Abstract

The Nurse Scheduling Problem (NSP) is an NP-Hard problem in which the shifts of nurses in a hospital are scheduled. The problem is to design a schedule which fills $s$ shifts with no less than $n$ nurses per shift, while also taking into consideration the personal wishes and preferences of the nurses. The NSP affects hospital personnel all over the world on a daily basis, due to the unique complexity of scheduling shifts at hospitals.

Much work has been done to find heuristic pseudo-solutions to the NSP, including genetic algorithms, simulated annealing, and particle swarm optimization.

1. The Problem

The Nurse Scheduling Problem (NSP), or Nurse Rostering Problem (NRP), is a NP-Hard problem in which the shifts of nurses in a hospital are scheduled. The shift schedule is comprised of $s$ shifts per day, which each must be worked by no less than $n$ nurses. For example, let there be $s = 3$ shifts per day: a morning, afternoon, and night shift. Also, let $n = 5$. Then, for a schedule to be valid, each of the three shifts must be worked by at least five nurses.

The problem is to design a schedule which fills all $s$ shifts with at least $n$ nurses, thus satisfying the $n$ per $s$ constraint. While this constraint is the only constraint required for a valid solution (in the generic form of the problem), it does not take into consideration the quality of the solution.

To evaluate the quality of a valid schedule, additional constraints, namely, each nurse’s personal wishes and preferences in regard for the schedule, are also taken into consideration. For example, if a nurse does not wish to work a certain shift on a certain day, that fact should be acknowledged when the schedule is created. These preferences need not all be satisfied for a schedule to be valid, but the quality of the schedule is directly influenced by the number of personal wishes respected by the schedule. The more wishes that a schedule adheres to, the higher quality the schedule becomes.

Constraints for the NSP may vary depending on the variation of the problem. In general, these constraints can be divided into two categories, such that each constraint is either a “hard” or a “soft” constraint (Solos et al., 2013).

Hard constraints are restrictions that must be satisfied for the schedule to be valid, such as the $n$ per $s$ constraint described above, or a hospital’s Minimum Coverage Constraint (Augustine et al., 2009), which places a restriction on the minimum number of nurses working at one time, and/or on the minimum number of shifts a single nurse must work within a specified amount of time. Another hard constraint that may be imposed is the physical inability for a nurse to work the morning shift after having worked the night shift the night before. In some variations of the problem (Solos et al., 2013), time restrictions imposed by the nurses’ personal schedules are also considered as hard constraints. That is, if a nurse states that they simply cannot work due to a conflict with a non-hospital-related matter, that limitation must be considered. Other variations of the problem instead categorize these personal conflicts as soft constraints.

Soft constraints are restrictions on the schedule that do not influence the validity of the schedule, but which instead impact the quality of the schedule. As described earlier, the nurses’ personal preferences toward the schedule are soft constraints and are the largest factor of the schedule’s quality.

A good solution to this problem should be a schedule that accommodates as many of the nurses’ wishes as possible, without failing to adhere to any of the hard constraints.

2. Applications

The Nurse Scheduling Problem affects hospital personnel all over the world on a daily basis (Burke et al., 2004). Hospital schedules are different than schedules for most other industries, as different numbers of workers are needed at different times on different days, leading to a specific need for optimal scheduling. In addition, hospitals do not have time-off, so around-the-clock shifting is necessary. This irregular shifting pattern can affect the well-being and work ethic of the staff, which is why a schedule that takes into consideration the “wants” as well as the “needs” of the workers is ideal for a healthy work environment (Burke et al., 2004).

The issues described above have historically been addressed by “self-scheduling”, the ability of staff members to choose the days and shifts that they will work. Self-scheduling can promote responsibility, accountability, job satisfaction, and personal growth.
(Bluett, 2008). However, self-scheduling is slow and time consuming, and historically, there has been no automatic tool to test the quality of a schedule constructed in this manner (Burke et al., 2004).

Thus, it is desired to create an efficient automated solution which not only creates a valid schedule, but also still allows for a degree of self-scheduling. As the NSP is NP-Hard, work towards an optimal solution is continually ongoing.

3. History

Employee scheduling has been investigated in multiple fields of study, including computer science, personnel managing, and operations research for more than 40 years (Burke et al., 2004). Self-scheduling, a foundational concept for the NSP, has been used since as early as 1963, when Jenkinson implemented a self-scheduling program at St. George’s Hospital in London (Hung, 2002).

In their general form, timetabling and scheduling problems are non-deterministic polynomial time NP-Complete (Cooper & Kingston, 1995), resulting in the lack of an acceptable solution in polynomial time (White et al. 2004 & Yang, S., Jat, S. N. (2011)). Therefore, alternative optimization methods, such as heuristics and metaheuristics, have been developed in order to reach near-optimal solutions for various nurse scheduling problems (Burke et al., 2004 & Van den Bergh et al., 2013).

Much work has been done to find heuristic pseudosolutions for the NSP, including, but not limited to, genetic algorithms (Aickelin & Dowsland, 2004), Tabu search (Burke & Soubeiga, 2003), simulated annealing (Parr & Thompson, 2007), variable neighborhood search (Burke et al., 2003), scatter search (Burke et al., 2010), iterated local search (Bellanti et al., 2004), particle swarm optimization (Gunther & Nissen, 2010), memetic algorithms (Ozcan, 2005), and ant colony optimization (Gutjahr & Rauner, 2007).

Aickelin and Dowsland (2004) presented a genetic algorithm approach to the NSP which used indirect coding based on permutations of nurses and a heuristic decoder to build schedules from those permutations. They performed computational experiments based on 52 weeks of live data with four well-known crossover operators. They claim that the results of their experiment reveal that their proposed algorithm is able to find high quality solutions and is faster and more flexible than a comparative Tabu Search approach.

Aickelin and Dowsland (2004) also point out that genetic algorithm (GA) approaches to the NSP must overcome the classic GA conflict between constraints and objectives, since, unfortunately, there is no pre-defined way of including constraints into GAs. Methods for handling constraints do exist, but their application and success are problem specific.

References


