



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS

FACULTAD DE CONTADURÍA, ADMINISTRACIÓN E INFORMÁTICA
MAESTRÍA EN OPTIMIZACIÓN Y COMPUTO APLICADO

**Heurística de Calendarización para el Problema de
Asignación de Turnos para Enfermeras de la
Competencia Internacional 2014**

T E S I S

Que para obtener el Grado de Maestría en
Optimización y Computo Aplicado

Presenta

BORUNDA CASTILLO EIBERTH ADAIR

Director de Tesis

DR. FEDERICO ALONSO PECINA

Co-Director

DR. MARTÍN G. MARTÍNEZ RANGEL

Revisores:

Dr. José Crispín Zavala Díaz

Dra. Irma Yazmín Hernández Báez

Dr. Cruz Rosales Martín Heriberto

Dr. Federico Alonso Pecina

Dr. Gerardo Martín Martínez Rangel

CUERNAVACA, MORELOS

FEBRERO, 2021





Cuernavaca, Morelos a 9 de noviembre del 2020.

DR. AUGUSTO RENATO PÉREZ MAYO
SECRETARIO DE INVESTIGACIÓN Y POSGRADO DE LA FCAeI
PRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Computo Aplicado, del estudiante Borunda Castillo Eiberth Adair, con matricula 10023088, con el título **Heurística de Calendarización para el Problema de Asignación de Turnos para enfermeras de la Competencia Internacional 2014** por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación y Posgrado tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. Cruz Rosales Martín Heriberto
Profesor- investigador
Facultad de Contaduría, Administración e Informática



Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

MARTIN HERIBERTO CRUZ ROSALES | Fecha:2020-12-10 21:52:24 | Firmante
FCWE/WiaFFKq w2Mk gSU2x:Cn2FHbRRFb+ VwEeeJ6/EgyXs/e9iPchwK5znY oeicD6 gyIHhAB6awx2abv73Y17jCpzzQGbnN97q5n3aTot1rC/B Ir3n2D3jd6SDL4uD AokUTMm5VkdI
OF1fC7Rj9OjjozmR3me9bG9z9in0IZW7+6AH8T6TSE DHR4gnh y6Gjbm bXpD Wvx2FoA +chNk8LIE5DMsPBt/Mrb83Hrs aDQnK dWb9o Bdbgz z9fh4Mdq+ 8pSoc7zUMqoQe2KbD
ewodEejfWOpDqsoY dntXxhrdzT50pUFWfwsTir3a/p9it+ CHHCBOHZQksLJGUeQB gOFQkkQ==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o
escaneando el código QR ingresando la siguiente clave:



6h7pxv

<https://efirma.uaem.mx/foRepublio/nNx5Klr5xqHYSPYXorOI8oBJ8kgXH40>





Cuernavaca, Morelos a 9 de noviembre del 2020.

DR. AUGUSTO RENATO PÉREZ MAYO
SECRETARIO DE INVESTIGACIÓN Y POSGRADO DE LA FCAEI
PRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Computo Aplicado, del estudiante Borunda Castillo Eiberth Adair, con matricula 10023088, con el título **Heurística de Calendarización para el Problema de Asignación de Turnos para enfermeras de la Competencia Internacional 2014** por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación y Posgrado tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. Martín Gerardo Martínez Rangel
Profesor- investigador
Facultad de Contaduría, Administración e Informática

Av. Universidad 1001 Chamilpa Cuernavaca Morelos México C.P. 62209, Edificio 19
Tel. (777) 329 7917, Ext. 3038, 3039/ posgrado.fcae@uaem.mx





Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

MARTIN GERARDO MARTINEZ RANGEL | Fecha:2020-12-11 15:29:37 | Firmante
eG3PjqucUmgf9yNUJFqDn9R1F8ulQ2Z3ga8BelGWjq6gQY2XQ8psiJnozKznBtwGdWRIR+ELrZYeCxVK5VaL7D.JM4cd75gP2cNjZUx+kzR2KEjU7EzwDrReTZEKamzSUN2SuFgJHCEzJRgNNoisGQJsc6Gxb1WYrI5aK4BAKC5SkelQ0vuQDAzWsdU86SsSJLg1TGRio5e73kB+HTnfDwoGgSVGU3nXDPvbaHVs1qlg0TWSnA6NVFYVOHyGW3O5GI8gxtQWj7HmvlUHP+sDWZEATjXK8ACxCqRIEDIGdVab62LwDiZCQSZygwK+HAWLWUMDuPZKA+hEQ==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:



V7s4pb

<https://efirma.uaem.mx/InoRepudio/BuoFZIsdLwgZECpUycPSXL84ccJRHns>



Una universidad de excelencia

RECTORÍA
2017-2023



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS



Cuernavaca, Morelos a 9 de noviembre del 2020.

