

2016

# NURSE ROSTERING PROBLEM: UN CASO APLICADO A UN HOSPITAL EN NUEVA ZELANDA

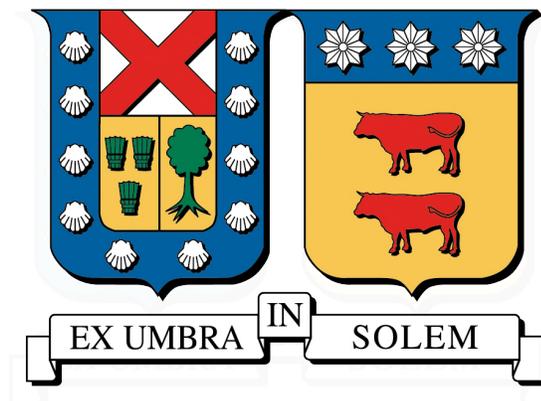
BARRAZA VARGAS, PATRICIO ALEJANDRO

---

<http://hdl.handle.net/11673/23662>

*Repositorio Digital USM, UNIVERSIDAD TECNICA FEDERICO SANTA MARIA*

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA  
DEPARTAMENTO DE INDUSTRIAS  
VALPARAÍSO - CHILE



**NURSE ROSTERING PROBLEM: UN CASO APLICADO A UN  
HOSPITAL EN NUEVA ZELANDA**

**PATRICIO ALEJANDRO BARRAZA VARGAS**

MEMORIA PARA OPTAR AL TÍTULO DE  
INGENIERO CIVIL INDUSTRIAL

PROFESOR GUÍA : Dr. PABLO ESCALONA R.  
PROFESOR CORREFERENTE : MSc. RAUL STEGMAIER B.

AGOSTO 2016

---

## RESUMEN EJECUTIVO

En este trabajo se estudia el problema de asignación de turnos de enfermeras (Nurse Rostering Problem) en un hospital de Nueva Zelanda. En este hospital, se trabajan tres turnos diarios, todos los días del año. El horizonte de programación se definió en cuatro semanas. Cada enfermera puede solicitar turnos libres, i.e., no trabajar en determinados turnos del horizonte de programación. El problema es formulado como un problema lineal entero (ILP) multiobjetivo, donde se requiere minimizar en forma simultánea la insatisfacción de las enfermeras producto del incumplimiento de sus solicitudes de turnos libres y la cantidad de enfermeras cuyos requerimientos no se satisfacen completamente. El enfoque de solución está basado en el *método de los pesos* que pondera cada objetivo por un factor de ponderación ( $\alpha$ ). Para determinar la calidad de la formulación propuesta y cubrir un amplio rango de datos, se generaron diez instancias. En cada instancia se modificó el factor de ponderación completando un total de cincuenta problemas. La factibilidad del problema depende del número de enfermeras disponibles que pueden trabajar en turnos de día (2 turnos) y noche (1 turno). Como se esperaba, a medida que se da mayor peso a la insatisfacción de turnos, la insatisfacción promedio disminuye y el promedio de enfermeras afectadas aumenta. Los problemas analizados permiten concluir que la formulación propuesta es tratable porque el tiempo promedio de uso de CPU fue 1.872 segundos y el peor caso fue de 24.406 segundos. Para las instancias probadas, se observa que la función objetivo es creciente en  $\alpha$  y que el mínimo de la función objetivo se alcanza en  $\alpha = 0$ .

**Palabras Clave.** Nurse rostering, Hospital Personnel Scheduling, Integer Linear Problem.

---

## ABSTRACT

In this work, we study the problem of assignment of nursing shifts (Nurse Rostering Problem) in a hospital in New Zealand. In this hospital, they work three shifts a day, every day of the year. The scheduling horizon was defined in four weeks. Each nurse can request shifts off, i.e., not work on certain shifts in the scheduling horizon. The problem is formulated as a multi-objective integer linear problem (ILP) where is required to simultaneously minimize the dissatisfaction of the nurses as result of the failure to fulfil the requirements and the number of nurses whose requirements are not fully satisfied. The solution approach is based on *weights method* which weights each objective by a weighting factor ( $\alpha$ ). To determine the quality of the proposed formulation and cover a wide range of data, ten instances were generated. In each instance, the weighting factor was changed for a total of fifty problems. The feasibility of the problem depends on the number of nurses available that can work in day shifts (2 shifts) and night shifts (1 shift). As expected, as more weight is given to the dissatisfaction of shifts, the average dissatisfaction decreases and the average number of nurses affected increases. The problems discussed allow us to conclude that the proposed formulation is treatable because the average CPU time was 1.872 seconds and the worst case was of 24.406 seconds. For the tested instances, we note that the objective function is increasing by  $\alpha$  and the minimum of the objective function is attained in  $\alpha = 0$ .

**Keywords.** Nurse rostering, Hospital Personnel Scheduling, Integer Linear Problem.

# Índice de Contenidos

|   |           |
|---|-----------|
| <b>1. Introducción y objetivos</b>  | <b>1</b>  |
| 1.1. Objetivos  | 2         |
| 1.1.1. Objetivo general   | 2         |
| 1.1.2. Objetivos específicos  | 3         |
| <b>2. Marco Teórico</b>   | <b>4</b>  |
| 2.1. La evolución de los sistemas de turno  | 5         |
| 2.1.1. Turno de 8 horas versus turno de 10 y 12 horas   | 5         |
| 2.1.2. El turno 4/40  | 6         |
| 2.2. Clasificación de turnos según patrones   | 7         |
| 2.2.1. Programación cíclica   | 7         |
| 2.2.2. Programación no cíclica  | 8         |
| 2.3. Revisión literaria   | 8         |
| 2.3.1. Programación Matemática: Lineal  | 9         |
| 2.3.1.1. An Implicit Tour Scheduling Model with Applications in Healthcare                                      | 9         |
| 2.3.1.2. Solving the problem of rostering nurse schedules with hard constraints: new multicommodity flow models | 9         |
| 2.3.2. Programación por metas y multiobjetivo   | 9         |
| 2.3.2.1. A multi criteria meta-heuristic approach to nurse rostering  | 10        |
| 2.3.3. Heurísticas  | 10        |
| 2.3.3.1. A subproblem-centric model and approach to the nurse scheduling problem                                | 11        |
| 2.3.3.2. A greedy-based neighborhood search approach to a nurse rostering problem                               | 11        |
| 2.3.4. Descripción del problema de Burke et al. (2010)  | 13        |
| 2.3.5. Descripción del problema de Lü y Hao (2012)  | 18        |
| <b>3. Formulación del modelo</b>  | <b>24</b> |
| <b>4. Estudio computacional</b>   | <b>31</b> |
| 4.1. Análisis de resultados   | 31        |
| 4.2. Análisis de tiempos computacionales  | 35        |
| <b>5. Conclusiones y trabajo futuro</b>   | <b>40</b> |

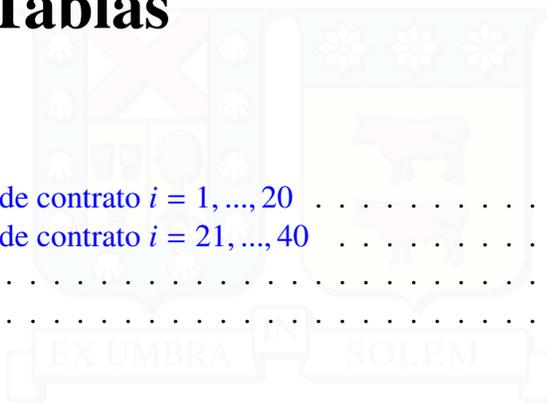
**Bibliografía**

**42**



# Índice de Tablas

|  |    |
|--|----|
| 4.1. Condiciones de contrato $i = 1, \dots, 20$ .....  | 32 |
| 4.2. Condiciones de contrato $i = 21, \dots, 40$ ..... | 32 |
| 4.3. $\overline{M}_{ks}$ .....                         | 32 |
| 4.4. $\underline{M}_{ks}$ .....                        | 32 |



# Índice de Figuras

|   |    |
|---|----|
| 4.1. Promedio de insatisfacción y enfermeras afectadas. . . . . | 33 |
| 4.2. Promedio de insatisfacción instancia 2. . . . .            | 34 |
| 4.3. Máximo de insatisfacción y enfermeras afectadas. . . . .   | 34 |
| 4.4. Máximo de insatisfacción instancia 4. . . . .              | 35 |
| 4.5. CPU time instancia 4. . . . .                              | 35 |
| 4.6. CPU time instancia 10. . . . .                             | 36 |
| 4.7. Box plot $\alpha = 0$ . . . . .                            | 36 |
| 4.8. Box plot $\alpha = 0,25$ . . . . .                         | 37 |
| 4.9. Box plot $\alpha = 0,5$ . . . . .                          | 37 |
| 4.10. Box plot $\alpha = 0,75$ . . . . .                        | 38 |
| 4.11. Box plot $\alpha = 1$ . . . . .                           | 38 |

# 1 | Introducción y objetivos

El problema de asignación de turnos de enfermeras en hospitales es un problema complejo al cual se enfrentan diariamente los hospitales. Este problema cobra especial importancia por tratarse de organizaciones enfocadas al servicio y con altos presupuestos gastados en personal. En las organizaciones que prestan servicios públicos el factor determinante de la operación es el servicio, lo que implica que el presupuesto es una función del servicio y por ende los presupuestos pueden subir o bajar dependiendo del servicio. De hecho el servicio proporcionado por las enfermeras consume entre el 25 % y 33 % del presupuesto [Welton et al. \(2006\)](#). Luego es relevante hacer una asignación de enfermeras que cumpla con los niveles de servicio pero que al mismo tiempo haga uso eficiente de los recursos.

Se han estudiado las consecuencias negativas de una asignación de turnos deficiente, tales como sobretiempo innecesario, frecuentes cambios en la programación, insatisfacción laboral, poca efectividad en el trabajo, las cuales generan ausentismo y alta tasa de rotación. Por otro lado, como la asignación de turnos determina qué días y horas deberán trabajar las enfermeras, estas decisiones afectan fuertemente la vida social y familiar de las enfermeras, así como también su salud por los tiempos de descanso que se ven reducidos. Todo esto tiene una inmediata repercusión en el paciente que recibe los cuidados ([Oldenkamp, 1996](#)).

El planteamiento y solución de la asignación de turnos afecta a distintos departamentos de todas las industrias y su estudio académico se puede abordar desde el área de Gestión de Operaciones. En la literatura este problema es conocido como *personnel scheduling* (programación de personal) y en particular para el caso de enfermeras es llamado *nurse rostering*. Estos problemas son combinatoriales, con gran número de parámetros y deben

satisfacer variadas restricciones, siendo catalogados por su complejidad computacional como NP-hard (Osogami y Imai, 2000).

Dada la complejidad de estos problemas el enfoque de solución son heurísticas, para hallar soluciones subóptimas que supuestamente son de buena calidad en un tiempo razonable (Tassopoulos et al., 2015). Entre ellos se cuentan algoritmos evolucionarios, *memetic algorithms*, *variable neighborhood search*, *scatter search*, *tabu search*, *iterated local search*, *ant colony optimization*, etc.

En la práctica es común resolver este problema manualmente, sin una herramienta que compruebe la calidad de la asignación construida. Incluso en hospitales con sistema computarizado no se aprovecha la capacidad del personal totalmente. Se ha estudiado como una mala asignación de turno afecta el estilo de vida del personal, aumentando niveles de insatisfacción y por lo tanto reduciendo el nivel de atención a los pacientes. Luego se hace necesario una buena asignación de turno, cercana al óptimo, más justa socialmente y que considere las preferencias personales de los trabajadores, de este modo se contará con una fuerza laboral más contenta y efectiva al realizar sus labores y además que permita ahorrar tiempo al personal administrativo a cargo de esta tarea (Burke et al., 2004).

Nuestro enfoque de solución al problema de Nurse Rostering es desarrollar un modelo de programación matemática que recoja todos los elementos de la instancia particular que nosotros vamos a analizar y luego determinar el tamaño de instancia que podemos resolver en tiempos polinomiales. Una extensión factible de nuestro enfoque de solución es desarrollar una heurística que respete las restricciones del problema de programación matemática y que permita resolver instancias de gran tamaño.

