicker. Instances 1 and 4 revealed the opposite results, floating regular nurses became helpful. The nurse supervisor can use this model along with her judgment to evaluate a float assignment policy based upon nurses' workload and staffing cost. As hospital administrations follow the right policy, they would reduce workload for nurses, increase care for patients, and reduce hospital budget.

Table 5.18. Instance 2 comparison of Policy 1 vs. Policy 2

Policy 1			Policy 2		
Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
16.51	1040	5	9.83	2096	9
24.35	720	4	22.43	1648	8
26.92	576	3	24.78	1392	6
42.25	560	3	44.40	880	5
90.17	480	3	153.24	832	4
481.57	320	2	153.71	560	3

Table 5.19. Instance 3 comparison of Policy 1 vs. Policy 2

Policy 1			Policy 2		
Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
20.42	3984	17	23.36	2848	12
22.57	3968	17	62.77	2608	12
25.22	3648	16	71.48	2544	10
27.07	3488	15	75.52	640	4
28.19	3472	15	156.28	480	3
30.81	3408	15	452.57	320	2
32.82	3392	15			
33.54	2912	13			
34.25	2848	13			
38.23	2832	13			
38.72	1216	6			
58.43	1136	6			
59.73	992	5			
61.97	960	5			
77.76	880	5			
132.52	800	4			
133.69	720	4			
242.50	640	4			
577.38	560	3			

Table 5.20. Instance 4 comparison of Policy 1 vs. Policy 2

Policy 1			Policy 2		
Excess workload (min)	Staffing cost ($)	No. of nurses	Excess workload (min)	Staffing cost ($)	No. of nurses
8.82	3664	15	6.10	2848	12
8.93	3648	15	7.51	2272	9
9.26	3584	15	10.91	1968	8
9.45	3072	13	11.54	1952	8
9.89	1552	7	11.62	1728	8
14.02	1056	5	11.95	1712	7
19.04	816	4	13.25	816	4
28.46	560	3	14.67	656	3
56.79	480	3	14.89	560	3
159.01	400	2	15.01	480	3
			56.63	320	2

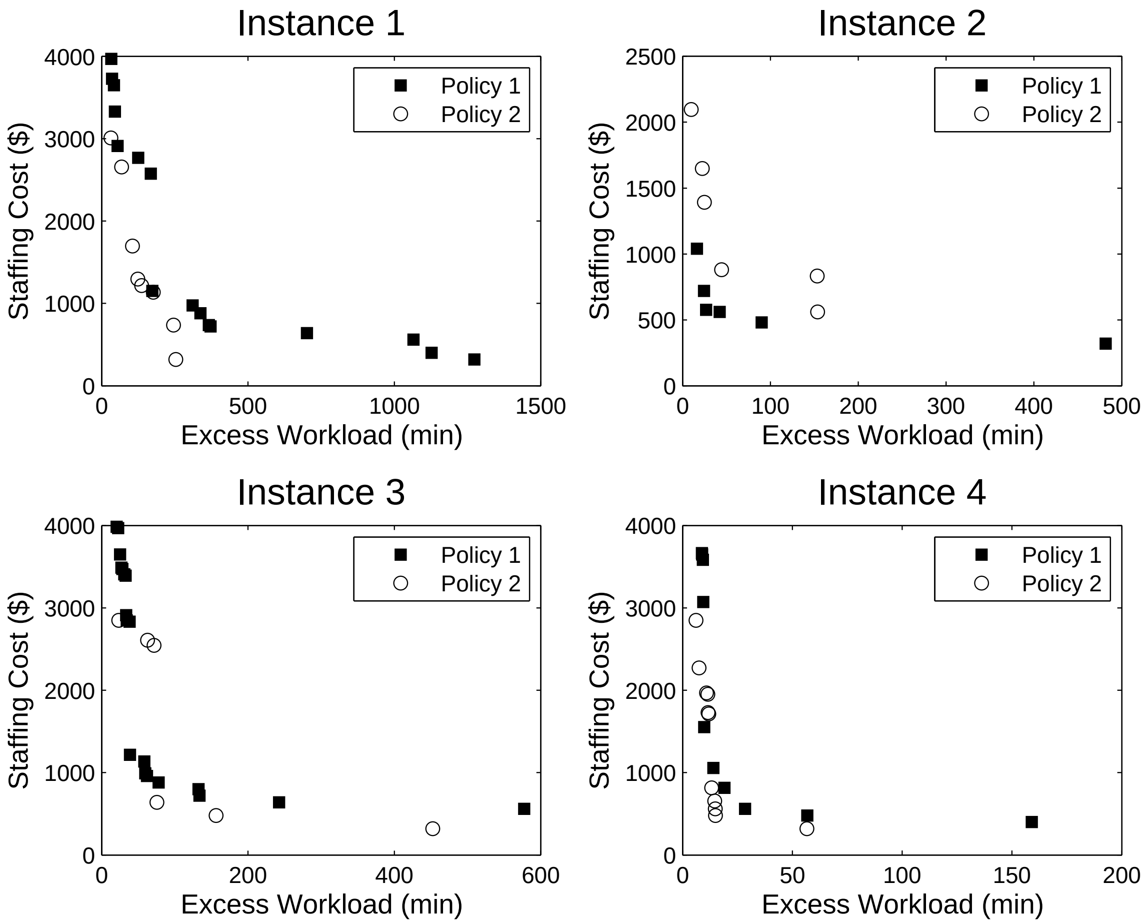


Figure 5.4. Efficient frontiers comparing average excess workload and staffing cost of Policy 1 vs. Policy 2 for all four instances.

CHAPTER 6

SUMMARY AND FURTHER RESEARCH

We developed a two-stage stochastic integer programming model for nurse assignment (SPA) with a recourse penalty function to minimize excess workload for nurses. Because of the special structure of the model, we employed the L-shaped method to solve our problem. Furthermore, we developed an optimal greedy algorithm to evaluate the recourse function. The computational results illustrated that our greedy algorithm is 30 times faster than the current commercial network simplex optimizer (CPLEX 9.1). Moreover, we discussed the symmetry issue that may arise when there are identical nurses. Sets of valid inequalities were proposed to improve the algorithmic performance as well as to reduce the symmetric assignments. We demonstrated that using SPA could save up to 1588 hours of excess workload on nurses per year in each medical-surgical unit. However, decisions made in earlier phases of nurse planning can have a dramatic effect on nurse assignment. Solutions for early phases that anticipate their consequences on nurse assignment would likely further reduce the burden of the nursing shortage. Observe that low EVPI upper bounds suggested that good sets of nurses scheduled for a shift reduced the necessity of perfect information as well as excess workload for nurses.