DR. AUGUSTO RENATO PÉREZ MAYO
SECRETARIO DE INVESTIGACIÓN Y POSGRADO DE LA FCAeI
PRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Computo Aplicado, del estudiante Borunda Castillo Eiberth Adair, con matrícula 10023088, con el título **Heurística de Calendarización para el Problema de Asignación de Turnos para enfermeras de la Competencia Internacional 2014** por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación y Posgrado tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dra. Irma Yazmín Hernández Báez
Profesor- investigador
Facultad de Contaduría, Administración e Informática



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS

Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

IRMA YAZMIN HERNANDEZ BAEZ | Fecha:2020-12-18 10:47:34 | Firmante

od/q6RugljY6JDvSsgoo0GCmGWmEQzcXJEKpJdsxu5qgwLJNISyPv6BwdvulM6Tt4vKLOfOPCY4atPUA8vGppmk0CDX0zATPSu8J4AwzL5HBrQFBUNobNnOPTUyN0BAqY
p62WGuXGnik5GkllPCZ+ETFypJmVBAXrE438DrmHoogh8wRWW7ReMvT77WNMD/Br7uvZfzkvJ54ByDTy21EiEDovING0jEZM1NZMm8D91ZXSoVcd/u/Zxj3eP QIP621WRi0o5
EkbOdZ4S99SKMcdObZZvZgrBIVPNf4bO+QG Aof3O5XvVan8m00mwy7HBpVB9Ro0fZrI5E50wjhw==



Puede verificar la autenticidad del documento en la siguiente dirección electrónica o
escaneando el código QR ingresando la siguiente clave:

[zDvdJa](#)

<https://firma.uaem.mx/noRepudio/wbnNblQycG6a2NNLTz4PoR3mT6Ushvmd>



Una universidad de excelencia

RECTORÍA
2017-2023



Cuernavaca, Morelos a 9 de noviembre del 2020.

DR. AUGUSTO RENATO PÉREZ MAYO
SECRETARIO DE INVESTIGACIÓN Y POSGRADO DE LA FCAEI
PRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Computo Aplicado, del estudiante Borunda Castillo Eiberth Adair, con matricula 10023088, con el título **Heurística de Calendarización para el Problema de Asignación de Turnos para enfermeras de la Competencia Internacional 2014** por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación y Posgrado tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. José Crispín Zavala Díaz
Profesor- investigador
Facultad de Contaduría, Administración e Informática

Av. Universidad 1001 Chamilpa Cuernavaca Morelos México C.P. 62209, Edificio 19
Tel. (777) 329 7917, Ext. 3038, 3039/ posgrado.fcae@uaem.mx





Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

JOSE CRISPIN ZAVALA DIAZ | Fecha:2020-12-10 18:09:11 | Firmante
Zg2iOLLxB8kcsXUExxXkcXaSWJJwU2cu5vnV9b1dekjrS8wYX PVMHoA5EPgKRuEj6HcWXPWak+eXNGa1tXwPmlXSffs0RPDBa4ccmTizVis5E9TTJdQkuaoGLR/U5F9OUdU9
WuKtC4xxxrvpGr1EIV606EPsPFTI/hygXZq+Loubh9U3GZqV4BLvOxbaJ8JgoWK+ssjlCHg+9JGaCTshLq2a5jQvdkDapj9k1Hpwck3ksI3S/kDXwSBET+TXzkLj6jMryAO/nwzG5
NkJVzrPbICBrec53LQA TEf5InCY 7qUvd+WrbWT15XVWuZuicAUPyYZC2tW4Fn9p9FXQ==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o escaneando el código QR ingresando la siguiente clave:



s9mRnJ

<https://efirma.uaem.mx/noRepudio/h3eHoWm3IKZFAXTpDn2ThXSRuiDfQXQ>



Una universidad de excelencia

RECTORÍA
2017-2023



UNIVERSIDAD AUTÓNOMA DEL
ESTADO DE MORELOS



Cuernavaca, Morelos a 9 de noviembre del 2020.

DR. AUGUSTO RENATO PÉREZ MAYO
SECRETARIO DE INVESTIGACIÓN Y POSGRADO DE LA FCAeI
PRESENTE

En mi carácter de revisor de Tesis, hago de su conocimiento que he leído con interés la tesis para obtener el grado de la Maestría en Optimización y Computo Aplicado, del estudiante Borunda Castillo Eiberth Adair, con matricula 10023088, con el título **Heurística de Calendarización para el Problema de Asignación de Turnos para enfermeras de la Competencia Internacional 2014** por lo cual, me permito informarle que después de una revisión cuidadosa de dicha tesis, concluyo que el trabajo se caracteriza por el establecimiento de objetivos académicos pertinentes y una metodología adecuada para su logro. Además, construye una estructura coherente y bien documentada, por lo cual considero que los resultados obtenidos contribuyen al conocimiento del tema tratado.

Con base en los argumentos precedentes me permito expresar mi **VOTO APROBATORIO** por lo que de mi parte no existe inconveniente para que el estudiante continúe con los trámites que esta Secretaría de Investigación y Posgrado tenga establecidos para obtener el grado mencionado.

Atentamente
Por una humanidad culta
Una universidad de excelencia

Dr. Federico Alonso Pecina
Profesor- investigador
Facultad de Contaduría, Administración e Informática

Av. Universidad 1001 Chamilpa Cuernavaca Morelos México C.P. 62209, Edificio 19
Tel. (777) 329 7917, Ext. 3038, 3039/ posgrado.fcae@uaem.mx





Se expide el presente documento firmado electrónicamente de conformidad con el ACUERDO GENERAL PARA LA CONTINUIDAD DEL FUNCIONAMIENTO DE LA UNIVERSIDAD AUTÓNOMA DEL ESTADO DE MORELOS DURANTE LA EMERGENCIA SANITARIA PROVOCADA POR EL VIRUS SARS-COV2 (COVID-19) emitido el 27 de abril del 2020.

El presente documento cuenta con la firma electrónica UAEM del funcionario universitario competente, amparada por un certificado vigente a la fecha de su elaboración y es válido de conformidad con los LINEAMIENTOS EN MATERIA DE FIRMA ELECTRÓNICA PARA LA UNIVERSIDAD AUTÓNOMA DE ESTADO DE MORELOS emitidos el 13 de noviembre del 2019 mediante circular No. 32.

