Abstract

1 Introduction

On-call schedules for a fixed number of health-care providers are central to the efficient running of hospitals. Hospital departments provide services where patient needs, and thus the system's demands, often exceed the available supply. For example, it is important that a hospital department allocates its resources, such as the availability of a finite number of clinicians, optimally, to ensure the best possible service for its patients. Carefully allocated on-call schedules are meant to simultaneously ensure sufficient resources are provided to patients while not overworking clinicians to prevent costly mistakes [ref]. It is common practice for on-call schedules to be created manually. Yet manually-created schedules are prone to errors and potential for biases [ref]. First, when there is a large number of clinicians in a single department, or the constraints that need to be satisfied by the department are very complex, a manual method may not provide an optimal schedule. Second, such methods are likely to overlook certain constraints that must be maintained to have an operational department, such as XXXX. Third, manual scheduling is often time-consuming for the person developing the schedule. For these reasons, it is important to develop automated methods that can generate optimal schedules that satisfy the given constraints of the hospital department.

Automated methods to optimize schedules have been studied and applied in many industries, including transportation [??], manufacturing [??], [...]. Of special interest to a clinician on-call scheduling problem are the approaches to schedule nurses, who often work in shifts. In the nurse scheduling problem, the goal is to find an optimal assignment of nurses to shifts that satisfies all of the hard constraints, such as hospital regulations, and as many soft constraints as possible, which may include nurse preferences. A wide variety of approaches, including exact and heuristic approaches, have been used to solve the nurse scheduling problem: integer linear programming [??], network flows [??], genetic algorithms [??], simulated annealing [??], and artificial intelligence [??].

[...] Many of these approaches were designed to satisfy the requirements of a specific hospital department which causes a large number of variables and constraints to be incorporated into the problem formulation. While these department-specific approaches allow end-users to find precise schedules that satisfy the needs of the department and the preferences of the nurses and clinicians in that department, they are difficult to readily adapt to other departments in the same hospital or other hospitals. Moreover, the large number of variables and constraints also leads to computational complexity issues [ref], especially when using exact methods for finding the solution. In this paper, we tackle a version of the nurse scheduling problem arising from a case study of one clinical division, providing two different services simultaneously (general infectious dis-

ease consults; and HIV consults service) at St. Michael's Hospital in Toronto, Canada. Our goal is to (1) present a simple formulation for the problem developed and tested at the hospital after switching from a manual approach to scheduling; and (2) analyze the performance of integer linear programming in solving difficult instances of the problem and compare the results with those of the manual approach; and (3) describe the adaptability of the formulation as a basic framework for solving similar problems in other departments.
We begin by describing the problem, then...

[...]

2 Problem

Clinicians at the Infectious Diseases (ID) and HIV departments of St. Michael's Hospital are scheduled throughout the year to receive patients during on-call hours. Each clinician is typically scheduled for a full week or a full weekend of on-call service. To prevent under- and over-working of clinicians, they each have a minimum and maximum number of weeks that they are required to work. Moreover, during holidays weekends, the work-load for on-call service increases drastically, and it is important to distribute these weekends equally. At the same time, clinicians often request to not be put on service during certain days or weeks, and the generated schedule should attempt to respect those requests as best as possible. Other considerations also need to be taken into account, such as making sure someone is available for on-call service at any time of the year, and preventing multiple back-to-back assignments for a single clinician. It is clear that ensuring all of these conditions are met manually, especially with an increasing number of clinicians, is a difficult task and can lead to mismanagement of the schedule. Therefore, in this paper we attempt to develop an efficient algorithm that can generate a satisfying schedule, while optimizing clinician and patient contentment.

In order to develop an algorithm, we need to formalize the variables and constraints of the problem mathematically. Table [??] presents the sets and indices that are used in the definition of the constraints. [...]

[...]

3 Methods

4 Results

5 Simulations

6 Discussion