## 1.1. Objetivos

### 1.1.1. Objetivo general

El objetivo general de esta memoria es mejorar el proceso de asignación de enfermeras en un hospital en Nueva Zelanda. Esta mejora consiste en (i) reducir los tiempos que

el programador de turnos utiliza para la asignación de las enfermeras, (ii) minimizar la insatisfacción de las enfermeras producto del incumplimiento de sus solicitudes de turnos libres y (iii) minimizar el número de enfermeras afectadas por tales asignaciones.

### 1.1.2. Objetivos específicos

- Plantear un ILP multiobjetivo que minimice la insatisfacción de las enfermeras producto del no cumplimiento de sus solicitudes de turnos libres y el número de enfermeras afectadas por tales asignaciones.
- Resolver varias instancias y realizar un análisis de sensibilidad de los componentes de la función objetivo.
- Evaluar la calidad de las soluciones obtenidas en términos de satisfacción del staff.
- Obtener los CPU time y concluir la tratabilidad de las soluciones.

## 2 | Marco Teórico

El problema de Nurse Rostering (NRP) se define como la asignación de enfermeras en periodos de trabajo en varias semanas. De acuerdo a [Burke et al. \(2004\)](#) la mayoría de los hospitales realizan sus programaciones de personal de forma manual (*self scheduling*), siendo una tarea que consume gran cantidad de tiempo. Existen distintos acercamientos a la programación de personal, dependiendo del horizonte de tiempo. Nos concentraremos en el corto plazo. Las decisiones a largo plazo caen en la categoría de *staffing*. Se presentará una revisión de los principales conceptos utilizados, para esto se utilizarán los siguientes términos:

- Período de programación: intervalo de tiempo durante el cual el personal debe ser programado, a saber cuatro semanas.
- Clases de enfermera: determina un nivel de calificación del personal.
- Tipo de turno: noche, mañana y tarde. Estos dos últimos son referidos también como los turnos de día.
- Restricciones de cobertura: expresan el número de personal requerido de cada clase y para cada turno o intervalo de tiempo durante el periodo de programación. También conocido como demanda de personal.
- Restricciones de tiempo: se refieren a los requerimientos y preferencias personales de las enfermeras, también al balance de carga de trabajo.
- Restricciones fuertes: aquellas que deben ser satisfechas sin excepción.

- Restricciones débiles: aquellas que son deseables de satisfacer, pero pueden ser violadas para generar una programación.
- Regulaciones laborales: se refiere a los contratos que el personal tiene con el hospital.
- Período de programación: intervalo de tiempo durante el cual el personal debe ser programado.

## 2.1. La evolución de los sistemas de turno

[Warner \(1976\)](#) define cinco criterios en el área de scheduling para evaluar los sistemas de turnos.

1. Cobertura: Compara el número de personas programados y requeridos para una tarea.
2. Calidad: Cuan justa es una programación y que tan largo es la jornada laboral
3. Estabilidad: Cómo perciben la programación en términos de consistencia, predictibilidad en días libres.
4. Flexibilidad: Cómo se adapta a los cambios en el entorno.
5. Costo: Cuántos recursos se consumen en tomar las decisiones, es decir tiempo del administrador de personal o tiempo de computadoras.

Un asunto importante a decidir de un sistema de turno es el largo del turno, esto debe hacerse según los criterios mencionados anteriormente. Un turno 5/40 se refiere a 5 días, 40 horas a la semana, entonces el empleado trabaja 8 horas por día. Similarmente un turno 4/40 se refiere a una semana de trabajo de 4 días y 40 horas, es decir 10 horas diarias. En [Özcan \(2009\)](#) se analiza la evolución de los distintos sistemas.

### 2.1.1. Turno de 8 horas versus turno de 10 y 12 horas

Tradicionalmente el turno 5/40 ha predominado. Sin embargo siempre debe buscarse la satisfacción del personal, flexibilidad de la programación y calidad del cuidado de los

pacientes. Luego se ha buscado la opción de una semana de trabajo comprimida. El cambio puede hacerse en los días, por ejemplo pasar de un turno 5/40 a uno 4/40, o bien en horas, cambiando de un turno 4/40 a 4/36. También puede hacerse en horas y días a la vez al cambiar de un turno 5/40 a 3/36. Muchas organizaciones han elegido cambiar el número de días trabajados en una semana pero mantener el total de horas constante. Para cambios en horas de turno, la alternativa más popular al turno de 8 horas son los turnos de 10 y 12 horas.

Los turnos de 10 horas, comparado a los de 8 horas, otorgan más días consecutivos y fines de semana libres, además de una mayor dotación de personal en periodos más ocupados, y obviamente una mayor continuidad del cuidado del paciente. Sin embargo este turno requiere mayor número de personal y aumenta el cansancio de este. Además de esto, el turno de 12 horas requiere menos reportes de turnos (entregas de turno) y es sencillo para contratar nuevo staff.

### 2.1.2. El turno 4/40

El turno 4/40, de 10 horas, logró popularidad a comienzo de los años 70, aunque fue implementado por primera vez en los años 40 por las compañías Gulf and Mobil Oil. Hoy en día muchas enfermeras reconocen las ventajas de este turno. Crea un traslape de turnos que se puede diseñar de modo que permita enfrentar los peak de demanda, evitando reacomodar al staff y el sobretiempo. Por otro lado tener tres días libres es un beneficio a las enfermeras que aumenta su productividad y ánimo. Tantos días libres y el traslape de turnos aumenta el staff requerido para este tipo de turnos, aumentando los costos. Una modificación al tradicional turno 4/40 es la semana de ocho días, donde se trabajan 4 días de 10 horas y luego tiene un descanso de 4 días. Así dos turnos en 4/40 son compatibles para alternar los días libres y de trabajo a lo largo del año.

Largas horas de trabajo como los turnos de 12 horas, además de las ventajas mencionadas también reduce el número de turnos que el administrador de personal debe programar, pero agotan al personal. Algunas organizaciones permiten alternar las horas de trabajo para darse cuenta de los beneficios de cada uno. [Özcan \(2009\)](#) recopila algunos ejemplos

de modificaciones de programaciones y patrones de turnos en un intento de mejorar la contratación, satisfacción y retención de enfermeras, además de aspectos del cuidado del paciente. En ellos concluye que el éxito de un cambio en el largo de los turnos, depende de la organización, cada una debe evaluar sus objetivos y que aspectos de sus operaciones desea mejorar. El proceso de implementación es importante y las enfermeras deben participar de él retroalimentando con su experiencia. Para enfrentar las falencias de trabajar con los turnos de 8, 10 y 12 horas muchas organizaciones están cambiando a programaciones flexibles.

## 2.2. Clasificación de turnos según patrones

### 2.2.1. Programación cíclica

La programación cíclica o fija es utilizada cuando el día está dividido en distintos turnos y los requerimientos del personal y de turno obedecen un patrón cíclico. La programación es planificada por un periodo de 4 a 6 semanas y se repite periodo tras periodo. Dentro de sus ventajas [Warner \(1976\)](#) destaca el conocimiento de antemano que el personal tiene de la programación, el uso repetido de los bloques y la forma equitativa en que se divide el trabajo. También permite elegir el turno que mejor se acomoda a su horario, pero una vez elegido ya queda fijo a ese turno, lo cual genera dificultades al contratar alguna nueva enfermera si es que se presenta alguna renuncia o licencia prolongada, pues debe aceptar el turno exacto que queda vacante. La programación cíclica promueve una cobertura pareja, alta estabilidad y bajos costos de programación, sin embargo es inflexible a los cambios del entorno. Un problema importante es la poca flexibilidad que dificulta el ajuste a la fluctuante demanda de personal y preferencias personales. Por otro lado este tipo de programación requiere un alto nivel de precisión para determinar el número de personal con la habilidad justa para ciertas tareas. Por ejemplo en Chile se utiliza en todos los establecimientos de salud una programación cíclica, conocida como cuarto turno, donde se trabaja el primer día de 8:00 a 20:00 horas, luego el segundo día una noche de 20:00 a 8:00 horas, seguido de 48 horas libres. El bloque siguiente, se inicia a las 8:00 horas del quinto día.

### 2.2.2. Programación no cíclica

También conocida como discreta. Se dividen en dos categorías: escalonados y flexibles. En el sistema escalonado no cambian el número de horas trabajados por semanas y permite al empleado decidir cuando empezar a trabajar. Este sistema destaca por sus altos costos de implementación. En el sistema flexible se puede enfrentar las fluctuaciones de la demanda, contando con una staff base y aumentándolo diariamente según necesidades. Se utiliza un grupo de enfermeras part time y full time para usar en periodos de alta demanda. La organización debe ser flexible también al responder a las necesidades del personal para reducir las renunciaciones. Así las enfermeras eligen la programación que mejor se ajusta a sus necesidades. Se utiliza este sistema dado la alta inestabilidad del ambiente en los hospitales en cuanto a los costos, escasez de personal, contratación, etc. Adoptar una programación manual o automatizada, así como utilizar programación cíclica o no-cíclica son decisiones claves. Por ejemplo cuando se programa manualmente la flexibilidad es la mayor ventaja. Cuando el programa es cíclico la complejidad está en considerar las solicitudes del personal.

## 2.3. Revisión literaria

En los artículos de nurse rostering se encuentra que la mayoría de los avances en programación matemática utilizan una función objetivo, que es optimizada sujeta a ciertas restricciones. En la práctica son pocos los que se pueden implementar. En instancias de tamaño real el NRP es intratable, por lo tanto el objetivo es encontrar una buena solución factible. Los administradores desean generar rápidamente una programación que satisfaga las restricciones fuertes y la mayor cantidad de las restricciones débiles posibles. En este sentido las programaciones cíclicas son consideradas menos complicadas de construir y se utilizan heurísticas para este propósito.

A continuación se presentan las principales investigaciones recopiladas por [Burke et al. \(2004\)](#).

### 2.3.1. Programación Matemática: Lineal

#### 2.3.1.1. An Implicit Tour Scheduling Model with Applications in Healthcare

Isken (2004) desarrolla un implicit tour scheduling model como parte de una herramienta de análisis de programación de personal en un hospital en US. Aborda problemas reales que sin esta herramienta serían demasiado grandes para técnicas de programación matemática. El principal objetivo es reducir los costos de labor y alcanzar los requerimientos de cobertura que son variables, a lo largo del periodo de programación de una semana. Esto se logra determinando las horas de comienzo del turno y patrones diarios. Después de resolver el tour scheduling problem con un MIP solver, un paso de post-procesamiento considera la asignación de turnos a los patrones, sin cambiar el valor obtenido en la función objetivo.

#### 2.3.1.2. Solving the problem of rostering nurse schedules with hard constraints: new multicommodity flow models

Moz y Pato (2003) abordaron el problema de re-rostering en un hospital real. A diferencia de otros investigadores, que lo hacen a través de enfermeras flotantes para responder a cargas de trabajo no planeadas, buscaron reemplazos en la misma unidad. Además de satisfacer las restricciones fuertes, buscan minimizar la diferencia entre la programación original y la nueva, para no perturbar la vida privada del staff innecesariamente.

Se introducen dos integer multicommodity flow models, donde solo se toman en cuenta las restricciones fuertes. Un roster para  $n$  enfermeras se presenta como un  $n$ -commodities binary flow en un multi-level network. El segundo modelo se asemeja al primero, excepto que los turnos iguales son agregados en un nodo.

La programación matemática se restringe a optimizar un solo objetivo o criterio, por lo que no es flexible para trabajar con un ranking relativo o varios criterios.

### 2.3.2. Programación por metas y multiobjetivo

La programación por metas y multiobjetivo define un target para cada criterio y prioridades relativas para alcanzar estos objetivos. Se busca una solución lo más cercana posible

a los targets en el orden de prioridad dado.