Consequently, we integrated nurse staffing and assignment within the same model. We presented the stochastic integrated nurse staffing and assignment model (SINSA) to capture patient care uncertainty with an objective to minimize an expected excess workload on nurses. We provided three solution approaches based on the L-shaped method, which are (1) Benders' decomposition, (2) Lagrangian relaxation with Benders' decomposition, and (3) nested Benders' decomposition. We demonstrated that our model can be considered as a two-stage stochastic program for the first two approaches and a three-stage stochastic program for the last approach. According to the Lagrangian relaxation with Benders' decomposition approach, the Lagrangian problem of SINSA can be viewed as a bicriteria programming problem in which both excess workload and staffing cost objectives are minimized. The Benders' decomposition embedded in Lagrangian relaxation formed a novel search approach for bicriteria stochastic integer programs where a Lagrange multiplier acted as a penalty for the second objective. Instead of solving the problem optimally, we presented alternative non-optimal ways to obtain good staffing and assignment solutions. We solved our model with three solution methodologies and we collected the nondominated solutions within 30 minutes. Results showed that the Lagrangian relaxation with Benders' decomposition provided the promising results among the three approaches, meaning that taking both excess workload on nurses and staffing cost into consideration is more beneficial. Simultaneously considering nurse staffing and assignment (as in two-stage stochastic program) is more desirable than sequentially doing them (as in three-stage stochastic program). We also provided efficient frontiers between excess workload and staffing cost of three solution approaches, which allow decision makers to play important roles in utilizing their judgments to comply the right staffing policy. Moreover, we demonstrated that our model can be used to to evaluate a float assignment policy based upon patients, available nurses of each type, and the budget for a shift.

Incorporating a nurse assignment within staffing decisions would likely provide better care for patients as well as balance workload for nurses. Hospitals also benefit from having better budget control, providing quality care to patients, and reducing liability cost. An integrated nurse staffing and assignment decision-support system that used our model would reduce the burden of the nursing shortage.

There are several interesting possibilities for future research. Because of the dynamic nature of the shift, patients are often admitted and discharged during a shift. One interesting topic is to consider how to assign newly admitted patients to nurses. Besides newly admitted patients, as patients go to run tests and come back to their units, they may not be assigned to the same nurse. A revised assignment balancing workload for nurses should be considered. Furthermore, our data included only primary diagnosis of patients, nurse types, and individual nursing skills. An extension to consider the following factors will likely to provide more accurate results.

- Multiple diagnoses. It is common for a patient to have multiple diagnoses during his/her stay in a hospital unit.
- Dynamic acuity. As the progress of patient's condition changes over time, the acuity level is changed. Patients with different levels of acuity require different amounts of required care from nurses.
- Educational level of nurses.

In addition, hiring additional nurses for a specific time (e.g. half of a shift) might be considered to provide better care to patients. Finally, incorporating the mid-term nurse scheduling into our model allowing feedback of a current shift for future corrections is another challenging area of research.

APPENDIX A

AN OPTIMIZATION-BASED INFORMATION TECHNOLOGY (IT) PROTOTYPE FOR NURSE ASSIGNMENT

In the Appendix A, we present the use of our nurse assignment model described in Chapter 2 in practice. We developed an optimization-based Information Technology (IT) prototype for a hospital unit with main function to assign nurses to a set of patients at the beginning of the shift. The users include charge nurses, administrative officer, and supervising nurses. Currently a nurse-patient assignment is done manually; therefore a computerized assignment tool can help a charge nurse with a cumbersome time-consuming task.

The organization in the appendix can be viewed as follows. In Section A.1, we summarize the underlying nurse assignment model of the IT prototype. In Section A.2, we present the structure of the IT prototype. In Section A.3, we describe an implementation and training of our IT prototype to potential users. In Section A.4, we summarize the results of the implementation and training. Finally, we discuss the conclusions and areas of future research of the IT prototype for nurse assignment in Section A.5.

A.1 An Underlying Model

We model the nurse assignment as a mixed-integer programming problem. The model uses decision variables, constraints, and input parameters based upon the nurses, the patients, and the shift information. The underlying model is the mean value problem described in Section 3.2. The input parameters include the amount of direct and indirect care that a nurse must provide to a patient in a time period if an assignment occurs. We obtained a direct care and indirect care parameter from mined encrypted data from a Northeast Texas hospital. Details about input parameter also can be found in Section 3.1.

A.2 IT Prototype Structure

In this section, the structure of IT prototype is presented. The IT prototype decision support system involved two software programs, Microsoft Excel and WinSCP, in a charge nurse personal computer. WinSCP is a freeware SFTP (Secure Shell file transfer protocol) for Microsoft Windows [109]. The IT prototype includes three main functions for the users:

1. Shift information entry

   The interface of the IT prototype is in Microsoft Excel. At the beginning of the shift, a user enters shift information containing nurses' and patients' information to a Microsoft Excel spreadsheet as follows:

   Shift information

   - Shift: day, evening, night,

   Patients' information

   - Number of patients,
   - Patient rooms,
   - Primary diagnosis,
   - Specific nurse requirement, i.e., a patient requires a special care from a certain type of nurse,
   - Admission time (if known),
   - Discharge time (if known),

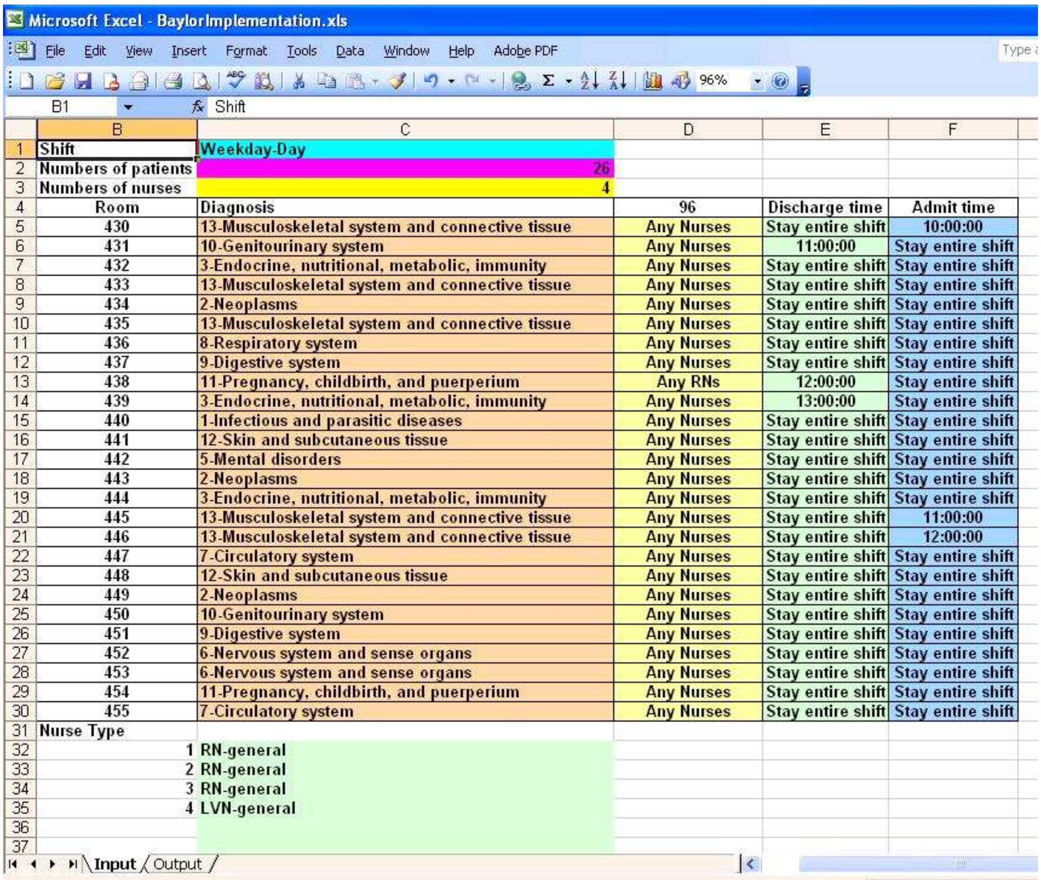
   Nurses' information

   - Number of nurses,
   - Nurse type: registered nurse (RN), licensed vocational nurse (LVN), nurse aid (NA), chemotherapy nurse, and pediatric nurse.