Sello electrónico

FEDERICO ALONSO PECINA | Fecha:2020-12-10 20:30:13 | Firmante

KvbIHUErDusyPqPpmFe8Xn9VoBZmVprOKXiEopHZeR2O78iJmVJNfTjdmJMq5JN5WCJ1aynCRlx1nC8eTbTD66GCYFaf1LPhznZsJ8Sx4iGp8VOrmxPFUe/PjHTSLI/mg3z/oB
gxeUMypki0LWUgSTxKYdzcNEZVFQNY64eXzVQE25bAG4MVP1oYIX2b5unBYsRIYPo6A7n9k+CSjXj4xtYs+6viUNC2BY/VT/PTcksYYUH2yDE1zOQK46M2gCdUWDr8Z3tM+
sdWLFplB3NnrL1XRwpGPtOfOhcXBx4hrOjc2QopdlIPzZoGnHuWSkXPacWEL1Y56uRRmVlrwVBw==

Puede verificar la autenticidad del documento en la siguiente dirección electrónica o
escaneando el código QR ingresando la siguiente clave:



JodS9

<https://efirma.uaem.mx/hnoRepudio/GggbzpFHjBIIPQVjAykY80x0Cvc2D1Bn>



Cuando te propones reparar un mundo tan complejo que además no conoces en su totalidad no sabes por dónde empezar, pero si lo deseas tanto como deseas respirar entonces empieza contigo mismo no tengas miedo a lo desconocido piensa como cuando eras niño, se curioso, humilde, respetuoso y siempre diviértete con lo que haces, enséñale esto a los demás y de pronto verás un mundo que no necesita ser arreglado.

Agradecimientos.

En este camino que llamamos vida donde algunos creen que todo está escrito en piedra y otros confían en que pueden cambiar lo impensable, doy gracias a Dios por ser nuestro amigo y que gracias al don del libre albedrío podemos llegar hasta donde nos lo proponemos. Doy gracias a que personas como mi padre dan todo de si por él y para que sus hijos salgan adelante dentro de sus posibilidades, enfermedades y debilidades. Doy gracias por las segundas oportunidades y que mis hermanas están sanas. Doy gracias a mi esposa por todo el amor, paciencia y entendimiento que me regala y la hermosa familia que hemos formado. Doy gracias por la salud y bienestar de todos mis amigos y seres amados. Doy gracias al Dr. Federico Alonso Pecina que no solo es un gran docente e investigador si no también una persona con un gran corazón.

Doy gracias y mando saludos a un gran equipo de docentes Dr. Martín G. Martínez Rangel, Dr. Gilberto Calvillo Vives, Dr. José Alberto Hernández Aguilar, Dr. Luis Manuel Gaggero Sager, Dr. Outmane Oubram, Dr. José Crispín Zavala Díaz, Dr. Martin Heriberto Cruz Rosales y Dra. Irma Yazmín Hernández Baez.

Además, agradezco a CONACYT por la beca que me otorgaron durante el desarrollo de esta tesis.

Resumen

El problema que se aborda en este trabajo es el de la automatización de la programación de turnos para enfermeras una actividad manual que existe en los hospitales. El estudio en las Ciencias Computacionales de este problema lleva más de cuarenta años, además debido a crisis económicas, falta de personal, o alguna otra circunstancia es necesario implementar métodos que ayuden a su entendimiento y solución. En este documento se realizó una recopilación de información y métodos que datan del año 1972 hasta el año 2019, donde se describe que algunos de los problemas principales abordados en la literatura especializada. Cada hospital tiene restricciones particulares, equipos de trabajo con grandes diferencias de contratos, habilidades, horarios y necesidades, el problema ha sido clasificado como NP-Duro en distintas variantes. El problema que se aborda en este documento es el presentado en el Second International Nurse Rostering Competition 2014, que cuenta con cuatro restricciones duras y siete restricciones suaves, el cual se resolvió desarrollando dos métodos para encontrar soluciones iniciales de distintas calidades y tiempos. En esta tesis se logró implementar una heurística ad-hoc de calendarización de enfermeras tomando en cuenta todas las restricciones duras, restricciones suaves, contratos, fuerza laboral para cada instancia, ordenamientos y la implementación de un criterio donde se consideran las restricciones suaves como duras a las que se les aplica una relajación creciente y aleatoria en los límites de los contratos esto permitió encontrar soluciones factibles de mejor calidad comparadas con un método determinista, estas soluciones iniciales también permiten la aplicación de vecindarios deterministas o aleatorios respetando las cuatro restricciones duras y las siete restricciones suaves es así que se logra mejorar las soluciones iniciales del método no determinista. En este trabajo se logró encontrar soluciones factibles para todas las instancias que componen el benchmark del INRC-II en tiempos razonables.

Abstract

The problem addressed in this work is the automation of shift scheduling for nurses, a manual activity that exists in hospitals. The study in Computer Science of this problem takes more than forty years, also due to economic crisis, lack of personnel, or some other circumstance it is necessary to implement methods that help to understand and solve it. In this document, a compilation of information and methods dating from 1972 to 2019 was made, which describes some of the main problems addressed in the specialized literature. Where each hospital has particular restrictions, work teams with great differences in contracts, skills, schedules and needs, the problem has been classified as NP-Hard in their different variants. The problem addressed in this document is the one presented in the Second International Nurse Rostering Competition 2014, which has four hard restrictions and seven soft restrictions, this problem was resolved by developing two methods to find initial solutions of different qualities and times. In this thesis, it was possible to implement an ad-hoc heuristic for nursing scheduling, taking into account all the hard restrictions, soft restrictions, contracts, workforce for each instance, regulations and the implementation of a criterion where soft restrictions are considered as hard to those that are applied an increasing and random relaxation in the limits of the contracts, this allowed finding feasible solutions of better quality compared to a deterministic method, these initial solutions also allow the application of deterministic or random neighborhoods respecting the four hard restrictions and the seven soft constraints are thus able to improve the initial solutions of the non-deterministic method. In this work, it was possible to find feasible solutions for all the instances that make up the INRC-II benchmark in reasonable times.