### **2.3.2.1. A multi criteria meta-heuristic approach to nurse rostering**

Burke et al. (2002) presentan un nuevo enfoque multicriterio. El espacio criterio, el cual corresponde al espacio generado por los distintos criterios utilizados en las restricciones de tiempo, es mapeado a un espacio de preferencia con unidades adimensionales. Las mayores ventajas de este estudio es la posibilidad para los usuarios de expresar su preferencia por ciertas restricciones en vez de tener que utilizar parámetros de costos abstractos. Los pesos pueden controlar la compensación de las restricciones. Esto facilita el manejo de restricciones diferentes, tomando posibles rangos para el criterio en consideración. El método requiere mayor desarrollo (para manejar el amplio rango de restricciones) antes de ser llevado a un hospital.

### **2.3.3. Heurísticas**

La mayoría de los enfoques matemáticos aplican métodos exactos para hallar una programación realizable. Dada la complejidad NP-hard del problema, casi todas las publicaciones utilizan heurísticas para asignar los patrones de trabajo y atender las preferencias personales. Muchas heurísticas se han desarrollado para obtener programaciones de alta calidad en un tiempo computacional aceptable, pero sin un modelo matemático explícito. Para aplicarlas se requiere una clara formulación de los requerimientos del hospital y cuantificar la calidad de las programaciones sin ambigüedad. La mayoría de las heurísticas son desarrolladas para generar programaciones cíclicas, emulando la prueba y error que usualmente efectúan los encargados de personal.

En general se cree que las metaheurísticas son más adecuadas que otros métodos para generar una solución aceptable en casos donde las restricciones son muy exigentes y en casos donde una solución realizable es muy difíciles de encontrar. Las desventajas de las metheurísticas es que no encuentran un certificado de optimalidad, i.e., no se tiene un certificado de calidad de la solución.

### 2.3.3.1. A subproblem-centric model and approach to the nurse scheduling problem

Ikegami y Niwa (2003) introdujeron una formulación en programación matemática en Japón y resolvieron el problema con metaheurísticas. Un largo estudio de 315 unidades en 23 hospitales reveló interesantes datos para la descripción del problema. La rápida rotación de los turnos es muy común, es decir los turnos cambian muchas veces a la semana y los modelos existentes no son aplicables. El modelo presentado cubre categorías de habilidades, preferencias personales, tomando el resultado de los periodos de programación previos, de modo de equilibrar la carga laboral. Ellos distinguieron entre restricciones de enfermeras (que nosotros llamamos restricciones de tiempo) y restricciones de turno (llamadas restricciones de cobertura), descomponiendo el problema en subproblemas donde la programación de las enfermeras, excepto una, están fijas. Los algoritmos intentan repetidamente satisfacer las restricciones en diferentes subproblemas. El algoritmo tabu search genera buenos resultados para un problema con dos turnos en un tiempo computacional razonable. El algoritmo branch and bound se presenta para dos y tres turnos.

### 2.3.3.2. A greedy-based neighborhood search approach to a nurse rostering problem

Bellanti et al. (2004) presentan un algoritmo para resolver un problema particular en un hospital italiano. Se enfrentaron con muchas y detalladas restricciones, dividiéndolas en restricciones de cobertura y contractuales y operacionales. Se incorporaron algunas relajaciones como permitir una desviación entre los requerimientos de cobertura y el número real de enfermeras programadas. De esta investigación resultó un software que está actualmente en uso, mejorando las programaciones manuales con un tiempo computacional aceptable. Las soluciones iniciales se construyeron aplicando métodos de tabu search e iterated local search.

Un aporte similar al de Burke et al. (2004) reuniendo los avances más destacados se puede encontrar en Ernst et al. (2004) donde presenta una revisión de la literatura en staff scheduling y rostering, enfocándose en staff scheduling.

Desde el 2005 en adelante se pueden destacar entre los artículos más citados:

- [Tassopoulos et al. \(2015\)](#) presentaron un algoritmo two-phase variable neighborhood search para resolver NRP, demostrando su eficiencia en el First International Nurse Rostering Competition (INRC-2010). Consideran dos restricciones fuertes y dieciocho restricciones débiles. Asignando a cada restricción suave un costo asociado con el número de veces que se rompe la restricción para cada enfermera. Además cada restricción y enfermera tiene un peso asignado. Así la función objetivo consiste en minimizar la suma de los costos ponderado por el correspondiente peso.
- [Komarudin et al. \(2013\)](#) consideran esencial la interacción entre el nurse staffing y el rostering, entonces describen un procedimiento para el problema de staffing, tomando en cuenta requerimientos de rostering y restricciones simultáneamente, introduciendo el problema RQS (roster quality staffing problem). El artículo va más allá de una simple comparación de personal disponible y requerido. La metodología permite hacer recomendaciones para modificar la estructura de personal para mejorar los roster que se pueden alcanzar.
- [Lü y Hao \(2012\)](#) presentaron un método ANS (adaptive neighborhood search) para resolver el problema propuesto en INRC-2010. Tal como [Tassopoulos et al. \(2015\)](#) en la misma competencia consideraron dieciocho restricciones débiles y dos fuertes. Además de la formulación matemática, presentaron un algoritmo ANS integrando características originales. De hecho utilizan conjuntamente dos distintos movimientos de vecindades y de forma adaptada alternan a través de tres estrategias de búsqueda. Se hicieron experimentos demostrar la importancia de la combinación de las dos vecindades y de las estrategias intensivas, intermedias y diversificadas.
- [De Causmaecker y Vanden Berghe \(2011\)](#) propone una notación para la clasificación de un conjunto de problemas de rostering de personal, esta consiste en  $\alpha | \beta | \gamma$  utilizada en scheduling. Trabajan en base a los primeros intentos de caracterizar los NRP de [Burke et al. \(2004\)](#). Categoría  $\alpha$  se refiere a la descripción del entorno del personal con información acerca del número de staff, habilidades y disponibilidades. También

restricciones de carga laboral individual y organización de trabajo en períodos de tiempo como noche y fines de semana. Categoría  $\beta$  se refiere a las características del trabajo que describen el servicio entregado y la estructura de tiempo. Categoría  $\gamma$  se refiere a los objetivos de optimización y permite distinguir entre varias formas de apoyo a las decisiones. Esto permite clasificar todos los problemas presentados por [Burke et al. \(2004\)](#) y se presenta la clasificación de treinta artículos más. Se espera que con el desarrollo de esta notación se extienda a otros dominios de rostering de personal. Abrieron el website <http://ingenieur.kahosl.be/vakgroep/it/nurse/archive.htm> con el objetivo de que otros investigadores contribuyan en esta clasificación.

- [Glass y Knight \(2010\)](#) analizaron cuatro casos de referencia para estudiar la naturaleza del NRP. Destacando la estructura del problema fueron capaces de reducir el espacio solución. Luego con programación lineal entera mixta consiguieron soluciones óptimas a estos cuatro problemas, en una hora y media cada uno. También mejoraron el manejo de cambios de turnos y la continuidad entre periodos de rostering
- [Burke et al. \(2010\)](#) proponen un modelo multiobjetivo híbrido que combina programación entera y VNS (variable neighborhood search) para enfrentar un NRP con gran cantidad de restricciones. Primero con la programación entera se resuelve el subproblema que incluye todas restricciones fuertes y un subconjunto de las restricciones débiles. Luego con VNS se mejora el resultado previo, siendo las restricciones excluidas el principal enfoque. Las ventajas de este método se probaron en un hospital alemán comparándose con un algoritmo genético.

#### 2.3.4. Descripción del problema de [Burke et al. \(2010\)](#)

El problema está basado en el área de cuidados intensivo en un hospital alemán. Se debe planificar cuatro tipos de turnos: mañana, día, tarde y noche, en un período de 5 semanas se debe programar 16 enfermeras. Para especificar el problema, se introducen variables de holgura y excedente en las restricciones suaves y los objetivos son minimizar los valores de las variables individuales. Se formula el problema asociado a un periodo

de programación de cinco semanas en un modelo de programación entera, que puede ser alterado para adaptar a otros problemas con restricciones diferentes.

### Parámetros

- $I$  : Conjunto de enfermeras, indexado por  $i \in \{1, \dots, |I|\}$ .
- $I_t$  : Subconjuntos de enfermeras que trabajan 20, 32, 36 horas por semana respectivamente con  $I = I_1 \cup I_2 \cup I_3$ .
- $J = \{7, 14, 28, 35\}$ , conjunto de índices del último día de cada semana en el periodo de programación.
- $K = \{1, 2, 3, 4\}$ , conjunto de tipos de turnos, mañana, día, tarde y noche respectivamente, indexado por  $k$ .
- $K' = \{(2, 1), (3, 1), (3, 2), (1, 4)\}$ , conjunto de sucesiones de tipos de turno indeseables.
- $d_{jk}$  : Demanda de enfermeras para el turno  $k$  en el día  $j = \{1, \dots, 35\}$ .
- $m_i$  : Número máximo de días de trabajo para la enfermera  $i$  en el período de programación.
- $n_1$  : Número máximo de turnos de noche consecutivos en el periodo de programación.
- $n_2$  : Número máximo de días de trabajo consecutivos en el periodo de programación.
- $c_k$  : Cota superior deseable de asignaciones consecutivas del turno  $k$ .
- $g_t$  : Cota superior deseable de días de trabajo semanales para el  $t$ -ésimo subconjunto de enfermeras.
- $h_t$  : Cota inferior deseable de días de trabajo semanales para el  $t$ -ésimo subconjunto de enfermeras.

Variable de decisión

$$x_{ijk} = \begin{cases} 1 & \text{si enfermera } i \text{ trabaja turno } k \text{ el día } j \\ 0 & \sim \end{cases}$$

Restricciones fuertes

RF1 Requerimientos diarios de cobertura para cada tipo.

$$\sum_{i \in I} x_{ijk} = d_{jk} \quad \forall j \in \{1, \dots, 35\}, k \in K$$

RF2 Para cada día una enfermera no debe comenzar más que un turno.

$$\sum_{k \in K} x_{ijk} = d_{jk} \quad \forall i \in I, j \in \{1, \dots, 35\}$$

RF3 Número máximo de días de trabajo durante el periodo de programación.

$$\sum_{j=1}^3 5 \sum_{k \in K} x_{ijk} \leq m_i \quad \forall i \in I$$

RF4 Número máximo de fines de semana trabajando durante el periodo de programación.

$$\sum_{j \in J} 5 \sum_{k \in K} x_{ijk} \leq 3 \quad \forall i \in I$$

RF5 Máximo número de turnos de noche durante el periodo de programación.

$$\sum_{j=1}^{35} x_{ij4} \leq 3 \quad \forall i \in I$$

RF6 Ningún turno de noche entre dos turnos que no son de noche.

$$x_{i(j-1)4} - x_{ij4} + x_{i(j+1)4} \geq 0 \quad \forall i \in I, j \in \{2, \dots, 34\}$$

RF7 Mínimo dos días libres después de una serie de turnos de noche.

$$x_{i(j-1)4} - \sum_{k=1}^3 x_{ijk} + \sum_{k=1}^3 x_{i(j+1)k} \leq 1 \quad \forall i \in I, j \in \{2, \dots, 34\}$$

$$x_{i(j-1)4} + \sum_{k=1}^3 x_{ijk} - \sum_{k=1}^3 x_{i(j+1)k} \leq 1 \quad \forall i \in I, j \in \{2, \dots, 34\}$$

$$x_{i(j-1)4} + \sum_{k=1}^3 x_{ijk} + \sum_{k=1}^3 x_{i(j+1)k} \leq 2 \quad \forall i \in I, j \in \{2, \dots, 34\}$$

RF8 Número máximo de turnos de noche consecutivos.

$$\sum_{j=r}^{r+n_1} x_{ijk} \leq n_1 \quad \forall i \in I, r \in \{1, \dots, 35 - n_1\}$$

RF9 Número máximo de días de trabajo consecutivo.

$$\sum_{j=r}^{r+n_2} \sum_{k \in K} x_{ijk} \leq n_2 \quad \forall i \in I, r \in \{1, \dots, 35 - n_2\}$$

RF10 Ningún turno de tarde para una enfermera en particular.

$$x_{16(j)3} = 0 \quad \forall j \in \{1, \dots, 35\}$$

### VARIABLES DE HOLSURA Y EXCEDENTE

Corresponden a las desviaciones positivas y negativas de los objetivos individuales.