   Figure A.1 shows the Microsoft Excel interface, which is a spreadsheet for shift information entry. The IT prototype uses the visual basic macros to create a batch file containing current shift information.

2. Data transferring

   WinSCP is used to transfer a batch file from a user personal computer to a Dual 3.06-GHz Intel Xeon Workstation. The underlying model is solved by using CPLEX 9.1 callable library. The workstation calls an executable file and returns an optimal



	B	C	D	E	F
1	Shift	Weekday-Day			
2	Numbers of patients	26			
3	Numbers of nurses	4			
4	Room	Diagnosis	96	Discharge time	Admit time
5	430	13-Musculoskeletal system and connective tissue	Any Nurses	Stay entire shift	10:00:00
6	431	10-Genitourinary system	Any Nurses	11:00:00	Stay entire shift
7	432	3-Endocrine, nutritional, metabolic, immunity	Any Nurses	Stay entire shift	Stay entire shift
8	433	13-Musculoskeletal system and connective tissue	Any Nurses	Stay entire shift	Stay entire shift
9	434	2-Neoplasms	Any Nurses	Stay entire shift	Stay entire shift
10	435	13-Musculoskeletal system and connective tissue	Any Nurses	Stay entire shift	Stay entire shift
11	436	8-Respiratory system	Any Nurses	Stay entire shift	Stay entire shift
12	437	9-Digestive system	Any Nurses	Stay entire shift	Stay entire shift
13	438	11-Pregnancy, childbirth, and puerperium	Any RNs	12:00:00	Stay entire shift
14	439	3-Endocrine, nutritional, metabolic, immunity	Any Nurses	13:00:00	Stay entire shift
15	440	1-Infectious and parasitic diseases	Any Nurses	Stay entire shift	Stay entire shift
16	441	12-Skin and subcutaneous tissue	Any Nurses	Stay entire shift	Stay entire shift
17	442	5-Mental disorders	Any Nurses	Stay entire shift	Stay entire shift
18	443	2-Neoplasms	Any Nurses	Stay entire shift	Stay entire shift
19	444	3-Endocrine, nutritional, metabolic, immunity	Any Nurses	Stay entire shift	Stay entire shift
20	445	13-Musculoskeletal system and connective tissue	Any Nurses	Stay entire shift	11:00:00
21	446	13-Musculoskeletal system and connective tissue	Any Nurses	Stay entire shift	12:00:00
22	447	7-Circulatory system	Any Nurses	Stay entire shift	Stay entire shift
23	448	12-Skin and subcutaneous tissue	Any Nurses	Stay entire shift	Stay entire shift
24	449	2-Neoplasms	Any Nurses	Stay entire shift	Stay entire shift
25	450	10-Genitourinary system	Any Nurses	Stay entire shift	Stay entire shift
26	451	9-Digestive system	Any Nurses	Stay entire shift	Stay entire shift
27	452	6-Nervous system and sense organs	Any Nurses	Stay entire shift	Stay entire shift
28	453	6-Nervous system and sense organs	Any Nurses	Stay entire shift	Stay entire shift
29	454	11-Pregnancy, childbirth, and puerperium	Any Nurses	Stay entire shift	Stay entire shift
30	455	7-Circulatory system	Any Nurses	Stay entire shift	Stay entire shift
31	Nurse Type				
32	1	RN-general			
33	2	RN-general			
34	3	RN-general			
35	4	LVN-general			
36					
37					

Figure A.1. The shift information entry spreadsheet in Microsoft Excel.

assignment file. WinSCP is again used to transfer an assignment file to a user personal computer.

3. Optimal assignment display

   The optimal assignment output can be displayed in the Microsoft Excel spreadsheet with the visual basic macros. The IT prototype proposes a nurse-patient assignment specifying an assignment for each nurse to a group of her patients with the minimal excess workload for all nurses. Figure A.2 displays the optimal assignment output spreadsheet for users.

   The architecture of the IT prototype is presented in Figure A.3.

A.3 Implementation and Trainings

In this section, we describe a training seminar for the IT prototype decision support system to potential users. Subjects for this seminar were nursing students at The University of Texas at Arlington (UTA). The IT prototype was installed at the computer lab in the School of Nursing at UTA. We provided a training course to use the system and collect feedback from them. The training course was given to two nursing research classes at UTA, which are required for RN-BSN students (undergraduate-level class) and MS students (graduate-level class) in the summer 2006 semester. The training curriculum included

- A discussion of the factors involved in assigning nurses to patients, such as patient conditions and room locations,
- Background and motivation of the project,
- Descriptions of the collected data and mining results from the four units at the Northeast Texas hospital,
- Overviews of the optimization models within the IT prototype,
- Demonstrations of the IT prototype on several examples,
- Feedback from the nursing students on the models, IT prototype, and training curriculum.

There were 20 and 13 subjects in the RN-BSN and the MS class, respectively. In the training classes, every subject received the following materials: nurse assignment pre-survey, nurse assignment IT-prototype post-survey, presentation materials, a prototype instruction document, trouble shooting documentation, a census matrix, and a scenario for assignment. We began the training course with the pre-survey queried about their background on nurse-patient assignment, computer skills, number of patients assigned for each shift, potential ways to improve a nurse assignment, etc. After the pre-survey, we gave a presentation of the above curriculum, followed by the IT prototype demonstration with an instruction documentation. Then, we simulated a nurse assignment in a medical-surgical unit by providing subjects with a scenario of one shift in a medical-surgical unit and asked them to use the IT prototype program. The scenario contained patients' information that described patients' conditions, room location, admission time, discharge time, and type of nurse required. Subjects were asked to choose a number and a type of nurses from a census matrix. After having obtained shift information, subjects entered data and used the IT prototype to make an assignment. There were three research assistants to assist subjects if any problems occurred. One research assistant was in charge of technical support for the IT prototype while the other two nursing research assistants were responsible for nursing related inquiries. There was no time limit for subjects to use the systems. However, all subjects finished the assignment within 30 minutes. Subjects made a print out and submitted their assignment. Post-surveys were given to subjects to evaluate the IT prototype based on level of difficulty, usefulness, features, and advantage-disadvantage. Subjects were also asked to list recommendations to improve the IT prototype. We collected all feedback from subjects.