Tabla de contenido

Capítulo 1.....	18
1.1. Antecedentes.....	18
1.2. Planteamiento del problema	21
1.3. Justificación de la investigación.....	23
1.4. Hipótesis	25
1.5. Objetivos.....	25
1.5.1. Objetivo General	25
1.5.2. Objetivos Especificos.....	25
1.6. Metodología general	26
1.7. Estructura de los capítulos	26
Capítulo 2.....	28
2. Estado del arte.....	28
2.1. Diagrama.....	28
2.2. Asignación de enfermeras descripción general.....	29
2.3. Pruebas de complejidad NP-Duro.....	30
2.4. Ramificación y poda	31
2.5. Programación lineal y entera	32
2.6. Optimización multi-objetivo	33
2.7. Inteligencia artificial y modelación matemática	35
2.8. Búsqueda en vecindario variable.....	36
2.9. Recocido simulado	38
2.10. Búsqueda tabú.....	41
2.11. Búsqueda local.....	44
2.12. Algoritmos Voraces	46
2.13. Enjambres	47
2.14. Coloreo de grafos	49
2.15. Sistemas expertos y conocimiento basado en sistemas	50
Capítulo 3.....	54
3. Origen del INRC-II	54
3.1. Descripción general de escenario, historial y demanda.	54
3.2. Restricciones duras.....	55
3.3. Restricciones suaves.....	58

3.4.	El formato de los archivos.....	67
3.4.1.	Escenario.....	67
3.4.2.	Archivo datos semanales (Demanda).....	71
3.4.3.	Archivo historial	72
3.4.4.	Archivo solución	74
3.4.5.	Evaluador de factibilidad	77
3.4.6.	Instancias del problema	80
3.5.	Modelo matemático.....	82
Capítulo 4.....		88
4.	Solución	88
4.1.	Esquema general	88
4.2.	Métodos desarrollados	90
4.2.1.	Algoritmo determinista.....	90
4.2.2.	Método no determinista.....	94
4.2.3.	Fase B optimización mediante vecindarios	105
4.2.4.	Tablas de resultados.....	115
Capítulo 5.....		123
5.	Conclusiones y trabajo futuro.....	123
5.1.	Conclusiones.....	123
5.2.	Trabajo futuro.....	124
Referencias.....		125

Capítulo 1.

1.1. Antecedentes

A lo largo de la historia de la humanidad ha existido la enfermedad y ante esa complicación quienes cuiden de los enfermos. Ahí es cuando hace presencia el altruismo de algunos humanos, se ve nacer la enfermería y toda profesión es una creación humana (Arratia F., 2005). ¿Pero quien cuida de lo que necesitan estos profesionistas?, ¿Dónde está el pago de esa humanidad que entregan todos los días a sus pacientes?. Teniendo en cuenta todos los problemas presentes al organizar o asignar de forma correcta, a todo el personal de enfermería debido a falta de tiempo, una mala organización o al hecho de que se cuenta con poco personal, Mischek & Musliu (2016) nos dicen que la generación automatizada de horarios para el personal con resultados de alta calidad, en particular para los hospitales, ha sido un problema muy importante por casi cuarenta años. Arratia F. (2005), nos dice que debemos enfatizar la necesidad de resaltar la importancia de la investigación en enfermería, y colocar sus teorías, modelos y desarrollo dentro de un contexto vigente y consistente con la realidad, también tener en cuenta las necesidades del medio en el que se desenvuelve.

Como podemos darnos cuenta, cada día están más llenas las salas de espera en los hospitales ya sean privados o generales, es así cómo podemos observar que la demanda del personal de cuidados de la salud llega a ser mayor que todas las flotillas que podrían sumar de doctores, enfermeras y paramédicos de todos los hospitales del país o el mundo. Legrain, Bouarab & Nadia(2015), nos dicen que se está experimentando un gran incremento en la demanda de personal para cuidados de la salud a causa de que el acceso a la salud se trata de que sea universal, es entonces y relacionado a esta medida que hay una disminución en la tasa bruta de mortalidad, pero debido al incremento de personas sanas de todas las edades, la población incrementa y el presupuesto no crece a la par por lo cual llega a no ser el suficiente ni el dinero

ni el personal. Debido a esta insuficiencia de presupuesto y personal en los hospitales, nos enfocaremos en el problema de asignación de enfermeras debido a que son un pilar muy afectado en los hospitales y este ha demostrado ser una profesión necesaria e inamovible para el funcionamiento correcto de las actividades que se tienen en el área de la salud día a día, Legrain, Bouarab, & Nadia (2015), nos dicen que las enfermeras son responsables de varias actividades médicas, representan aproximadamente el 25% del presupuesto operativo total del hospital y el 44% de los costos de atención directa. Es por eso que se considera necesario contar con una forma más viable de obtener una asignación de enfermeras que no sea el método tradicional manual ya que de acuerdo a K. De Causmaecker, Berghe, & Landeghem(2004), la administración del personal de un hospital en particular es un reto, porque para cubrir satisfactoriamente la planeación de una demanda semanal se necesita que las enfermeras se presenten en distintos días, turnos y es debido a esto que no es fácil escoger ya que a esta demanda y asignación está sujeta a que todos tienen una o más habilidades, contratos y horarios disponibles además se debe tomar en cuenta que las instituciones de la salud trabajan contra reloj y que cada enfermera tiene sus demandas personales de descanso o emergencias familiares.