$$\begin{aligned}
 s_{ij}^1, s_{ij}^2 &\geq 0 & \forall i \in I, j \in J \\
 s_{ij}^3, s_{ij}^4 &\geq 0 & \forall i \in I, j \in \{2, \dots, 34\} \\
 s_{irk}^5 &\geq 0 & \forall i \in I, r \in \{1, \dots, 32\}, k \in \{1, 3\} \\
 s_{ijk}^6 &\geq 0 & \forall i \in I, j \in \{2, \dots, 34\}, k \in \{1, 3\} \\
 s_{iiv}^7, s_{iiv}^8 &\geq 0 & \forall t \in \{1, 2, 3\}, i \in I_t, w \in \{1, 2, 3, 4, 5\} \\
 s_{ir}^9 &\geq 0 & \forall i \in I_1, r \in \{1, \dots, 32\} \\
 s_{ijk'}^{10} &\geq 0 & \forall i \in I, j \in \{1, \dots, 34\}, k' = (k_1, k_2) \in K'
 \end{aligned}$$

### Restricciones débiles

RD1 Fines de semana completos, es decir ningún turno o bien dos turnos el fin de semana.

$$\sum_{k \in K} [x_{i(j-1)k} - x_{ijk}] + s_{ij}^1 - s_{ij}^2 = 0 \quad \forall i \in I, j \in J$$

RD2 Evitar un día de trabajo entre dos días libres.

$$\sum_{k \in K} [x_{i(j-1)k} - x_{ijk} + x_{i(j+1)k}] + s_{ij}^3 \geq 0 \quad \forall i \in I, j \in \{2, \dots, 34\}$$

RD3 Número mínimo de días libres después de una serie de turnos.

$$\sum_{k \in K} [x_{i(j-1)k} - x_{ijk} + x_{i(j+1)k}] - s_{ij}^4 \leq 1 \quad \forall i \in I, j \in \{2, \dots, 34\}$$

RD4 Número máximo/mínimo de asignaciones consecutivas de turnos de mañana y tarde.

$$\sum_{j=r}^{r+3} x_{ijk} - s_{irk}^5 \leq c_k \quad \forall i \in I, j \in \{1, \dots, 32\}, k \in \{1, 3\}$$

$$x_{i(j-1)k} - x_{ijk} + x_{i(j+1)k} + s_{ijk}^6 \geq 0 \quad \forall i \in I, j \in \{2, \dots, 34\}, k \in \{1, 3\}$$

RD5 Número máximo/mínimo de días de trabajo semanal.

$$\sum_{j=7w-6}^{7w} \sum_{k \in K} x_{ijk} - s_{iiv}^7 \leq g_t \quad \forall t \in \{1, 2, 3\}, i \in I_t, w \in \{1, 2, 3, 4, 5\}$$

$$\sum_{j=7w-6}^{7w} \sum_{k \in K} x_{ijk} + s_{iiv}^8 \geq h_t \quad \forall t \in \{1, 2, 3\}, i \in I_t, w \in \{1, 2, 3, 4, 5\}$$

RD6 Número máximo de días de trabajo consecutivo para enfermeras part-time.

$$\sum_{j=r}^{r+3} \sum_{k \in K} x_{ijk} - s_{ir}^9 \leq 3 \quad \forall i \in I_1, r \in \{1, \dots, 32\}$$

RD7 Evitar cierto tipo de sucesiones de turno.

$$x_{ijk_1} + x_{i(j+1)k_2} - s_{ijk'}^{10} \leq 2 \quad \forall i \in I, j \in \{1, \dots, 34\}, k' = (k_1, k_2) \in K'$$

### Función objetivo

$$\text{mín } F(x) = [f_1(x)^T, f_2(x)^T, f_3(x)^T, f_4(x)^T, f_5(x)^T, f_6(x)^T, f_7(x)^T]^T$$

donde cada función vector está definida por

$$\begin{aligned} f_1(x) &= [s_{ij}^1, s_{ij}^2]^T \quad \forall i \in I, j \in J \\ f_2(x) &= [s_{ij}^3, s_{ij}^4]^T \quad \forall i \in I, j \in \{2, \dots, 34\} \\ f_3(x) &= [s_{irk}^5]^T \quad \forall i \in I, j \in \{1, \dots, 32\}, k \in \{1, 3\} \\ f_4(x) &= [s_{ijk}^6]^T \quad \forall i \in I, j \in \{2, \dots, 34\}, k \in \{1, 3\} \\ f_5(x) &= [s_{iiv}^7, s_{iiv}^8]^T \quad \forall t \in \{1, 2, 3\}, i \in I_t, w \in \{1, 2, 3, 4, 5\} \\ f_6(x) &= [s_{ir}^9]^T \quad \forall i \in I_1, r \in \{1, \dots, 32\} \\ f_7(x) &= [s_{ijk'}^{10}]^T \quad \forall i \in I, j \in \{1, \dots, 34\}, k' = (k_1, k_2) \in K' \end{aligned}$$

El número de objetivos en el modelo presentado podría ser reducido al resumir las variables de excedente y holgura que están asociadas con restricciones suaves del mismo tipo y que tienen el mismo dominio. Por lo tanto la función objetivo  $F(X)$  puede ser reemplazada por  $G(X)$ , que consiste de las siguientes sub-funciones:

$$G(x) = [g_1(x), g_2(x), g_3(x), g_4(x), g_5(x), g_6(x), g_7(x), g_8(x)]$$

donde :

$$\begin{aligned} g_1(x) &= \sum_{i \in I} \sum_{j \in J} (s_{ij}^1 + s_{ij}^2) \\ g_2(x) &= \sum_{i \in I} \sum_{j=2}^{34} s_{ij}^3 \\ g_3(x) &= \sum_{i \in I} \sum_{j=2}^{34} s_{ij}^4 \\ g_4(x) &= \sum_{i \in I} \sum_{j=1}^{32} \sum_{k \in \{1,3\}} s_{ijk}^5 \\ g_5(x) &= \sum_{i \in I} \sum_{j=2}^{34} \sum_{k \in \{1,3\}} s_{ijk}^6 \\ g_6(x) &= \sum_{t=1}^3 \sum_{i \in I_t} \sum_{w=1}^5 (s_{iiv}^7 + s_{iiv}^8) \\ g_7(x) &= \sum_{i \in I_1} \sum_{r=1}^{32} s_{ir}^9 \\ g_8(x) &= \sum_{i \in I} \sum_{j=1}^{34} \sum_{k' \in K'} s_{ijk'}^{10} \end{aligned}$$

Este es un problema multiobjetivo y su objetivo es hallar o aproximar el conjunto de

soluciones óptimas de Pareto. El tradicional acercamiento a este tipo de problema es la suma ponderada que combina múltiples objetivos en un solo objetivo escalar.

### 2.3.5. Descripción del problema de Lü y Hao (2012)

EL NRP considerado en este trabajo consiste en asignar turnos a enfermeras dado un conjunto de restricciones. El objetivo es minimizar la suma ponderada total de las violaciones de las restricciones suaves, satisfaciendo todas las restricciones fuertes.

#### Conjuntos

- $D$  : Conjunto de días durante los cuales las enfermeras son programadas e indexado por  $d \in \{1, \dots, |D|\}$ .
- $S$  : Conjunto de enfermeras, siendo cada una asociada con un conjunto de habilidades y trabajando en un contrato e indexado por  $s \in \{1, \dots, |S|\}$ .
- $H$  : Conjunto de turnos, cada uno siendo caracterizado por un conjunto de habilidades requeridas e indexado por  $h \in \{1, \dots, |H|\}$ .
- $P$  : Conjunto de patrones de turnos que la enfermera no desea trabajar seguidos.
- $C$  : Conjunto de contratos, cada uno siendo caracterizado por un número de regulaciones que deberían ser respetadas por las enfermeras.

Se representa un candidato a solución por una matriz  $|S| \times |D|$  llamada  $(x_{ij})$ , donde  $x_{ij}$  corresponde al tipo de turno asignado a la enfermera  $s_i$  en el día  $d_j$ . Si no hay turno asignado entonces  $x_{ij} = -1$ . Con esta representación se asegura que una enfermera trabaje a lo más un turno por día.

#### Parámetros

- $sc(d, h)$  : Número total de enfermeras requeridas para el día  $d$  en el turno  $h$ .
- $shift(s)^{+,-}$  : Número máximo/mínimo de turnos que pueden asignarse a enfermera  $s$ .
- $work(s)^{+,-}$  : Número máximo/mínimo de días consecutivos de trabajo de la enfermera  $s$ .

- $free(s)^{+,-}$  : Número máximo/mínimo de días consecutivos libres de la enfermera  $s$ .
- $Night$  : Turno de noche.
- $wkd(s)^{+,-}$  : Número máximo/mínimo de fines de semana consecutivos de trabajo de la enfermera  $s$ .
- $nwkd$  : Número total de fines de semana.
- $nwkd(s)^+$  : Número máximo de fines de semana de trabajo de la enfermera  $s$ .
- $nd(s)$  : Número total de días para cada fin de semana de la enfermera  $s$ , que pueden ser 2 o 3.
- $dayreq(s, d)$  : On(off) si enfermera  $s$  pide o (no) trabajar el día  $d$ .  
Es nulo de otro modo.
- $shreq(s, d, h)$  : On(off) si enfermera  $s$  pide o (no) trabajar el turno  $h$  el día  $d$ .  
Es nulo de otro modo.
- $qual(s, h)$  : Verdadero si la enfermera  $s$  tiene todas las habilidades requeridas de turno  $h$ .  
Es falso de otro modo.
- $unwantp(s)$  : El conjunto de patrones no deseados de la enfermera  $s$ .
- $nwkdsect(s)$  : Número de secciones de trabajo, esto es series consecutivas de días trabajando de la enfermera  $s$ .
- $lenwkdsect(s, i)$  : Largo de la  $i$ -ésima sección de trabajo de la enfermera  $s$ .
- $nfrsect(s)$  : Número de secciones libres, esto es series de días libres consecutivos.
- $lenfrsect(s, i)$  : Largo de la  $i$ -ésima seccion libre de la enfermera  $s$ .
- $nwkdsect(s)$  : Número de secciones de trabajo de fines de semana de la enfermera  $s$ , esto es una series de fines de semana de trabajo consecutivo.
- $lenwkdsect(s, i))$  : Largo de la  $i$ -ésima sección de trabajo de fin de semana de la enfermera  $s$ .

- $hwkd(s, i, j)$  : Tipo de turno asignado en el día  $j$  del  $i$ -ésimo fin de semana de la enfermera  $s$ .
- $nwkd(s, i)$  : Número de días de trabajo de la enfermera  $s$  en el  $i$ -ésimo fin de semana.
- $nh(s, i, h)$  : Número de tipos de turno  $h$  de la enfermera  $s$  en el  $i$ -ésimo fin de semana.
- $nunwp(s, p)$  : Número total de patrones del tipo  $p$  para enfermera  $s$ .

### Restricciones fuertes

Se utiliza la función indicatriz  $\chi$  que toma el valor 1 cuando la proposición es verdadera y toma el valor 0 cuando la proposición es falsa.

- H1 Cobertura de la demanda: Para cada día todos los turnos demandados deben ser asignados a enfermeras.
- $$\sum_{s=1}^{|S|} \chi(x_{sd} = h) = sc(d, h)$$
- H2 Un único turno por día: Una enfermera puede trabajar un solo turno por día. Esta restricción fuerte es siempre satisfecha, dada la representación de la variable  $x_{ij}$ .