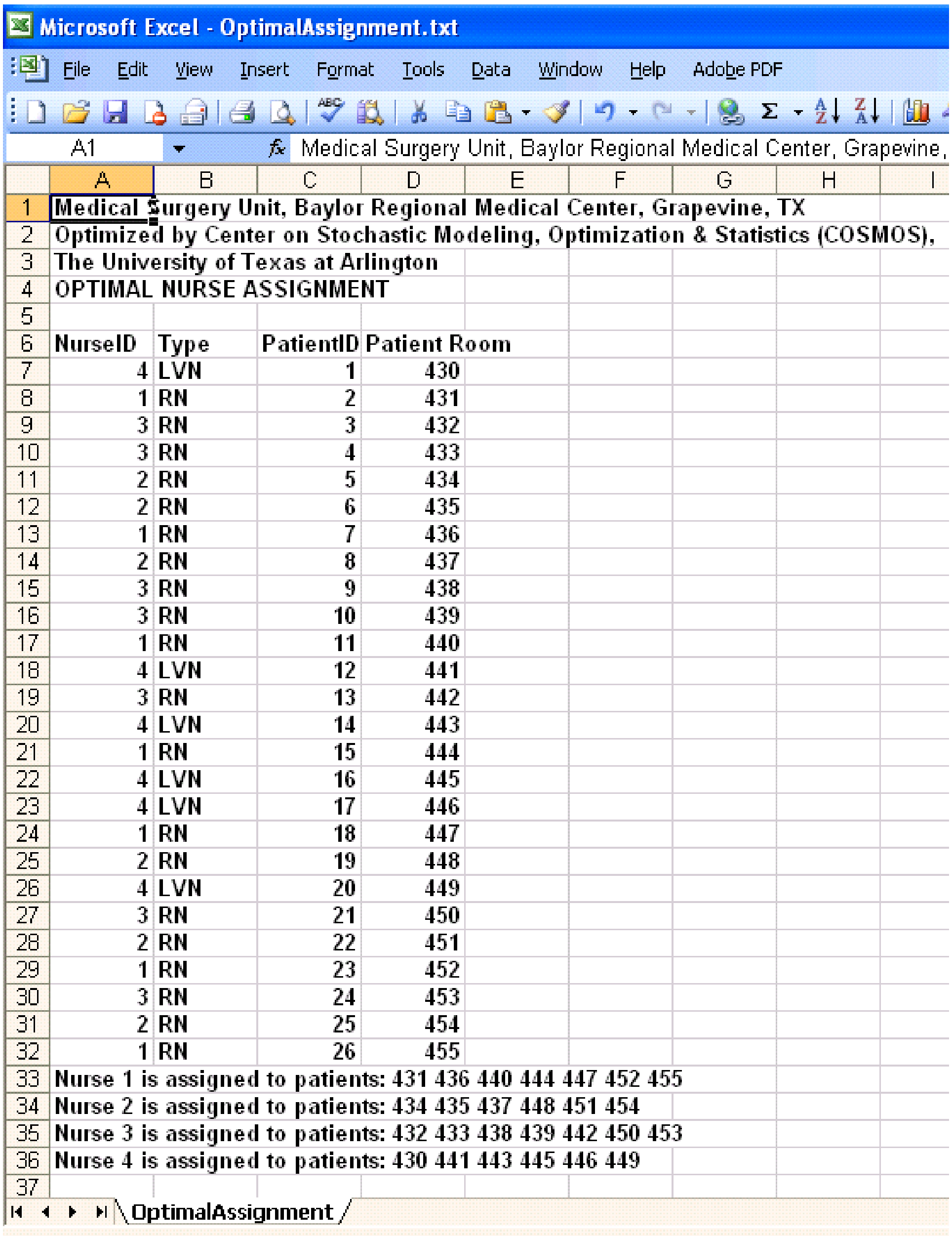
A.4 Summary of Results and Discussion

In this section, we present an overview for results from the pre- and post-surveys of the IT prototype. There are two similar sets of surveys given to nursing students; one for RN-BSN and another one for MS students. We obtained feedback from various professions in the health care industry. Almost all of the MS students were employed in several health care units. Seventy percent of RN-BSN students and more than half of MS students experienced greater than or equal to four patients for a shift. Having a high number of patients assigned, a nurse was likely to have job dissatisfactory and burnout. One-forth of RN-BSN and more than fifteen percent of MS students were not satisfied with the current assignment at work. We excluded one RN-BSN subject in the post-survey result because (s)he did not participate in the post-survey. More than forty percent of subjects reported that the IT prototypes were user friendly while thirty percent did not agree, and the remaining did not answered. We obtained positive feedback from 15 out of 17 RN-BSN students (88.26%) and 11 out of 13 MS students

(84.62%) supporting the use of IT prototype at their workplace. Two RN-BSN students were not eligible for this question because one worked in clinical setting and another one's hospice made an assignment by demographics. The reason that one subject did not support the IT prototype at her work was that it was too time consuming. Another subject did not answer this question. Subjects stated that the IT prototype benefited users for several reasons such as unbiased assignments, better assignments, speed, and decreased hand-written data. More details about survey results can be found at Baker et al. [12].

A.5 Conclusions and Future Research

We presented an optimization-based prototype for assigning nurses to patients for a nursing shift with minimal excess workload for all nurses. Instead of manually determining a nurse-patient assignment, it can now be promptly computerized by a charge nurses personal computer. We implemented the IT prototype and provided training courses to two groups of nursing students, RN-BSN and MS students at The University of Texas at Arlington. Subjects were asked to complete the pre- and post-survey to identify their background and opinion about the IT prototype. More than four-fifths of the subjects had positive feedback supporting the use of the IT nurse-patient assignment prototype at their workplace. One interesting topic of future research is to incorporate acuity and continuity of care into the model. Lastly, the IT prototype with a nicer user interface and fewer steps could potentially attract more attention from nurses.



Microsoft Excel - OptimalAssignment.txt

Medical Surgery Unit, Baylor Regional Medical Center, Grapevine, TX

Optimized by Center on Stochastic Modeling, Optimization & Statistics (COSMOS),

The University of Texas at Arlington

OPTIMAL NURSE ASSIGNMENT

NurseID	Type	PatientID	Patient Room
4	LVN	1	430
1	RN	2	431
3	RN	3	432
3	RN	4	433
2	RN	5	434
2	RN	6	435
1	RN	7	436
2	RN	8	437
3	RN	9	438
3	RN	10	439
1	RN	11	440
4	LVN	12	441
3	RN	13	442
4	LVN	14	443
1	RN	15	444
4	LVN	16	445
4	LVN	17	446
1	RN	18	447
2	RN	19	448
4	LVN	20	449
3	RN	21	450
2	RN	22	451
1	RN	23	452
3	RN	24	453
2	RN	25	454
1	RN	26	455

Nurse 1 is assigned to patients: 431 436 440 444 447 452 455

Nurse 2 is assigned to patients: 434 435 437 448 451 454

Nurse 3 is assigned to patients: 432 433 438 439 442 450 453

Nurse 4 is assigned to patients: 430 441 443 445 446 449

OptimalAssignment

Figure A.2. The optimal assignment spreadsheet in the Microsoft Excel.