Esto nos hace ver que una mala planeación de la cobertura de la demanda repercute en todos los niveles de un hospital, desde la calidad de la atención que una enfermera con pocas horas de sueño pueda brindar, hasta la economía del hospital por pagar outsourcing o un exceso de tiempo extra a las enfermeras que cubran horas fuera de su contrato, y como es bien sabido en esta área se tiene el tiempo encima debido a que las emergencias llegan en cualquier momento esto desplaza toda la planeación que a un grupo de personas le tomo horas llevarla a cabo manualmente.

El concepto de la planeación de cobertura de la demanda de enfermeras ha cambiado con el paso de los años, programar a mano esta tarea consume una gran cantidad de tiempo, y varios hospitales que ya cuentan con sistemas computarizados para programar esta tarea no están explotando en su totalidad las herramientas con las que podrían contar, y la importancia de crear

un aproximamiento sistemático es necesario, pero este tipo de generación de horarios no está disponible en general, y es sumamente importante crear tablas de manejo de tiempos de alta calidad en el área de cuidados de la salud, (K., De Causmaecker, Berghe, & Landeghem, 2004).

Podríamos denotar que hacer los cálculos de éstas asignaciones estocásticamente o deterministamente es posible, unas de las publicaciones que tienen que ver exactamente con este problema hasta el momento, es hecha por (Mischek & Musliu, 2016) y Santos D. et al., (2015), quienes para resolver este problema usarón un enfoque de satisfacción de las restricciones ponderado. Ceschia, T. et al., (2018) los organizadores de este concurso en su segunda edición, nos dicen que la asignación de enfermeras es un problema a resolver muy importante en la administración del cuidado de la salud, y que los primeros papeles que se tienen documentados datan de los años sesentas, aun así ésta tomo mayor importancia en particular el siglo XXI, y fue entonces que se hizo una primer convocatoria en 2010 con quince aportes y tres categorías, donde varios grupos de investigadores produjeron resultados muy remarcables así como también se encontrarón y reportarón soluciones optimas, gracias a los resultados obtenidos del concurso anterior en el concurso del 2015 proveen de dos programas, un simulador y un evaluador para los concursantes e interesados en resolver de forma más avanzada y realista el problema de asignación de enfermeras al tener más restricciones y agregar el estado multi-etapas. En el 2020 lo antes mencionado se agravo debido a una pandemia global causada por el virus SARS-CoV-2 (Coronavirus) donde se puede observar que uno de los mayores impactatos fue en el sector salud y debido a la alta demanda del personal, explotación, enfermedad, ect. Se perdio una gran cantidad de enfermeras y medicos lo que afecto en gran medida a los hospitales al volverse aun más indispensables pero inestables, lo que los llevo a tener salas de espera saturadas y calles con multitud de personas que buscaban saber el estado de su familia o familiares pero al mismo tiempo agravaban la situación por no respetar las indicaciones para evitar mas contagios y ejercian presión sobre el personal de la salud, es por ello que es de vital importancia sistematizar tareas administrativas complejas que permitan un mejor cuidado del personal teniendo en cuenta o suponiendo que las computadoras utilizadas en estos hospitales tienen un bajo rendimiento computacional estos programas deberan ser capaces de correr en estos

computadores y es por eso que los problemas como el presentado en el Second International Nurse Rostering Competition (INRC-II) ayudan con el planteamiento de las necesidades que deben ser cubiertas y de los resultados se aprende que se puede mejorar en el futuro. Para descargar la información de este concurso se puede visitar el siguiente URL: <http://mobiz.vives.be/inrc2/>.

1.2. Planteamiento del problema

Conforme el tiempo transcurre las necesidades humanas cambian debido a la implementación de los resultados logrados por la investigación, estos logros impactan en la vida de las personas al descubrir nuevas formas de hacer, mejorar o sistematizar procesos, lo que nos lleva actualmente a investigar como desarrollar un programa que se apegue a las necesidades actuales del personal de enfermería en los hospitales, con la finalidad de que en su implementación se pueda disminuir los desajustes en el tiempo y minimizar el gasto del presupuesto de hospitales, estos problemas son causados por algún error no persivido o utilizar el tiempo de un grupo de enfermeras en hacer esta programación manual y no en otras actividades más relacionadas a la salud. En la planeación de asignación de turnos para enfermeras se busca cubrir la demanda proyectada por instancia la cual puede compartir un tamaño de entre cuatro u ocho semanas pero cada una de estas semanas tendra una demanda distinta que varia solo por instancia, esto se observa que fue hecho con la finalidad de apegarse a lo variante que es un hospital con sus citas programas y emergencias del momento, se busca satisfacer este problema con la distribución adecuada del personal enfermero para evitar gastos inecesarios cuando se cuenta con apenas el personal suficiente para satisfacer los escenarios presentes en una demanda en la que una asignación tiene distintos requerimientos como tipo de turno que se busca cubrir y que habilidad se solicita para poder cubrir este turno, el enfoque tomado para la factibilidad y optimización es respetar las once restricciones que se tienen disponibles en el documento de INRC-II las cuales son:

Restricciones duras:

- H1 Cada enfermera puede cubrir máximo un turno por día.
- H2 Cubrir la demanda mínima de enfermeras en la instancia.
- H3 las asignaciones continuas deberán respetar las condiciones de asignación legales.
- H4 La enfermera asignada deberá contar con la habilidad solicitada en la demanda.