### Restricciones débiles

- S1 Máxima programación: El número máximo de turnos que pueden ser asignados a  $s$ .
- $$f_{s,1} = \max \{ \sum_{d \in D} \chi(x_{s,d} \neq -1) - shift(s)^+, 0 \}$$
- S2 Mínima programación: El número mínimo de turnos que pueden ser asignados a  $s$ .
- $$f_{s,2} = \max \{ shift(s)^- - \sum_{d \in D} \chi(x_{s,d} \neq -1), 0 \}$$
- S3 Días máximos consecutivos de trabajo: El número máximo de días consecutivos que un turno ha sido asignado a  $s$ .
- $$f_{s,3} = \sum_{i=1}^{nwkdsect(s)} \max \{ lenwkdsect(s, i) - work(s)^+, 0 \}$$

- S4 Días mínimos consecutivos de trabajo: El número mínimo de días consecutivos que un turno ha sido asignado a  $s$ .

$$f_{s,4} = \sum_{i=1}^{n_{wkdsect}(s)} \text{máx} \{work(s)^- - len_{wkdsect}(s, i), 0\}$$

- S5 Máximos días consecutivos libres: El número máximo de días consecutivos en que  $s$  no tiene turno asignado.

$$f_{s,5} = \sum_{i=1}^{n_{frsect}(s)} \text{máx} \{len_{frsect}(s, i) - free(s)^+, 0\}$$

- S6 Mínimos días consecutivos libres: El número máximo de días consecutivos en que  $s$  no tiene turno asignado.

$$f_{s,6} = \sum_{i=1}^{n_{frsect}(s)} \text{máx} \{free(s)^- - len_{frsect}(s, i), 0\}$$

- S7 Dos días libres despues de un turno de noche: Ningún turno excepto un turno de noche debe ser asignado a  $s$  los dos días siguientes de un turno de noche.

$$f_{s,7} = \sum_{i=1}^{D-2} \chi(a_i) + \chi(x_{s,D-1} = Night \wedge x_{s,D} \neq Night \wedge x_{s,D} \neq -1)$$

donde

$$a_i = (x_{s,i} = Night \wedge ((x_{s,i+1} \neq Night \wedge x_{s,i+1} \neq -1) \vee (x_{s,i+2} \neq Night \wedge x_{s,i+2} \neq -1)))$$

- S8 Máximo fines de semana consecutivos trabajando: El número máximo de fines de semana consecutivos en que al menos un turno es asignado a  $s$ .

$$f_{s,8} = \sum_{i=1}^{n_{wkdsect}(s)} \text{máx} \{len_{wkdsect}(s, i) - wkd(s)^+, 0\}$$

- S9 Mínimo fines de semana consecutivos trabajando: El número mínimo de fines de semana consecutivos en que al menos un turno es asignado a  $s$ .

$$f_{s,9} = \sum_{i=1}^{n_{wkdsect}(s)} \text{máx} \{wkd(s)^- - len_{wkdsect}(s, i), 0\}$$

- S10 Número máximo de fines de semana trabajando: El número máximo de fines de semana en un periodo de cuatro semanas en que al menos un turno es asignado a  $s$ .

$$f_{s,10} = \text{máx} \left\{ \sum_{i=1}^{n_{wkd}} \chi(nwd(s, i) > 0) - n_{wkd}(s)^+, 0 \right\}$$

- S11 Fines de semana completos:  $s$  debe trabajar en todos los días de un fin de semana si al menos trabaja en un día del fin de semana.

$$f_{s,11} = \sum_{i=1}^{n_{wkd}} Comp_{wkdcost}(s, i)$$

donde

$$Compwkdcost(s, i) = \begin{cases} 4 & nd(s) = 3 \wedge nwd(s, i) = 2 \wedge hwkd(s, i, 2) = -1 \\ nd(s - nwd(s, i)) & 0 < nwd(s, i) < nd(s) \\ 0 & \sim \end{cases}$$

S12 Turnos idénticos en un fin de semana complete:  $s$  debe trabajar el mismo tipo de turno en los días de un fin de semana completo trabajando.

$$f_{s,12} = \sum_{i=1}^{nwk} Identwkdcost(s, i)$$

donde

$$Identwkdcost(s, i) = \begin{cases} \sum_{h \in H, nh(s, i, h) > 0} (nd(s) - nh(s, i, h)) & nwd(s, i) = nd(s) \\ 0 & \sim \end{cases}$$

S13 Solicitud de trabajar un día:  $s$  solicita trabajar un día específico.

$$f_{s,13} = \sum_{d \in D} \chi(dayreq(s, d) = on \wedge x_{s,d} = -1)$$

S14 Solicitud de día libre:  $s$  solicita no trabajar un día específico.

$$f_{s,14} = \sum_{d \in D} \chi(dayreq(s, d) = off \wedge x_{s,d} \neq -1)$$

S15 Solicitud de trabajar un turno:  $s$  solicita trabajar un turno y día específico.

$$f_{s,15} = \sum_{d \in D} \sum_{h \in H} \chi(shreq(s, d, h) = on \wedge x_{s,d} \neq h)$$

S16 Solicitud de no trabajar un turno:  $s$  solicita no trabajar un turno y día específico.

$$f_{s,16} = \sum_{d \in D} \sum_{h \in H} \chi(shreq(s, d, h) = off \wedge x_{s,d} = h)$$

S17 Habilidad alternativa:  $s$  debe trabajar un turno donde posea todas las características requeridas por ese turno.

$$f_{s,17} = \sum_{d \in D} \chi(x_{s,d} \neq -1 \wedge qual(s, x_{s,d}) = false)$$

S18 Patrones de turnos no deseados:  $s$  no debe trabajar un patron de turno específico.

$$f_{s,18} = \sum_{p \in unwnp(s)} nunwp(s, p)$$

### Función objetivo

$$\min \sum_{s \in S} \sum_{i=1}^{18} w_{s,i} \cdot f_{s,i}$$

donde  $w_{s,i}$  es el peso asociado a la restricción débil  $S_i$ , regulado por el contrato de la enfermera  $s$ .



## 3 | Formulación del modelo

Para el desarrollo de esta memoria investigamos un NRP con restricciones propias de un hospital en Nueva Zelanda, considerando un horizonte de programación de 28 días, tres turnos diarios y tres categorías de enfermeras. Formulamos el problema como un Integer Linear Problem (ILP), específicamente un modelo multi objetivo que utiliza el método de los pesos para determinar la óptima asignación de enfermeras que minimiza la insatisfacción de los requerimientos y la cantidad de enfermeras cuyos requerimientos no se satisfacen completamente. La factibilidad del problema radica en tener una cantidad apropiada (y mínima) de enfermeras dispuestas a trabajar en turnos de noche, las cuales a petición del manager pueden cubrir a lo más un turno de día cada dos semanas.

### Model framework

Un hospital cuenta con  $N$ ,  $i = 1, \dots, |N|$  enfermeras que se dividen en  $K$ ,  $k = 1, \dots, |K|$  clases, donde  $k = 1$  son Enfermeras Registradas,  $k = 2$  son Enfermeras Enroladas y  $k = 3$  son Health Care Assistant. Sea  $N_k = \{i \in N : i \text{ es de clase } k\}$ . Cada enfermera  $i$  tiene un contrato por  $c_i$  turnos, válido por dos semanas. Se necesita asignar los  $S$ ,  $s = 1, 2, 3$  turnos diarios del hospital en un horizonte de programación  $T$ ,  $t = 1, \dots, |T|$ , de modo de satisfacer la demanda de enfermeras  $d_{st}$  de cada día  $t \in T$  y cada turno  $s \in S$ . Cada enfermera  $i$  puede decidir trabajar  $o_i$  turnos extra durante las dos primeras semanas y  $u_i$  turnos extras durante las dos segundas semanas del periodo de programación. También se debe considerar que ninguna enfermera puede trabajar dos turnos seguidos y debe tener dos turnos de descanso entre asignaciones. Cada enfermera debe tener dos días consecutivos libres, luego de trabajar un turno de noche. Toda enfermera luego de 7 días de turnos consecutivos debe tener 48 horas libres. En cada turno  $s$  deben haber mínimo  $\underline{M}_{ks}$  enfermeras de categoría  $k$  y máximo  $\overline{M}_{ks}$  enfermeras de categoría  $k$ . Además hay otra clasificación: las enfermeras

de día  $J_1$  y las enfermeras de noche  $J_2$ . Las enfermeras de día pueden realizar a lo más un turno de noche cada dos semanas y las enfermeras de noche pueden realizar a lo más un turno de día cada dos semanas. Existe un parámetro que representa los requerimientos de las enfermeras,  $q_{ist} = 1$ , cuando enfermera  $i$  solicita no trabajar el turno  $s$  del día  $t$  y  $q_{ist} = 0$  en otro caso.

### Extended model formulation

#### Conjuntos

- $S$  : Conjunto de turnos diarios, indexado por  $s \in \{1, 2, 3\}$   
 $N$  : Conjunto de enfermeras, indexado por  $i \in \{1, \dots, |N|\}$   
 $K$  : Conjunto de categorías de enfermera indexado por  $k \in \{1, \dots, |K|\}$   
 $N_k$  : Conjunto de enfermeras de clase  $k$   
 $J_1$  : Conjunto de enfermeras de día  
 $J_2$  : Conjunto de enfermeras de noche  
 $T$  : Conjunto de días del horizonte de programación, indexado por  $t \in \{1, \dots, |T|\}$

#### Parámetros

- $d_{st}$  : Demanda de enfermeras del turno  $s$  el día  $t$   
 $c_i$  : Número de turnos contratados a la enfermera  $i$  cada dos semanas  
 $o_i$  : Número de turnos extras que la enfermera  $i$  está dispuesta a trabajar durante las dos primeras semanas del horizonte de programación  $T$   
 $u_i$  : Número de turnos extras que la enfermera  $i$  está dispuesta a trabajar durante las dos últimas semanas del horizonte de programación  $T$   
 $\overline{M}_{ks}$  : Número máximo de enfermeras clase  $k$  en turno  $s$   
 $\underline{M}_{ks}$  : Número mínimo de enfermeras clase  $k$  en turno  $s$   
 $q_{ist} = \begin{cases} 1 & \text{si } i \text{ solicita no trabajar turno } s \text{ el día } t \\ 0 & \sim \end{cases}$

VARIABLES

$$r_{ist} = \begin{cases} 1 & \text{si enfermera } i \text{ trabaja turno } s \text{ el día } t \\ 0 & \sim \end{cases}$$

$$x_i = \begin{cases} 1 & \text{si enfermera } i \text{ no está completamente satisfecha con sus requerimientos} \\ 0 & \sim \end{cases}$$

RESTRICCIONES

*Restricción 1:* Satisfacción de demanda del turno  $s$  del día  $t$ .

$$\sum_{i \in N} r_{ist} \geq d_{st} \quad \forall s \in S, t \in T$$

*Restricción 2:* Cantidad de turnos asignados a enfermera debe ser menor que la cantidad de turnos contractuales.

$$\sum_{s \in S} \sum_{t=1}^{14} r_{ist} \leq c_i + o_i \quad \forall i \in N$$

$$\sum_{s \in S} \sum_{t=15}^{28} r_{ist} \leq c_i + u_i \quad \forall i \in N$$

*Restricción 3:* Ninguna enfermera puede hacer dos turnos seguidos.

$$r_{ist} + r_{i,t}^{s+1} \leq 1 \quad \forall i \in N, t \in T, s \in S, s \neq |S|$$

$$r_{i|S|t} + r_{i1}^{t+1} \leq 1 \quad \forall i \in N, t \in T, t \neq |T|$$

*Restricción 4:* Dos turnos de descanso entre asignaciones para cada enfermera i.e; si asigna el primer turno no puede asignar el tercer turno, si asigna el segundo turno no puede asignar el primer turno del día siguiente, si asigna el tercero no puede asignar el segundo del día siguiente.

$$r_{i1t} + r_{i3t} \leq 1 \quad \forall i \in N, t \in T$$

$$r_{i2t} + r_{i1}^{t+1} \leq 1 \quad \forall i \in N, t \in T, t \neq |T|$$

$$r_{i3t} + r_{i2}^{t+1} \leq 1 \quad \forall i \in N, t \in T, t \neq |T|$$

*Restricción 5:* Las enfermeras de día pueden realizar a lo más un turno de noche cada dos semanas.

$$\sum_{t=1}^{14} r_{i1t} \leq 1 \quad \forall i \in J_1$$

$$\sum_{t=15}^{28} r_{i1t} \leq 1 \quad \forall i \in J_1$$

*Restricción 6:* Las enfermeras de noche pueden realizar a lo más un turno de día cada dos semanas.

$$\sum_{s=2,3} \sum_{t=1}^{14} r_{ist} \leq 1 \quad \forall i \in J_2$$

$$\sum_{s=2,3} \sum_{t=15}^{28} r_{ist} \leq 1 \quad \forall i \in J_2$$

*Restricción 7:* Cada enfermera debe tener dos días consecutivos libres luego de una noche.

$$r_{i1t} = 1 \quad \Rightarrow \quad r_{i1}^{t+1} + r_{i1}^{t+2} + \sum_{s \neq 1} (r_{ist} + r_{is}^{t+1}) \leq 0 \quad \forall i \in N, t \in T, t \neq |T|, |T| - 1$$

equivalentemente

$$r_{i1}^{t+1} + r_{i1}^{t+2} + \sum_{s \neq 1} (r_{ist} + r_{is}^{t+1}) \leq (1 - r_{i1t}) \cdot M_1 \quad \forall i \in N, t \in T, t \neq |T|, |T| - 1, M_1 = 6$$

*Restricción 8:* Cada enfermera luego de 7 días de trabajo consecutivo debe tener 48 horas libres.