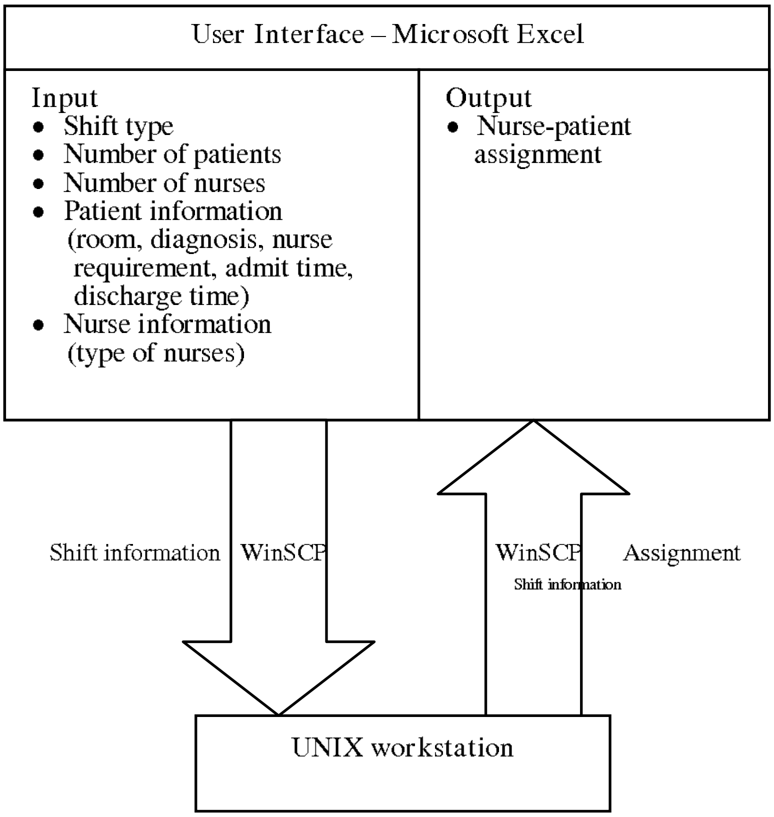


Figure A.3. The IT prototype structure.

APPENDIX B

ADDITIONAL TABLES

In the Appendix B, we provide additional tables from solving SINSA with three different approaches in Section 5.2.2.

Table B.1. Instance 1 results from solving SINSA with three different approaches

Excess workload (min)	Staffing cost ($)	No. of nurses	Med-Surg1				Med-Surg2			
			No. of				No. of			
			Reg	PRN	OT	Agen	Reg	PRN	OT	Agen
SINSA-BD										
59.19	1856	8	2	0	0	0	1	1	2	2
294.79	1792	7	1	1	1	1	0	1	1	1
302.61	1712	7	2	0	1	1	0	2	0	1
470.36	1472	6	1	0	2	0	0	2	0	1
1291.76	1056	4	0	0	2	0	0	1	0	1
1339.31	896	3	0	1	0	1	0	0	0	1
SINSA-LRBD										
32.83	3968	17	3	1	0	1	3	2	4	3
35.48	3728	16	3	1	1	1	3	2	2	3
42.12	3648	16	3	0	1	1	3	3	3	2
45.09	3328	15	3	1	2	1	3	2	2	1
54.60	2912	13	3	1	0	0	3	1	2	3
125.21	2768	13	3	0	1	0	3	3	2	1
168.12	2576	12	3	0	0	1	3	1	3	1
172.93	1152	6	2	0	0	0	2	2	0	0
311.06	976	5	2	0	0	0	1	1	1	0
337.68	880	4	1	0	0	0	1	0	1	1
366.12	736	4	1	0	0	0	2	1	0	0
372.15	720	4	1	0	0	0	2	0	1	0
701.54	640	4	1	0	0	0	3	0	0	0
1065.57	560	3	1	0	0	0	1	0	1	0
1127.59	400	2	1	0	0	0	0	0	1	0
1273.56	320	2	1	0	0	0	1	0	0	0
SINSA-NBD										
49.10	1936	9	2	1	0	0	3	0	1	2
618.68	896	4	1	1	0	0	1	0	0	1
675.97	640	3	1	0	0	0	1	0	0	1

Table B.2. Instance 2 results from solving SINSA with three different approaches

Excess workload (min)	Staffing cost ($)	No. of nurses	Med-Surg1 No. of Reg	PRN	OT	Agen	Med-Surg2 No. of Reg	PRN	OT	Agen
SINSA-BD										
19.18	1936	9	2	0	0	0	2	1	3	1
19.54	1872	8	1	0	0	0	2	2	1	2
23.06	1856	8	2	1	0	0	1	0	2	2
28.18	1152	6	2	1	0	0	2	1	0	0
151.36	1072	5	0	1	1	0	2	1	0	0
726.81	480	2	1	0	0	0	0	0	0	1
751.49	400	2	1	0	0	0	0	0	1	0
SINSA-LRBD										
16.51	1040	5	1	0	1	0	2	0	0	1
24.35	720	4	1	0	0	0	2	0	1	0
26.92	576	3	1	0	0	0	1	1	0	0
42.25	560	3	1	0	0	0	1	0	1	0
90.17	480	3	1	0	0	0	2	0	0	0
481.57	320	2	1	0	0	0	1	0	0	0
SINSA-NBD										
12.40	1968	10	2	0	0	0	4	3	1	0
45.41	1392	7	2	1	1	0	2	1	0	0

Table B.3. Instance 3 results from solving SINSA with three different approaches

Excess workload (min)	Staffing cost ($)	No. of nurses	Med-Surg1				Med-Surg2			
			No. of				No. of			
			Reg	PRN	OT	Agen	Reg	PRN	OT	Agen
SINSA-BD										
25.68	1952	9	3	1	0	0	1	1	2	1
50.46	1936	9	3	0	1	0	2	1	0	2
55.02	1920	9	2	0	1	1	3	0	1	1
68.32	1856	8	3	0	0	1	0	1	2	1
92.18	1776	8	1	0	2	0	2	1	1	1
117.20	1632	7	0	2	1	1	2	0	1	0
431.96	1616	7	1	1	2	1	1	0	1	0
941.84	416	2	1	0	0	0	0	1	0	0
SINSA-LRBD										
20.42	3984	17	3	1	1	0	3	3	2	4
22.57	3968	17	3	1	1	1	3	2	3	3
25.22	3648	16	3	1	0	2	3	2	4	1
27.07	3488	15	3	0	1	3	3	3	1	1
28.19	3472	15	3	0	1	0	3	2	2	4
30.81	3408	15	3	0	1	0	3	3	2	3
32.82	3392	15	3	2	1	1	3	0	3	2
33.54	2912	13	3	1	0	0	3	1	2	3
34.25	2848	13	3	1	1	0	3	2	1	2
38.23	2832	13	3	0	1	0	3	2	2	2
38.72	1216	6	2	0	0	0	1	1	2	0
58.43	1136	6	2	0	0	0	2	1	1	0
59.73	992	5	2	0	0	0	1	2	0	0
61.97	960	5	2	0	0	0	1	0	2	0
77.76	880	5	2	0	0	0	2	0	1	0
132.52	800	4	2	0	0	0	1	0	0	1
133.69	720	4	2	0	0	0	1	0	1	0
242.50	640	4	2	0	0	0	2	0	0	0
577.38	560	3	1	0	0	0	1	0	1	0
SINSA-NBD										
22.60	1952	9	2	1	1	0	2	1	1	1
505.59	800	4	1	0	1	0	1	0	1	0
586.56	560	3	1	0	0	0	1	0	1	0