Restricciones suaves:

- S1 Cobertura óptima de la demanda (30).
- S2 S2a. Respetar límites mínimos y máximos de turnos consecutivos asignados (15).
S2b Respetar límites mínimos y máximos de días consecutivos de trabajo asignados (30).
- S3 Respetar los límites mínimos y máximos de días consecutivos de descanso (30).
- S4 Preferencia de día o turno libre (10).
- S5 Trabajar completo el fin de semana (Sábado y domingo) si esta restricción esta activa (30).
- S6 Respetar los límites mínimos y máximos de días trabajados de forma global (20).
- S7 Respetar el límite máximo de fines de semana trabajados (30).

Las restricciones duras se enfocan en evitar el agotamiento físico y mental de las enfermeras mientras buscan satisfacer el mínimo número de enfermeras con las habilidades requeridas para cubrir la demanda de atención del día. La restricción H1 de tener un solo turno de trabajo al día se presenta en varias regiones del mundo como Canadá y Europa, esto ayuda a que las enfermeras tengan un descanso adecuado y desempeño para tratar con las urgencias y pacientes difíciles. La restricción H2 busca cubrir el personal mínimo necesario para cumplir todas las tareas del día. restricción H3, en el mismo tenor de la H1 prohíbe sucesiones de turnos que afecten el descanso adecuado al personal. La restricción H4 se asegura que la demanda esté cubierta por enfermeras con la habilidades necesarias para cubrir las tareas del día.

Las restricciones suaves buscan respetar los límites mínimos y máximos en días trabajados, días de descanso, preferencias de las enfermeras, fines de semana, etc, con el fin de mejorar la calidad del horario para el personal de enfermería. La restricción S1 busca satisfacer la demanda óptima de personal para las actividades en el trabajo, naturalmente algunas tareas se llevarían a cabo de forma más eficiente cuando más de una enfermera se encuentra disponible. La restricción S2 penaliza cuando se viola el mínimo y máximo número de turnos o días consecutivos asignados a una enfermera lo que sugiere la rotación del personal en distintos turnos; los días de trabajo mínimos y máximos asignados consecutivamente nos penalizan como es de forma natural en cualquier trabajo. La restricción S3 como un complemento de la anterior penaliza el exceso y la falta de descanso de una enfermera, ya que no descansar adecuadamente baja la calidad de atención y servicio a los pacientes. La restricción S4 se enfoca en permitir que las enfermeras puedan elegir un día en el que prefieren no trabajar o solo no ser asignadas a un turno específico, esto tiene beneficios personales al ocupar estas oportunidades para las actividades pendientes en su vida privada, teniendo en cuenta que solo es una petición que la programación de las enfermeras tomara en cuenta cuando su asistencia no sea de vital importancia. La restricción S5 asignar el fin de semana completo: trabajar tanto sábado y domingo o bien ninguno de los dos, está se encuentra activa en algunas instancias posiblemente por el requerimiento de que algunos hospitales en el mundo trabajen con este formato. La restricción S6 los límites globales se refiere a una característica en los contratos en la que se debe respetar una cantidad de asignación de mínimos y máximos para todo el periodo que se está trabajando, esto se refiere a que se tomará en cuenta que sucede con cada enfermera durante todo el tiempo que comprende la programación (4 u 8 semanas). La restricción S7 trata de limitar el número de fines de semana que trabaja una enfermera durante el periodo que comprende la programación.

1.3. Justificación de la investigación

Este problema según Steven Den Hartog, J.A. Hoogeveen & H.L. Bodlaender (2016), está categorizado como NP-Duro donde encontrar un óptimo global para cada enfermera consumen un tiempo de 2^{n-1} , en esta complejidad cuando se permiten tiempos largos la experimentación

para resolver una instancia de dieciséis enfermeras consume un tiempo exponencial de procesos computacionales en un algoritmo híbrido como se observa en el trabajo de Edmund K. Burke, Timothy Curtois, Gerhard Post, Rong Qu & Bart Veltman, (2007) [A hybrid heuristic ordering and variable neighborhood search for the nurse rostering problem]. Hay muchos problemas sobre asignación de enfermeras con diferentes metas en general tratan distintas cantidades de días o turnos, algunos detalles específicos y problemáticos para otros hospitales, los resultados en la literatura para comparar los resultados obtenidos son muchos pero debido a la naturaleza del problema de asignación de enfermeras se necesita fomentar la investigación de este tema como se sugiere por Sanja Petrovic & Greet Vanden Berghe, (2010) [Comparison of Algorithms for Nurse Rostering Problems] se debe conocer cual variedad de asignación de enfermeras se está trabajando para poder conocer las heurísticas que se pueden aplicar, que tipo de poder expresivo, flexibilidad, capacidades de aprendizaje, mantenimiento, capacidad de reprogramación o parámetros de afinación son compatibles y que resultados se obtienen al implementarlos, se considera que el desarrollo de un algoritmo ad-hoc que nos entregue soluciones factibles es el pilar más importante en la solución del problema de Nurse Rostering, ya que después se puede implementar una metaheurística que optimice esta solución, y así poder cumplir los estrictos tiempos de proceso que se tienen para generar una respuesta óptima, la finalidad de tomar este camino sería resolver el problema INCR-II en tiempo aceptable de acuerdo con las recomendaciones de los organizadores del concurso INRC-II que varía conforme cuantas enfermeras componen la instancia que se está solucionando, esta tarea de asignación a un grupo de enfermeras profesionales les toma aproximadamente cuatro horas, y aun así la precisión humana suele no ser la mejor ya que tienden muchas veces a parchar los errores sobre la marcha, esto lleva que a fin de mes los hospitales se vean cortos de presupuesto, debido a los gastos extra que conlleva corregir una mala cobertura de la demanda de enfermeras en tiempo real, ya que es técnicamente imposible que se alcance un estado de cero violaciones en un caso real.