Dado que las restricciones 3 y 4 implican que sólo se trabaja un turno diario, entonces se cuenta la cantidad de turnos trabajados como los días trabajados:

$$\sum_{s \in S} \sum_{j=t}^{t+6} r_{isj} \geq 7 \quad \Rightarrow \quad r_{is}^{t+7} + r_{is}^{t+8} \leq 0 \quad \forall i \in N, s \in S, t \in T - \{20, \dots, 28\}$$

equivalentemente

$$\sum_{s \in S} \sum_{j=t}^{t+6} r_{isj} \geq 7 \quad \Rightarrow \quad r_{is}^{t+7} = 0 \wedge r_{is}^{t+8} = 0 \quad \forall i \in N, s \in S, t \in T - \{20, \dots, 28\}$$

equivalentemente

$$r_{is}^{t+7} = 1 \vee r_{is}^{t+8} = 1 \quad \Rightarrow \quad \sum_{s \in S} \sum_{j=t}^{t+6} r_{isj} \leq 6 \quad \forall i \in N, s \in S, t \in T - \{20, \dots, 28\}$$

equivalentemente

$$\sum_{s \in S} \sum_{j=t}^{t+6} r_{isj} \leq 6 + (1 - r_{is}^{t+7}) \cdot M_2 \quad \forall i \in N, s \in S, t \in T - \{21, \dots, 28\}, M_2 = 15$$

$$\sum_{s \in S} \sum_{j=t}^{t+6} r_{isj} \leq 6 + (1 - r_{is}^{t+8}) \cdot M_3 \quad \forall i \in N, s \in S, t \in T - \{21, \dots, 28\}, M_3 = 15$$

*Restricción 9:* Máximo  $\overline{M}_{ks}$  enfermeras categoría  $k$  en turno  $s$ .

$$\sum_{i \in N_k} r_{ist} \leq \overline{M}_{ks} \quad \forall k \in K, t \in T, s \in S$$

*Restricción 10:* Mínimo  $\underline{M}_{ks}$  enfermeras categoría  $k$  en turno  $s$ .

$$\sum_{i \in N_k} r_{ist} \geq \underline{M}_{ks} \quad \forall k \in K, t \in T, s \in S$$

*Restricción 11:* Si la enfermera  $i$  está completamente satisfecha con sus requerimientos, entonces  $x_i = 0$ .

$$\sum_{s \in S, t \in T} r_{ist} \cdot q_{ist} \geq x_i \quad \forall i \in N$$

*Restricción 12:* Si la enfermera  $i$  no está completamente satisfecha con sus requerimientos, entonces  $x_i = 1$ .

$$\sum_{s \in S, t \in T} r_{ist} \cdot q_{ist} \geq M_4 \cdot x_i \quad \forall i \in N, M_4 = 84$$

*Restricción 13:* Naturaleza de las variables.

$$r_{ist} \in \{0, 1\} \quad \forall i \in N, s \in S, t \in T$$

$$x_i \in \{0, 1\} \quad \forall i \in N$$

Función Objetivo

$$\min_r \quad \alpha \sum_{i \in N} \sum_{s \in S} \sum_{t \in T} q_{ist} \cdot r_{ist} + (1 - \alpha) \sum_{i \in N} x_i$$

Compact model

$$\min_r \quad \alpha \sum_{i \in N} \sum_{s \in S} \sum_{t \in T} q_{ist} \cdot r_{ist} + (1 - \alpha) \sum_{i \in N} x_i \quad (1.1)$$

$$\text{s.t.} \quad \sum_{i \in N} r_{ist} \geq d_{st} \quad \forall s \in S, t \in T \quad (1.2)$$

$$\sum_{s \in S} \sum_{t=1}^{14} r_{ist} \leq c_i + o_i \quad \forall i \in N \quad (1.3)$$

$$\sum_{s \in S} \sum_{t=15}^{28} r_{ist} \leq c_i + u_i \quad \forall i \in N \quad (1.4)$$

$$r_{ist} + r_{it}^{s+1} \leq 1 \quad \forall i \in N, t \in T, s \in S, s \neq |S| \quad (1.5)$$

$$r_{i|S|t} + r_{i1}^{t+1} \leq 1 \quad \forall i \in N, t \in T, t \neq |T| \quad (1.6)$$

$$r_{i1t} + r_{i3t} \leq 1 \quad \forall i \in N, t \in T \quad (1.7)$$

$$r_{i2t} + r_{i1}^{t+1} \leq 1 \quad \forall i \in N, t \in T, t \neq |T| \quad (1.8)$$

$$r_{i3t} + r_{i2}^{t+1} \leq 1 \quad \forall i \in N, t \in T, t \neq |T| \quad (1.9)$$

$$\sum_{t=1}^{14} r_{i1t} \leq 1 \quad \forall i \in J_1 \quad (1.10)$$

$$\sum_{t=15}^{28} r_{i1t} \leq 1 \quad \forall i \in J_1 \quad (1.11)$$

$$\sum_{s=2,3} \sum_{t=1}^{14} r_{ist} \leq 1 \quad \forall i \in J_2 \quad (1.12)$$

$$\sum_{s=2,3} \sum_{t=15}^{28} r_{ist} \leq 1 \quad \forall i \in J_2 \quad (1.13)$$

$$r_{i1}^{t+1} + r_{i1}^{t+2} + \sum_{s \neq 1} (r_{ist} + r_{is}^{t+1}) \leq (1 - r_{i1t}) \cdot 6 \quad \forall i \in N, t \in T, t \neq |T|, |T| - 1 \quad (1.14)$$

$$\sum_{s \in S} \sum_{j=t}^{t+6} r_{isj} \leq 6 + (1 - r_{is}^{t+7}) \cdot 15 \quad \forall i \in N, s \in S, t \in T - \{21, \dots, 28\} \quad (1.15)$$

$$\sum_{s \in S} \sum_{j=t}^{t+6} r_{isj} \leq 6 + (1 - r_{is}^{t+8}) \cdot 15 \quad \forall i \in N, s \in S, t \in T - \{21, \dots, 28\} \quad (1.16)$$

$$\sum_{i \in N_k} r_{ist} \leq \overline{M_{ks}} \quad \forall k \in K, t \in T, s \in S \quad (1.17)$$

$$\sum_{i \in N_k} r_{ist} \geq \underline{M_{ks}} \quad \forall k \in K, t \in T, s \in S \quad (1.18)$$

$$\sum_{s \in S, t \in T} r_{ist} \cdot q_{ist} \geq x_i \quad \forall i \in N \quad (1.19)$$

$$\sum_{s \in S, t \in T} r_{ist} \cdot q_{ist} \geq 84 \cdot x_i \quad \forall i \in N \quad (1.20)$$

$$r_{ist} \in \{0, 1\} \quad \forall i \in N, s \in S, t \in T \quad (1.21)$$

$$x_i \in \{0, 1\} \quad \forall i \in N \quad (1.22)$$



## 4 | Estudio computacional

### 4.1. Análisis de resultados

El estudio computacional fue desarrollado con dos objetivos, (i) evaluar el desempeño de la solución respecto a la satisfacción del staff y los CPU time; y (ii) realizar un análisis de sensibilidad de las soluciones respecto del peso  $\alpha$  de la función objetivo.

El modelo fue resuelto en un procesador Intel(R) Core(TM) i3-2350M 2,30 GHz, memoria 4 GB, utilizando el sistema operativo Windows 7 y el solver CPLEXAMP.

En el estudio computacional de la presente memoria se utilizó el siguiente set de valores como parámetros del modelo.

- Horizonte de programación:  $T = 28$
- Número total de enfermeras:  $N = 40$ .
- Número de enfermeras categoría 1:  $N_1 = 30$ .
- Número de enfermeras categoría 2:  $N_2 = 2$ .
- Número de enfermeras categoría 3:  $N_3 = 8$ .
- Número de enfermeras de día:  $J_1 = 31$ .
- Número de enfermeras de noche:  $J_2 = 8$ .
- Demanda de enfermeras turno 1 día  $t$ :  $d_{1t} = 5, \forall t \in T$ .
- Demanda de enfermeras turno 2 día  $t$ :  $d_{2t} = 11, \forall t \in T$ .

|       |    |   |   |    |    |   |    |    |   |    |    |    |    |    |    |    |    |    |    |    |
|-------|----|---|---|----|----|---|----|----|---|----|----|----|----|----|----|----|----|----|----|----|
| $i$   | 1  | 2 | 3 | 4  | 5  | 6 | 7  | 8  | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| $c_i$ | 10 | 8 | 8 | 10 | 10 | 8 | 10 | 10 | 8 | 8  | 8  | 10 | 10 | 10 | 10 | 8  | 8  | 8  | 8  | 10 |
| $o_i$ | 0  | 1 | 0 | 0  | 0  | 0 | 1  | 0  | 0 | 0  | 0  | 1  | 0  | 0  | 1  | 0  | 1  | 0  | 0  | 0  |
| $u_i$ | 1  | 0 | 1 | 0  | 1  | 0 | 0  | 0  | 1 | 0  | 0  | 0  | 0  | 1  | 0  | 0  | 0  | 1  | 1  | 0  |

**Tabla 4.1:** Condiciones de contrato  $i = 1, \dots, 20$ 

|       |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
|-------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| $i$   | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 |
| $c_i$ | 8  | 8  | 10 | 8  | 10 | 10 | 10 | 8  | 8  | 8  | 8  | 8  | 10 | 10 | 8  | 8  | 10 | 8  | 8  | 10 |
| $o_i$ | 0  | 0  | 0  | 0  | 1  | 0  | 0  | 1  | 0  | 1  | 1  | 0  | 1  | 1  | 0  | 0  | 0  | 1  | 0  | 0  |
| $u_i$ | 1  | 0  | 0  | 0  | 0  | 0  | 0  | 1  | 0  | 1  | 1  | 0  | 1  | 0  | 1  | 0  | 0  | 0  | 0  | 0  |

**Tabla 4.2:** Condiciones de contrato  $i = 21, \dots, 40$ 

- Demanda de enfermeras turno 3 día  $t$ :  $d_{3t} = 8, \forall t \in T$ .
- Número de turnos contratados a la enfermera  $i$ . Las tablas 4.1 y 4.2 ilustran cada ítem  $c_i$ .
- Número de turnos extras que la enfermera  $i$  está dispuesta a trabajar durante las dos primeras semanas del horizonte de programación. Las tablas 4.1 y 4.2 ilustran cada ítem  $o_i$ .
- Número de turnos extras que la enfermera  $i$  está dispuesta a trabajar durante las dos últimas semanas del horizonte de programación. Las tablas 4.1 y 4.2 ilustran cada ítem  $u_i$ .
- Número máximo de enfermeras clase  $k$  en turno  $s$ . La tabla 4.3 ilustra cada ítem  $\bar{M}_{ks}$ .
- Número mínimo de enfermeras clase  $k$  en turno  $s$ . La tabla 4.4 ilustra cada ítem  $\underline{M}_{ks}$ .

|                  |   |   |   |
|------------------|---|---|---|
| $k \backslash s$ | 1 | 2 | 3 |
| 1                | 0 | 0 | 0 |
| 2                | 0 | 0 | 0 |
| 3                | 1 | 3 | 1 |