Table B.4. Instance 4 results from solving SINSA with three different approaches

Excess workload (min)	Staffing cost ($)	No. of nurses	Med-Surg1				Med-Surg2			
				No. of				No. of		
			Reg	PRN	OT	Agen	Reg	PRN	OT	Agen
SINSA-BD										
8.44	1872	8	1	0	0	0	2	2	1	2
9.14	1808	8	2	1	0	0	1	2	1	1
10.03	1792	8	2	0	0	0	1	2	2	1
10.25	1776	7	1	0	0	0	1	1	1	3
10.75	1696	7	2	0	0	0	0	1	2	2
12.87	1680	7	1	0	0	0	2	0	1	3
14.49	1616	7	1	0	1	0	2	1	0	2
15.68	1600	7	2	0	0	1	1	0	2	1
18.78	1552	7	2	0	1	1	1	2	0	0
19.88	1440	6	1	0	0	0	2	0	0	3
31.67	672	3	1	0	0	0	0	2	0	0
162.13	640	3	1	0	1	0	0	0	1	0
SINSA-LRBD										
8.82	3664	15	2	2	1	1	2	2	2	3
8.93	3648	15	2	0	0	0	2	3	4	4
9.26	3584	15	2	1	2	0	2	3	2	3
9.45	3072	13	2	1	1	2	2	1	3	1
9.89	1552	7	2	0	0	0	1	2	1	1
14.02	1056	5	1	0	0	0	1	1	2	0
19.04	816	4	1	0	0	0	1	1	1	0
28.46	560	3	1	0	0	0	1	0	1	0
56.79	480	3	1	0	0	0	2	0	0	0
159.01	400	2	1	0	0	0	0	0	1	0
SINSA-NBD										
3.34	1936	8	1	0	0	1	2	1	1	2
128.81	640	3	1	0	0	1	1	0	0	0

REFERENCES

[1] AARP. A profile of older Americans: 2000. http://research.aarp.org/general/profile_2000.pdf, 2000.

[2] W. J. Abernathy, N. Baloff, J. C. Hershey, and S. Wandel. A three-stage manpower planning and scheduling model–a service-sector example. *Operations Research*, 21(3):693–711, 1973.

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

[4] U. Aickelin and K. A. Dowsland. An indirect genetic algorithm for a nurse scheduling problem. *Computers & Operations Research*, 31(5):761–778, 2003.

[5] L. H. Aiken, S. P. Clark, D. M. Sloane, J. Sochalski, and J. H. Silber. Hospital nurse staffing and patient mortality, nurse burnout, and job dissatisfaction. *Journal of the American Medical Association*, 288(16):1987–1993, 2002.

[6] L. H. Aiken, S. P. Clarke, D. M. Sloane, and J. A. Sochalski. Nurses reports of hospital quality of care and working conditions in five countries. *Health Affairs*, (20):43–53, 2001.

[7] R. Alonso-Zaldivar. US healthcare tab grows faster than the economy. *Los Angeles Times*, January 10, A15, 2006.

[8] American Association of College of Nursing. Nursing shortage fact sheet. http://www.aacn.nche.edu/Media/FactSheets/NursingShortage.htm, 2002.

[9] M. Anzai and Y. Miura. Computer program for quick work scheduling of nursing staff. *Medical Informatics*, 12:43–52, 1987.

[10] J. Arthur and A. Ravindran. A multiple objective nurse scheduling model. *IIE Transactions*, 13(1):55–60, 1981.

[11] M. N. Azaiez and S. S. A. Sharif. A 0-1 goal programming model for nurse scheduling. *Computers & Operations Research*, 32(3):491–507, 2005.

[12] R. L. Baker, D. F. B. Behan, K. Goss, P. Punnakitikashem, P. Turpin, and J. M. Rosenberger. A pilot study to evaluate a computerized prototype used to generate hospital staff nurses a patient/shift assignment. Technical Report Technical Report, The University of Texas at Arlington, Arlington, TX, 2006.

[13] J. F. Bard and H. W. Purnomo. Real-time scheduling for nurses in response to demand fluctuations and personnel shortages. In M.A. Trick and E.K. Burke, editors, *Proceedings of the 5th International Conference on the Practice and Theory of Automated Timetabling*, pages 67–87, Pittsburgh, PA, 2004.

[14] J. F. Bard and H. W. Purnomo. Hospital-wide reactive scheduling of nurses with preference considerations. *IIE Transactions*, 35(7):589–608, 2005.

[15] J. F. Bard and H. W. Purnomo. Short-term nurse scheduling in response to daily fluctuations in supply and demand. *Health Care Management Science*, 8(4):315–324, 2005.

[16] B. F. B. Behan. *Staffing ratio, patient diagnoses, direct and indirect nursing care time, and nurse level of education on weekend versus weekday staffing*. PhD thesis, Texas Woman's University, School of Nursing, 2007.

[17] J. F. Benders. Partitioning procedures for solving mixed variables programming problems. *Numerische Mathematics*, 4:238–252, 1962.

[18] I. Berrada, J. A. Ferland, and P. Michelon. A multi-objective approach to nurse scheduling with both hard and soft constraints. *Socio-Economic Planning Science*, 30(3):183–193, 1996.

[19] D. P. Bertsekas. *Nonlinear Programming: 2nd Edition*. Athena Scientific, 1999.

[20] J. R. Birge. Decomposition and partitioning methods for multistage stochastic linear programs. *Operations Research*, 33(5):989–1007, 1985.

[21] J. R. Birge. Stochastic programming computation and applications. *Journal on Computing*, 9(2):111–133, 1997.

[22] J. R. Birge and F. Louveaux. *Introduction to Stochastic Programming.* Springer, New York, New York, 1997.

[23] R. Blau and A. Sear. Nurse scheduling with a microcomputer. *Journal of Ambulance Care Management*, pages 1–13, 1983.

[24] M. J. Brusco and L. W. Jacobs. Cost analysis of alternative formulations for personnel scheduling in continuously operating organisations. *European Journal of Operational Research*, 86:249–261, 1995.

[25] J. Buchan. A certain ratio? the policy of implications of minimum staffing ratios in nursing. *Journal of Health Services Research & Policy*, 10:239–244, 2005.

[26] P. I. Buerhaus, D. O. Staiger, and D. I. Auerbach. Implications of an aging registered nurse workforce. *The Journal of the American Medical Association*, 283(22):2948–2954, June 2000.

[27] E. K. Burke, P. D. Causmaecker, and G. V. Berghe. A hybrid tabu search algorithm for the nurse rostering problem. In *SEAL'98: Selected papers from the Second Asia-Pacific Conference on Simulated Evolution and Learning on Simulated Evolution and Learning*, pages 187–194, London, UK, 1999. Springer-Verlag.

[28] E. K. Burke, P. D. Causmaecker, G. V. Berghe, and H. V. Landeghem. The state of the art of nurse rostering. *Journal of Scheduling*, 7(6):441–499, November 2004.

[29] E. K. Burke, P. D. Causmaecker, S. Petrovic, and G. V. Berghe. A disaggregation model of a flexible nurse scheduling support system. In *Proceedings of Congress on Evolutionary Computation*, number CEC2002, pages 1197– 1202, Honolulu, HI, 2002. IEEE Press.

[30] E. K. Burke, P. Cowling, P. D. Caumaecker, and G. V. Berghe. A memetic approach to the nurse rostering problem. *Applied Intelligence special issue on Simulated Evolution and Learning*, 15(3):199–214, 2001.

[31] California Department of Health Services. State health director announces proposed changes to nurse-to-patient ratio regulations.

http://www.applications.dhs.ca.gov/pressreleases/store/PressReleases/04-70.html, November 2004.

[32] C. C. Caroe and J. Tind. L-shaped decomposition of two-stage stochastic programs with integer recourse. *Mathematical Programming*, 83(3):451–464, November 1998.

[33] B. Cheang, H. Li, A. Lim, and B. Rodrigues. Nurse rostering problems-a bibliographic survey. *European Journal of Operational Research*, 151(3):447–460, December 2003.

