Abstract

1 Introduction

In order for various hospital departments to function efficiently it is necessary for clinicians to have carefully allocated on-call schedules that simultaneously ensure sufficient resources are provided to their patients while not overworking clinicians to prevent *errors* (?). When generating these schedules, there are many variables and constraints that need to be taken into account, such as preventing multiple consecutive assignments for a given clinician or ensuring that assignments are spread out evenly throughout a certain period of time. It is very common for departments to opt for a manual method of generating these schedules [??]. Unfortunately, such methods are unreliable and can lead to unfavourable assignments, since a manual method may miss certain constraints or forget to account for a certain variable during the process.

Automated methods for scheduling in hospital settings have been studied extensively, most notably in the context of the Nurse Scheduling Problem (NSP) in Operations Research. In an NSP, the goal is to assign nurses to shifts, while satisfying various constraints pertaining to both hospital regulations and nurse preferences. The constraints can be classified as soft or hard constraints, based on the importance of satisfying them. Typically, it is necessary for a solution to satisfy all of the hard constraints, while the soft constraints are considered optional. The NSP, in its general case, is known to be NP-complete, meaning that although it is possible to efficiently check whether a given schedule satisfies all the constraints using an algorithm, it is not yet known whether an efficient algorithm for finding a satisfying schedule exists, and it is in fact equivalent to the hardest computational problems in Computer Science, Operations Research, and *etc...* [??]. In order to tackle NP-complete problems such as NSP in real-life, algorithms either try to use various heuristics to approximate a potential solution rather than finding an exact solution or they focus on relatively small problem sizes, ensuring that the runtime of the algorithm is still practical. *Some examples of heuristic approaches...* . *Some examples of exact approaches (LP, etc.)* . A comprehensive review of various approaches is presented by Burke et al. [??].

In this paper we present the application of two methods, Network Flows and Integer Linear Programming (ILP), to a variation of the Nurse Scheduling Problem in order to generate a yearly schedule for clinicians working simultaneously at two different divisions of St. Michael's Hospital in Toronto, ON. The paper is organized as follows. Section [??] contains a detailed description of the scenario and a mathematical formalization of the problem. Section [??] describes first the Network Flow approach followed by the ILP approach to tackle the problem. Section [??] presents the results of both approaches when applied to the clinician data from St. Michael's Hospital. Section [??] presents results from simulations with a variety of problem sizes. We conclude the paper with a discussion of the real-world and simulation results in Section [??].

2 Problem

3 Methods

4 Results

5 Simulations

6 Discussion