**Tabla 4.3:**  $\bar{M}_{ks}$ 

|                  |   |   |   |
|------------------|---|---|---|
| $k \backslash s$ | 1 | 2 | 3 |
| 1                | 4 | 8 | 7 |
| 2                | 1 | 1 | 1 |
| 3                | 1 | 3 | 2 |

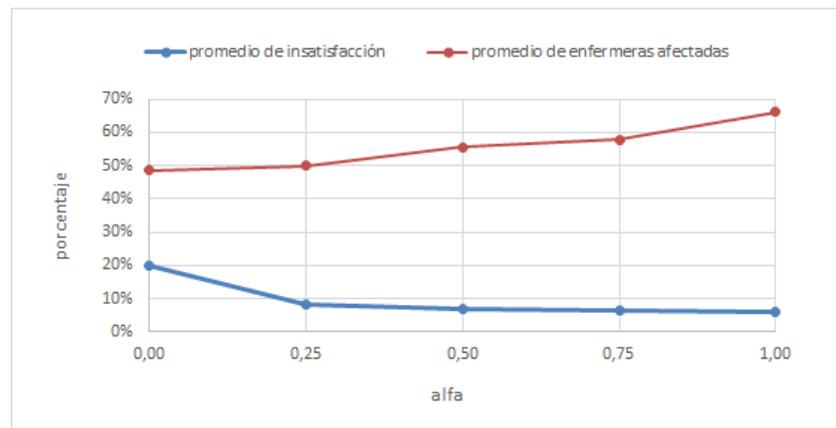
**Tabla 4.4:**  $\underline{M}_{ks}$ 

Generamos 50 problemas del siguiente modo: se escogieron aleatoriamente 10 instancias de requerimientos  $q_{ist}$  y cada uno de ellos fue optimizado con  $\alpha \in \{0; 0, 25; 0, 5; 0, 75; 1\}$ .

Cada problema arroja el número de enfermeras afectadas  $\sum_{i \in N} x_i$ , es decir el número de enfermeras para las cuales no se cumplió al menos uno de sus requisitos. Así cada  $\alpha$  tiene un promedio de enfermeras afectadas calculado sobre las diez instancias. Estos promedios varían entre 49 % y 66 % . Información resumida en la figura 4.1.

Por otro lado cada problema también entrega el porcentaje de insatisfacción de cada enfermera  $i$ , calculado como el cociente  $\frac{\sum_{s \in S, t \in T} r_{ist} \cdot q_{ist}}{\text{número de turnos libres solicitados}}$ .

Así cada problema tiene asociado un promedio de insatisfacción, calculado entre todas las enfermeras afectadas. Luego cada  $\alpha$  tiene un promedio de insatisfacción calculado entre las diez instancias, los cuales fluctúan entre 6 % y 20 % . Información resumida en la figura 4.1.

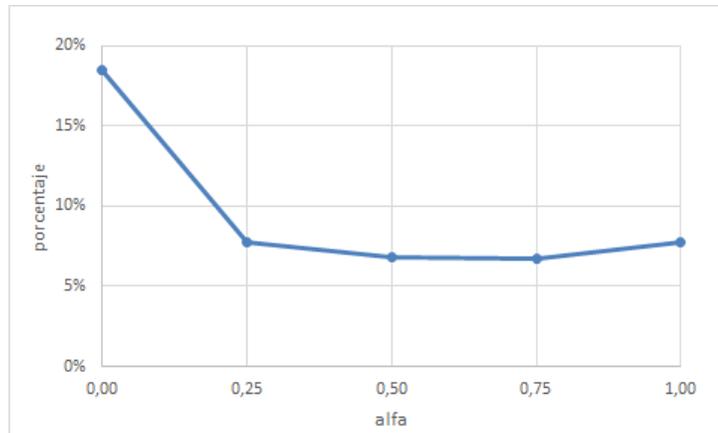


**Figura 4.1:** Promedio de insatisfacción y enfermeras afectadas.

En figura 4.1 a medida que  $\alpha$  aumenta se observa que el promedio de insatisfacción decrece, mientras que el número de enfermeras afectadas crece. Esta situación es esperable, considerando que en la función objetivo  $\alpha$  pondera al factor  $\sum_{s \in S, t \in T} r_{ist} \cdot q_{ist}$  asociado con la insatisfacción, mientras que  $1 - \alpha$  pondera al número de enfermeras afectadas  $\sum_{i \in N} x_i$ .

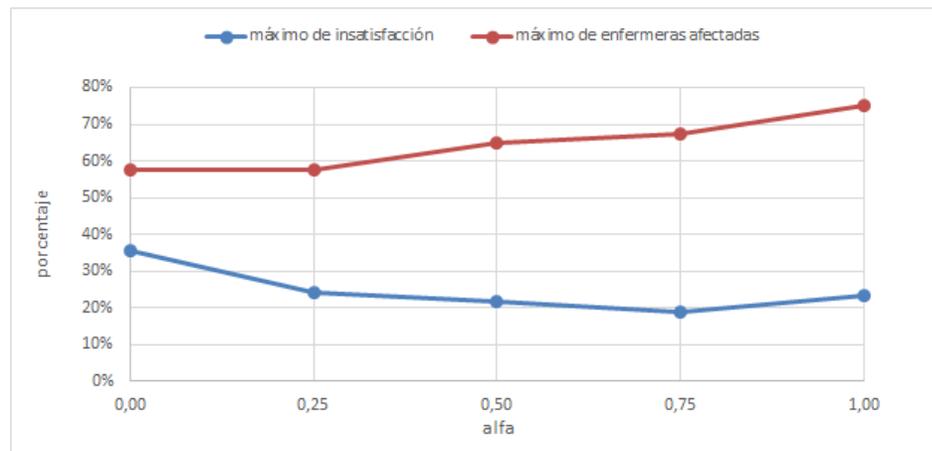
Sin embargo, analizando un instancia fijo, por ejemplo instancia 2 en la figura 4.2. Observamos que el promedio de insatisfacción no es estrictamente decreciente.

Por otro lado cada instancia tiene un máximo de insatisfacción calculado entre todas las enfermeras. Así cada  $\alpha$  tiene un máximo de insatisfacción calculado entre sus instancias,



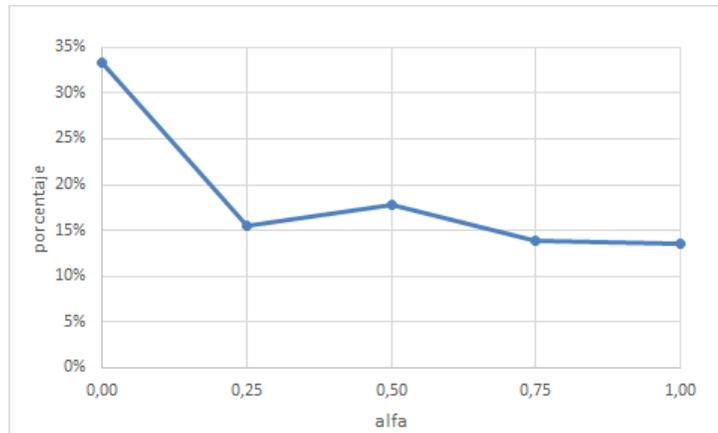
**Figura 4.2:** Promedio de insatisfacción instancia 2.

los cuales varían 23 % y 36 %. Análogamente cada instancia tiene un número de enfermeras afectadas. Así cada  $\alpha$  tiene un máximo de enfermeras afectadas, que se expresa como porcentaje dividiendo por el número total de enfermeras. Estos fluctúan entre 58 % y 75 %. Información resumida en figura 4.3.



**Figura 4.3:** Máximo de insatisfacción y enfermeras afectadas.

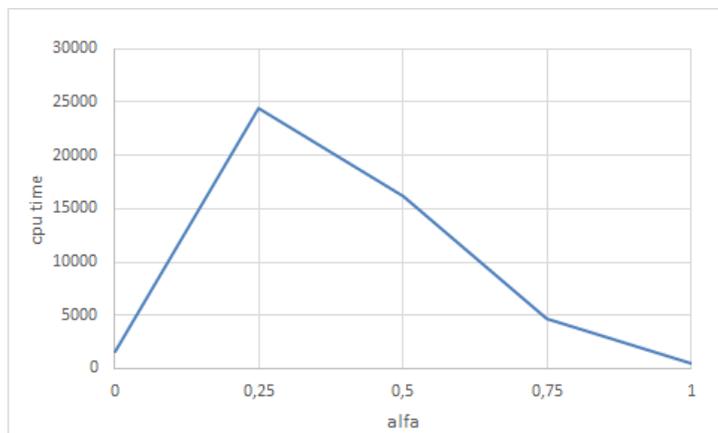
También de la figura 4.3 se observa que el máximo de enfermeras afectadas crece, aunque no estrictamente. Sin embargo el máximo de insatisfacción no presenta monotonía. De hecho analizando un instancia en particular, por ejemplo el instancia 4 en la figura 4.4, se observa el mismo fenómeno en la insatisfacción.



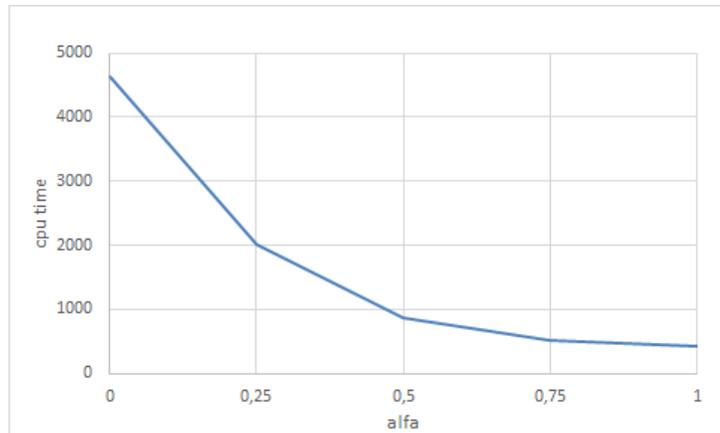
**Figura 4.4:** Máximo de insatisfacción instancia 4.

## 4.2. Análisis de tiempos computacionales

Para cada instancia se graficó CPU time v/s alfa. En el instancia 4 en la figura 4.5 se observa que los CPU time crecen y luego decrecen. Sin embargo en el instancia 10 en la figura 4.6, se puede verificar que es creciente.

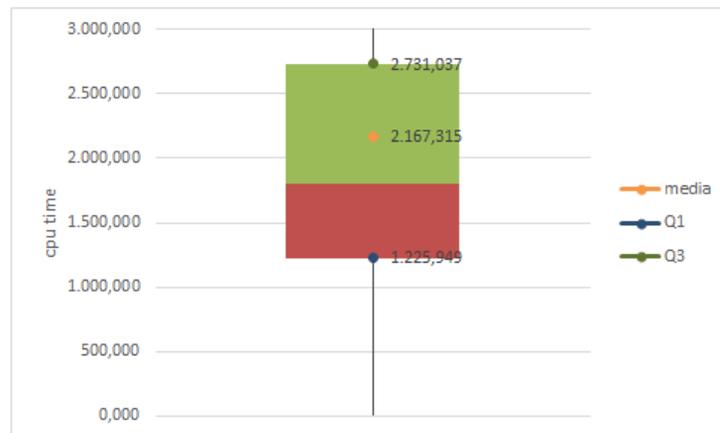


**Figura 4.5:** CPU time instancia 4.



**Figura 4.6:** CPU time instancia 10.

A continuación se presentan un análisis de los CPU time mediante box plot:



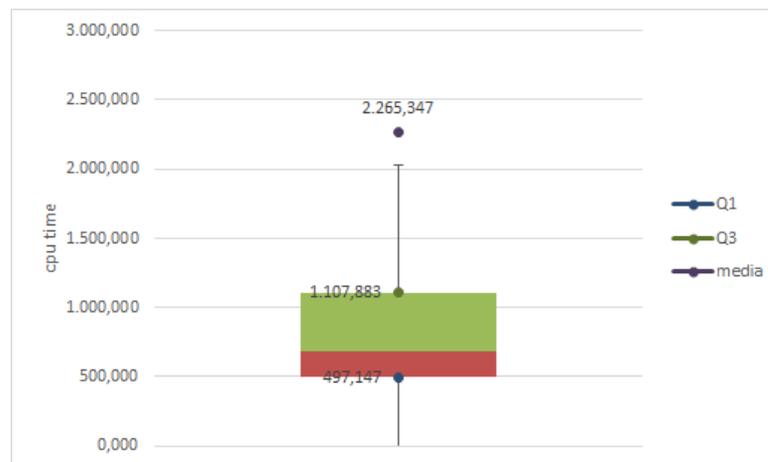
**Figura 4.7:** Box plot  $\alpha = 0$ .