En el INRC-II (Second International Nurse Rostering Competition) toman en cuenta aspectos reales de los problemas que se presentan en un hospital como se menciona en el trabajo de Dragan Simic, Svetlana Simic, Dragana Milutinovic & Jovanka Djordjevic, (2014) en este problema

la complejidad es alta al tener cuatro restricciones que al ser violadas cualquiera de ellas vuelven infactible la solución, siete suaves que penalizan con cada violación y a esto se le suma que se deben resolver más de una semana por instancia los casos son cuatro o ocho semanas, de las cuales sólo se da a conocer la información de la semana actual debido a que estas semanas se componen de demandas variantes donde varios o todos los casos de la demanda cambiaran y estos cambios solo se conocerán al terminar de resolver una semana y poder continuar con la siguiente semana que compone a la instancia actual, también se agrega una lista de preferencias que tienen las enfermeras y un historial que nos dice que sucedió anteriormente en el hospital, debiendo tomar en cuenta esta información para que las futuras asignaciones no violen las restricciones duras. Se encuentra disponible un validador de soluciones del problema INCR-II en internet en el siguiente URL: http://mobiz.vives.be/inrc2/?page_id=245.

1.4. Hipótesis

Al desarrollar una heurística ad-hoc se encontrarán soluciones factibles en cada instancia del benchmark propuesto en el problema INRC-II y la calidad de estas soluciones factibles podrá mejorarse mediante el uso de vecindarios.

1.5. Objetivos

1.5.1. Objetivo General

Resolver el problema en dos fases: la primera fase utilizar una heurística ad hoc para encontrar un solución factible y en la segunda una heurística basada en vecindarios para la mejora del peso de la solución.

1.5.2. Objetivos Especificos

- 1• Producir un algoritmo ad-hoc con base en la información y el benchmark del concurso.

- 2• Hacer un estudio de las restricciones suaves violadas para conocer la distribución de la violación de las restricciones suaves
- 3• Diseñar vecindarios que se enfoquen en las restricciones que más impactan a la función objetivo

1.6. Metodología general

Para resolver el problema INRC-II, se llevarán a cabo los siguientes pasos:

- I- El primer paso es diseñar una heurística que satisfaga semana a semana la demanda mínima de enfermeras mientras respeta las sucesiones de turnos y no asigna a una enfermera más de una vez al día. El algoritmo se hará respetando las restricciones duras del problema INRC-II.
- II- El segundo paso es efectuar un estudio de las restricciones suaves violadas para conocer cuáles son las que más impactan a la función objetivo.
- III- El tercer paso, es diseñar vecindarios que mejoren la función objetivo, tomando en cuenta la literatura de las investigaciones de quienes ya solucionaron este problema o similares. Se tomará en cuenta los resultados alcanzados con base en el tiempo máximo y sin el tiempo máximo que se establece en la competencia para procesar cada instancia del benchmark.
- IV- En el cuarto paso, se hará la experimentación pertinente, para sacar medidas estadísticas como media, mediana y desviación estándar, para conocer el desempeño del algoritmo