[34] J. G. Chen and T. Yeung. Hybrid expert system approach to nurse scheduling. *Computers in Nursing*, pages 183–192, 1993.

[35] L. L. Curtin. An integrated analysis of nurse staffing and related variables: effects on patient outcomes. *Online Journal of Issues in Nursing*, 8(3), 2003.

[36] G. B. Dantzig. Linear programming under uncertainty. *Management Science*, 1:197–206, 1955.

[37] G. B. Dantzig and P. Wolfe. The decomposition principle for linear programms. *Operations Research*, (8):101–111, 1960.

[38] F. de Véricourt and O. B. Jennings. Nurse-to-patient ratios in hospital staffing: a queuing perspective. `http://faculty.fuqua.duke.edu/%7Efdv1/bio/ratios3.pdf`, 2006.

[39] A. J. Dieck and S. Yeralan. Forcasting in-patient admissions to public health facilities: a comparison of Box-Jenkins and Winters' approaches. In *Modeling and Simulation, Proceedings of the Fifteenth Annual Pittsburgh Conference*, volume 15, pages 1321–1325, Pittsburgh, PA, 1984. ISA.

[40] K. Dowsland. Nurse scheduling with tabu search and strategic oscillation. *European Journal of Operational Research*, 106:393–407, 1998.

[41] K. A. Dowsland and J. M. Thompson. Solving a nurse scheduling problem with knapsacks, networks and tabu search. *Journal of the Operational Research Society*, 51:825–833, 2000.

[42] J. Baileyand J. Field. Personnel scheduling with flexshift models. *Journal of Operations Management*, 5:327–338, 1985.

[43] M. L. Fisher. The lagrangian relaxation method for solving integer programming problems. *Management Science*, 50(12):1861–1871, 2004.

[44] L. S. Franz, H. M. Baker, G. K. Leong, and T. R. Rakes. Mathematical model for scheduling and staffing multiclinic health regions. *European Journal of Operational Research*, 41(3):277–289, 1989.

[45] GAO. Nursing workforce: Emerging nurse shortage due to multiple factors. Technical Report GAO-01-944, General Accounting Office, 2001.

[46] A. Gardner and E. M. Gemme. Virtual scheduling a 21st-century approach to staffing. *Nursing Administration Quarterly*, 27(1):77–82, 2003.

[47] L. Girion and R. Alonso-Zaldivar. Steep rise projected for health spending. *Los Angeles Times*, February 22, C1, 2006.

[48] L. M. Hall, D. Doran, G. R. Baker, G. H. Pink, S. Sidani, L. O'Brien-Pallas, and G. J. Donner. Nurse staffing models as predictors of patient outcomes. *Medical Care*, 41(9):1096–1109, 2003.

[49] D. E. Hecker. Occupational employment projections to 2014. *Monthly Labor Review, U.S. Bureau of Labor Statistics*, pages 70–101, 2005.

[50] M. Held, P. Wolfe, and H. D. Crowder and. Validation of subgradient optimization. *Mathematical Programming*, 6:62–88, 1974.

[51] J. L. Higle and S. Sen. *Stochastic Decomposition.* Kluwer Academic Publishers, Norwell, MA, 1996.

[52] M. Isken. An implicit tour scheduling model with applications in healthcare. *Annals of Operations Research*, 128:91–109, 2004.

[53] M. W. Isken and M. W. Hancock. Tactical scheduling analysis for hospital ancillary units. *Journal of the Society for Health Systems*, 5(4):11–24, 1998.

[54] A. Jaszkiewicz. A metaheuristic approach to multiple objective nurse scheduling. *Foundations of Computing and Decision Sciences*, 22(3):169–184, 1997.

[55] B. Jaumard, F. Semet, and T.Vovor. A generalized linear programming model for nurse scheduling. *European Journal of Operational Research*, 107(1):1–18, 1998.

[56] P. Kall and S. W. Wallace. *Stochastic Programming*. John Wiley & Sons, Chichester, U.K., 1994.

[57] E. P. C. Kao and M. Queyranne. Budgeting costs of nursing in a hospital. *Management Science*, 31(5):608–621, 1985.

[58] E. P. C. Kao and G. G. Tung. Forecasting demands for inpatient services in a large public health care delivery system. *Socio-Economic Planning Science*, 14(2):97–106, 1980.

[59] H. Kawanaka, K. Yamamoto, T. Yoshikawa, T. Shinogi, and S. Tsuruoka. Genetic algorithm with the constraints for nurse scheduling problem. In *Proceedings of Congress on Evolutionary Computation*, pages 1123–1130, Seoul, 2001. IEEE Press.

[60] G. Kazahaya. Harnessing technology to redesign labor cost management reports. *Healthcare Financial Management*, 59(4):94–100, 2005.

[61] M. M. Kostreva and K. S. B. Jennings. Nurse scheduling on a microcomputer. *Computers & Operations Research*, 18(9):731–739, December 1991.

[62] A. J. Lankshear, T. A. Sheldon, and A. Maynard. Nurse staffing and healthcare outcomes a systematic review of the international research evidence. *Advances in Nursing Science*, 28(2):163–174, 2005.

[63] H. Li, A. Lim, and B. Rodrigues. A hybrid ai approach for nurse rostering problem. In *Proceedings of the 2003 ACM symposium on Applied computing*, Melbourne, FL, 2003. IEEE Press.

[64] G. Lynn. The nursing shortage: Causes, impact and innovative remedies. `http://edworkforce.house.gov/hearings/107th/fc/nurses92501/lynn.htm`, 2001. Testimony of the American Hospital Association before the United State House of Representatives Committee on Education and the Workforce.

[65] A. Martel and J. Ouellet. Stochastic allocation of a resource among partially interchangeable activities. *European Journal of Operational Research*, 74(3):528–539, 1994.

[66] H. H. Millar and M. Kiragu. Cyclic and non-cylcic scheduling of 12h shift nurses by network programming. *European Journal of Operational Research*, 104(3):582–592, 1998.

[67] H. E. Miller, W. P. Pierskalla, and G. J. Rath. Nurse scheduling using mathematical programming. *Operations Research*, 24(5):857–870, 1976.

[68] M. Moz and M. Pato. Solving the problem of rerostering nurse schedules with hard constraints: New multicommodity flow models. *Annals of Operations Research*, 128:179–197, 2004.

[69] M. Moz and M. Pato. A genetic algorithm approach to a nurse rerostering problem. *Computers & Operations Research*, 34:667–691, 2007.

[70] C. Mullinax and M. Lawley. Assigning patients to nurses in neonatal intensive care. *Journal of the Operational Research Society*, 53(1):25–35, 2002.

[71] A. Musa and U. Saxena. Scheduling nurses using goal-programming techniques. *IIE Transactions*, 16(3):216–221, 1984.

[72] G. L. Nemhauser and L. A. Wolsey. *Integer and Combinatorial Optimization*. John Wiley & Sons, Inc., 1988.

[73] D. Nicholls, E. Duplaga, and L. Meyer. Nurses' attities about floating. *Nursing Management*, 27(1):56–58, 1996.

[74] M. Okada. An approach to the generalized nurse scheduling problem: generation of a declarative program to represent institution-specific knowledge. *Computers and Biomedical Research*, 25(5):417–434, 1992.