Para  $\alpha = 0$  el 50 % de los experimentos poseen CPU time en el intervalo [1.225, 2.732] en torno a la media 2.167[s]. Ver figura 4.7.



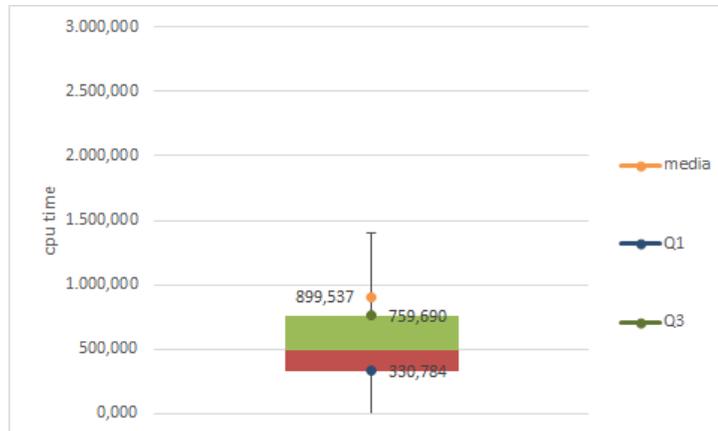
**Figura 4.8:** Box plot  $\alpha = 0,25$ .

Para  $\alpha = 0,25$  el 50 % de los experimentos poseen CPU time en el intervalo [746,2.902], en este caso la media 3.838[s] está fuera de este rango, lo cual se explica por el valor atípico que se presenta con 24.406[s]. Ver figura 4.8.



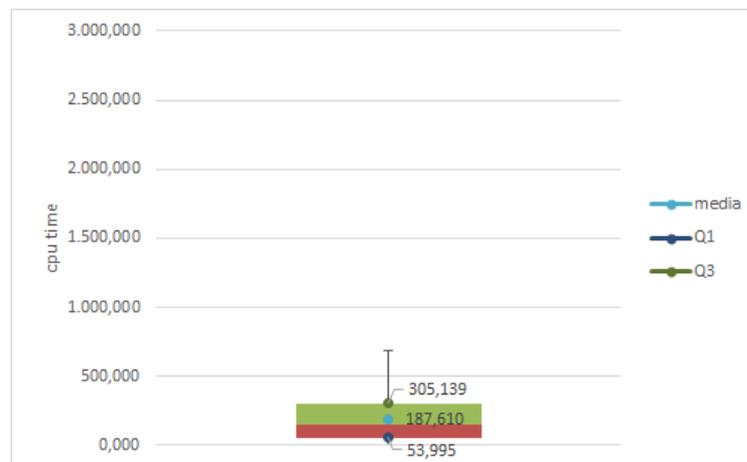
**Figura 4.9:** Box plot  $\alpha = 0,5$ .

Para  $\alpha = 0,5$  el 50 % de los experimentos poseen CPU time en el intervalo [497, 1.108], en este caso la media 2.265[s] está fuera de este rango, lo cual se explica por el valor atípico que se presenta con 16.225[s]. Ver figura 4.9.



**Figura 4.10:** Box plot  $\alpha = 0,75$ .

Para  $\alpha = 0,75$  el 50 % de los experimentos poseen CPU time en el intervalo [330; 760], en este caso la media 890[s] está fuera de este rango, lo cual se explica por el valor atípico que se presenta con 4.749[s]. Ver figura 4.10.



**Figura 4.11:** Box plot  $\alpha = 1$ .

Para  $\alpha = 1$  el 50 % de los experimentos poseen CPU time en el intervalo [53, 306], en torno a la media la media 188[s]. Ver figura 4.11.

Además se destacan los siguientes resultados:

- Para  $\alpha = 1$  la distribución es simétrica y a medida que  $\alpha$  es más pequeño es cada vez más sesgada a la derecha, es decir los CPU time tienden a concentrarse hacia la parte inferior de la distribución y la media es mayor que la mediana.

- La dispersión de los datos decrece considerablemente a partir de  $\alpha = 0,25$ , es decir los CPU time se concentran en un intervalo cada vez más pequeño.
- El máximo de los CPU time se presentó siempre en el instancia 4, salvo para  $\alpha = 0$ .



## 5 | Conclusiones y trabajo futuro

En esta memoria se aborda un nurse rostering problem de un hospital en Nueva Zelanda y se logra plantear todas las restricciones del roster como un problema ILP, cuya función es multi objetivo y utiliza el método de los pesos, de modo que minimiza la insatisfacción de los requerimientos de turnos libres por parte de las enfermeras y la cantidad de enfermeras cuyos requerimientos no se satisfacen completamente.

Para efectos del estudio computacional se consideraron diez instancias dados por los requerimientos de turnos libres elegidos de manera aleatoria. Cada uno de estos se resolvió utilizando CPLEXAMP con distintos pesos en la función objetivo  $\alpha = 0; 0, 25; 0, 5; 0, 75; 1$ , obteniendo cincuenta problemas.

Se estudió cómo afecta el parámetro de peso  $\alpha$  en el comportamiento de las componentes de la función objetivo: insatisfacción y número de enfermeras afectadas, ambos expresados como porcentajes, donde  $\alpha$  pondera la insatisfacción y  $1 - \alpha$  el número de enfermeras afectadas. Se concluye que cuando  $\alpha$  crece, el promedio de insatisfacción decrece entre de 20 % a 6 %, mientras que el promedio de enfermeras afectadas crece de 49 % a 66 %. Similarmente a medida que  $\alpha$  crece, el máximo de enfermeras afectadas crece de 58 % y 75 %. El máximo de insatisfacción no presenta un comportamiento monótono y varía entre 23 % y 36 %.

Se observó también que estos comportamientos de crecimiento y de decrecimiento respecto de  $\alpha$ , que se detectaron para máximos y promedios, no se aprecian en un instancia fijo, es decir no presentan monotonía.

Finalmente se obtuvieron los CPU time, resultando que en cada instancia estos no varían de manera monótona en función de  $\alpha$ . También para cada  $\alpha$  fijo, se hizo un box plot que distribuye los CPU time de los diez instancias, arrojando valores atípicos, salvo para

$\alpha = 0$  y  $\alpha = 1$  y también se concluye que para  $\alpha = 1$  la distribución es simétrica y a medida que  $\alpha$  es más pequeño es cada vez más sesgada a la derecha, sin embargo en  $\alpha = 0$  vuelve a ser simétrica. Además la dispersión de los datos decrece considerablemente a partir de  $\alpha = 0,25$ .

Las instancias analizadas permiten concluir que la formulación propuesta es tratable porque el tiempo promedio de uso de CPU fue 1.872 segundos y el peor caso fue de 24.406 segundos. En cada instancia el mínimo de la función objetivo se alcanza en  $\alpha = 0$ , de hecho la función objetivo es creciente en la variable  $\alpha$ .

Dentro de las extensiones a esta trabajo destacan:

(i) Trabajar con restricciones suaves, es decir permitir la no satisfacción de ciertas restricciones, asignando una penalización y luego minimizar estas penalizaciones. (ii) Incorporar el concepto de igualdad que distribuya de manera más justa los turnos menos populares, por ejemplo días feriados, vacaciones de invierno, verano. (iii) Investigar la continuidad de un horizonte de programación al siguiente. (iv) Medir la mejora de los resultados ante la incorporación de un self scheduling. (v) Incorporar staff que no trabaja en sistema de turnos, cuyo trabajo debe ser reasignado en días libres.

## Bibliografía

- Bellantini, F.; Carello, G.; Croce, F. Della; y Tadei, R. (2004). A greedy-based neighborhood search approach to a nurse rostering problem. *European Journal of Operational Research*, 153(1), 28 – 40. Timetabling and Rostering. [2.3.3.2](#)
- Burke, Edmund K.; De Causmaecker, Patrick; Petrovic, Sanja; y Vanden Berghe, Greet (2002). A multi criteria meta-heuristic approach to nurse rostering. *Proceedings of the 2002 Congress on Evolutionary Computation, CEC 2002*, 2, 1197–1202. [2.3.2.1](#)
- Burke, Edmund K.; De Causmaecker, Patrick; y Vanden Berghe, Greet (2004). The state of the art of nurse scheduling. *Journal of scheduling*, 7, 441–499. [1](#), [2](#), [2.3](#), [2.3.3.2](#)
- Burke, Edmund K.; Li, Jingpeng; y Qu, Rong (2010). A hybrid model of integer programming and variable neighbourhood search for highly-constrained nurse rostering problems. *European Journal of Operational Research*, 203(2), 484–493. ([document](#)), [2.3.3.2](#), [2.3.4](#)
- De Causmaecker, Patrick y Vanden Berghe, Greet (2011). A categorisation of nurse rostering problems. *Journal of Scheduling*, 14(1), 3–16. [2.3.3.2](#)
- Ernst, a. T.; Jiang, H.; Krishnamoorthy, M.; y Sier, D. (2004). Staff scheduling and rostering: A review of applications, methods and models. *European Journal of Operational Research*, 153(1), 3–27. [2.3.3.2](#)
- Glass, Celia a. y Knight, Roger a. (2010). The nurse rostering problem: A critical appraisal of the problem structure. *European Journal of Operational Research*, 202(2), 379–389. [2.3.3.2](#)
- Ikegami, A y Niwa, A (2003). A subproblem-centric model and approach to the nurse scheduling problem. *Mathematical Programming, Series B*, 97(3), 517–541. [2.3.3.1](#)
- Isken, Mark W. (2004). An Implicit Tour Scheduling Model with Applications in Healthcare. *Annals of Operations Research*, 128(1-4), 91–109. [2.3.1.1](#)
- Komarudin; Guerry, Marie-Anne; De Feyter, Tim; y Vanden Berghe, Greet (2013). The roster quality staffing problem – A methodology for improving the roster quality by modifying the personnel structure. *European Journal of Operational Research*, 230(3), 551–562. [2.3.3.2](#)
- Lü, Zhipeng y Hao, Jin-Kao (2012). Adaptive neighborhood search for nurse rostering. *European Journal of Operational Research*, 218(3), 865–876. ([document](#)), [2.3.3.2](#), [2.3.5](#)

- Moz, Margarida y Pato, MV (2003). Solving the problem of rostering nurse schedules with hard constraints: new multicommodity flow models. *Annals of Operations Research*, (1976), 285–301. [2.3.1.2](#)
- Oldenkamp, J.H. (1996). Quality in fives : on the analysis, operationalization and application of nursing schedule quality. [1](#)
- Osogami, T y Imai, H (2000). Classification of various neighborhood operations for the nurse scheduling problem. *Lecture Notes in Computer Science*, 1969, 72–83. [1](#)
- Özcan, Y. (2009). *Quantitative Methods in Health Care Management: Techniques and Applications*. [2.1](#), [2.1.2](#)
- Tassopoulos, Ioannis X.; Solos, Ioannis P.; y Beligiannis, Grigorios N. (2015). A two-phase adaptive variable neighborhood approach for nurse rostering. *Computers & Operations Research*, 60, 150–169. [1](#), [2.3.3.2](#)
- Warner, D. Michael (1976). Scheduling nursing personnel according to nursing preference: A mathematical programming approach. *Operations Research*, 24(5), 842. [2.1](#), [2.2.1](#)
- Welton, John M; Fischer, Mary Hughes; DeGrace, Sharon; y Zone-Smith, Laurie (2006). Hospital nursing costs, billing, and reimbursement. *Nursing Economics*, 24(5), 239. [1](#)