[75] M. Okada and M. Okada. Prolog-based system for nursing staff scheduling implemented on a personal computer. *Computers and Biomedical Research*, 21(1):53–63, 1988.

[76] R. O'Neill. Nested decomposition for multistage convex programs. *SIAM Journal of Control and Optimization*, 14:409–418, 1976.

[77] F. Overfelt. Nursing ratios–are they good or bad? In *Proceedings of The 16th Annual Society for Health Systems Management Engineering Forum, Orlando, Florida*, 2004.

[78] I. Ozkarahan. A disaggregation model of a flexible nurse scheduling support system. *Socio-Economical Planning Science*, 25(1):9–26, 1991.

[79] I. Ozkarahan and J. E. Bailey. Goal programming model subsystem of a flexible nurse scheduling support system. *IIE Transactions*, 16:306–316, 1988.

[80] J. Powers. Accepting and refusing assignments. *Nursing Management*, 24(9):64–73, September 1993.

[81] P. Punnakitikashem, J. M. Rosenberger, and D. B. Behan. Stochastic programming for nurse assignment. To appear in *Computational Optimization and Applications*, 2006.

[82] P. Punnakitikashem, J. M. Rosenberger, D. F. B. Behan, Robert L. Baker, and K. Goss. An optimization-based prototype for nurse assignment. In *Proceedings of the 7th Asia Pacific Industrial Engineering and Management Systems Conference*, pages 1080–1087, Bangkok, Thailand, December 2006.

[83] J. M. Rosenberger, D. B. Green, B. Keeling, and P. G. Turpin. Optimizing nurse assignment. In *Proceedings of The 16th Annual Society for Health Systems Management Engineering Forum*, Orlando, FL, 2004.

[84] E. S. Rosenbloom and N. F. Goertzen. Nurse scheduling using mathematical programming. *Operations Research*, 31:857–870, 1976.

[85] M. Rothberg, I. Abraham, P. Lindenauer, and D. Rose. Improving nurse-to-patient staffing ratios as a cost-effective safety intervention. *Medical Care*, 43(8):785–791, 2005.

[86] S. Rudy and J. Sions. Floating managing a recruitment and retention issue. *Journal of Nursing Administration*, 33(4):196–198, 2003.

[87] S. Scott and R. Simpson. Case-bases incorporating scheduling constraint dimensions - experiences in nurse rostering. In *Proceedings of the 4th European Workshop on Advances in Case-Based Reasoning*, pages 392–401, September 1998.

[88] S. Shaha and C. Bush. Fixing acuity: a professional approach to patient classification and staffing. *Nursing Economics*, 14:346–356, 1996.

[89] H. D. Sherali and J. C. Smith. Improving discrete model representations via symmetry considerations. *Management Science*, 47(10):1396–1407, October 2001.

[90] S. P. Siferd and W. C. Benton. A decision model for shift scheduling of nurses. *European Journal of Operational Research*, 74(3):519–527, 1994.

[91] D. Sitompul and S. Randhawa. Nurse scheduling models: a state-of-the-art review. *Journal of the Society for Health Systems*, 2(1):62–72, 1990.

[92] J. C. Smith, A. J. Schaefer, and J. W. Yen. A stochastic integer programming approach to solving a synchronous optical network ring design problem. *Networks*, 44(1):12–26, 2004.

[93] C. Sparkman. Focus on health care delivery, quality, and nursing. *AORN Journal*, 2005.

[94] M.W. Stanton and M. K. Rutherford. Hospital nurse staffing and quality of care. Research in Action Issue 14 04-0029, Agency for Healthcare Research and Quality, March 2004.

[95] D. Sundaramoorthi, V. C. P. Chen, and J. M. Rosenberger. Knowledge discovery and mining for nurse activity and patient data. In *Proceeding of IIE Annual Conference, Atlanta, Georgia*, 2005.

[96] D. Sundaramoorthi, V. C. P. Chen, J. M. Rosenberger, S. B. Kim, and D. B. Behan. A data-integrated simulation model to evaluate nurse-patient assignments. Technical Report Technical Report IMSE 06-05, The University of Texas at Arlington, Arlington, TX, 2006.

[97] The Kaiser Family Foundation and Agency for Healthcare Research and Quality and Harvard School of Public Health. National survey

on consumers experiences with patient safety and quality information. http://www.kff.org/kaiserpolls/7209.cfm, November 2004.

[98] V. M. Trivedi. Mixed-integer goal programming model for nursing service budgeting. *Operations Research*, 29(5):1019–1034, September-October 1981.

[99] V. M. Trivedi and M. Warner. A branch and bound algorithm for optimum allocation of float nurses. *Management Science*, 22(9):972–981, 1976.

[100] C. R. Victor and I. Higginson. Effectiveness of care for older people: A review. *Quality Health Care*, 3:210–216, 1994.

[101] S. W. Wallace and W. T. Ziemba. *Applications of stochastic programming.* MPS-SIAM, Philadelphia, PA, 2005.

[102] L. Walts and A. S. Kapadia. Patient classification system: An optimization approach. *Health Care Management Review*, 21:75–82, 1996.

[103] D. M. Warner. Forecasting the demand for nursing personnel and its optimal scheduling by ward. In *Proceeding of a Conference on Manpower Planning Models*, pages 465–474. Engl Univ Press, 1974.

[104] D. M. Warner. Scheduling nursing personnel according to nursing preferences: A mathematical programming approach. *Operations Research*, 24(5):842–856, 1976.

[105] D. M. Warner and J. Prawda. A mathematical programming model for scheduling nursing personnel in a hospital. *Management Science*, 19(4):411–422, 1972.

[106] M. Warner, B. Kelle, and S. Martel. Automated nurse scheduling. *Journal of the Society for Health Systems*, 2(2):66–80, 1990.

[107] K. M. White. Policy spotlight: staffing plans and ratios: What's the latest U.S. perspective? *Nursing Management*, 37(4):18–22, 2006.

[108] D. P. Wiens, J. Cheng, and N. C. Beaulieu. A class of method of moments estimators for the two-parameter gamma family. *Pakistan Journal of Statistics*, 19:129–141, November 2003.

[109] WinSCP. Freeware sftp and scp client for windows. http://www.winscp.com/, 2006.

[110] P. D. Wright, K. M. Bretthauer, and M. J. Côté. Reexamining the nurse scheduling problem: Staffing ratios and nursing shortages. *Decision Sciences*, 37(1):39–70, 2006.

BIOGRAPHICAL STATEMENT

Prattana "Sandy" Punnakitikashem was born on August 26th 1980 in Bangkok, Thailand. She received a bachelor's degree in Chemical Engineering from Chulalongkorn University, Bangkok, Thailand, in 2001 and a Master's degree in Industrial and Manufacturing Systems Engineering at the University of Texas at Arlington (UTA) in 2003. Before joining UTA, she had internships at the Petroleum Authority of Thailand Exploration & Production, Public Company Limited, the Unocal Corporation, the Sherwin Williams Company, and the Advance Paint & Chemical Public Company Limited. She works as a graduate research assistant at the Center on Stochastic Modeling, Optimization and Statistics (COSMOS) at UTA. She is a member of INFORMS, IIE, Alpha Pi Mu, and Tau Beta Pi. Her email address and web address are prattana.punnakitikashem@uta.edu and http://students.uta.edu/px/pxp1742/, respectively.