1.7. Estructura de los capítulos

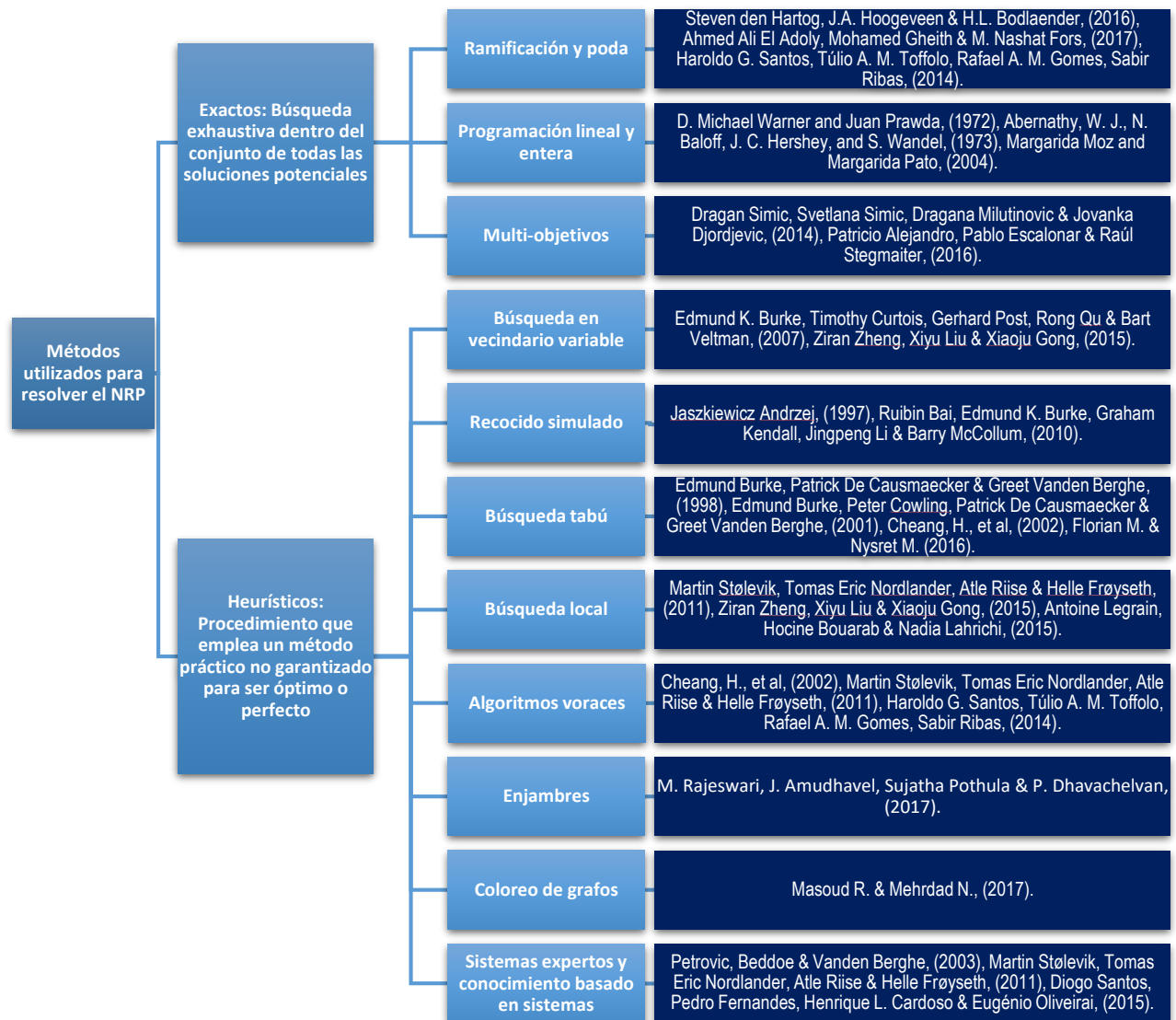
El capítulo dos contiene algunos trabajos de cómo han solucionado a través del tiempo el problema de asignación de turnos de enfermeras en los hospitales desde instancias artificiales hasta instancias reales. El capítulo tres contiene el planteamiento y el modelado del problema y las instancias del INCR-II. El capítulo cuatro muestra el método de solución para crear el solucionador de programación de turnos de enfermeras y los resultados obtenidos con el desarrollo del solucionador de programación de turnos de enfermeras, por último, el capítulo cinco contiene las conclusiones y trabajo futuro.

Capítulo 2.

2. Estado del arte

2.1. Diagrama

En la siguiente imagen se puede observar una parte de los métodos utilizados para resolver el problema de asignación de enfermeras:



2.2. Asignación de enfermeras descripción general

La estimulación de la investigación en esta área es necesaria para encontrar conjuntos de algoritmos que ayuden a mejorar la literatura y resultados conocidos para que en la continua superación se logre el registro de nuevos logros con la finalidad de que aquellas personas interesadas en resolver el problema de asignación de enfermeras: Nurse Rostering Problem (NRP) en inglés. En la vida real cuentan con estados del arte actualizados, repletos de métodos y heurísticas ya probadas con anterioridad, todo esto a favor de que la ciencia siga avanzando Edmund K. Burke, Patrick De Causmaecker, Greet Vanden Berghe & Hendrik Van Landeghem, (2004) este problema contará con dos tipos de restricción duras y suaves dependiendo del objetivo del caso a resolver se buscare factibilidad u optimizar. En los problemas más apegados a la realidad de optimización de NRP se debe tener en cuenta que casi ningún hospital tendrá las mismas políticas administrativas y se deberá tomar en cuenta que hay cuatro áreas mayores para las decisiones las cuales son poder laboral, variabilidad de la demanda, administración y reasignación, todo esto acompañado de cinco criterios:

- Cobertura: que tanta diferencia hay entre nuestro equipo de trabajo y la tarea a cubrir.
- Calidad: que tan ajustada esta la solución o que tan demandante es la solución propuesta.
- Estabilidad: cómo percibe el personal la consistencia de la administración.
- Flexibilidad: que tan adaptable es el sistema a los cambios del ambiente laboral.
- Costo: cuantos recursos consume la decisión tomada, como ejemplos cuanto personal o tiempo de cómputo toma esta.

Gracias a estudios recientes como el de Amy Witkoski Stimpfel, Douglas M. Sloane & Linda H. Aiken, (2019) en la actualidad se cuenta con mejores descripciones sobre las necesidades de las enfermeras en los hospitales teniendo en cuenta también la satisfacción del paciente en los resultados de sistemas de encuesta anuales, tomando una muestra de hasta 22,275 encuestados en 577 hospitales de varias zonas de EUA donde al menos hay diez enfermeras por hospital, pero la media es de treinta y nueve enfermeras con horarios de entre ocho a trece horas más horas extra trabajadas “voluntariamente”, esto debería estar regulado en horarios de ocho u doce

horas como lo marca la ley en esta región, en general este tipo de estudios es para lograr una dicotomía de la información entre una buena o mala asignación tomando en cuenta las variables de control el análisis de los datos y las limitaciones de estos estudios para lograr resultados veraces. Este estudio arroja que el mayor porcentaje de enfermeras insatisfechas son las que trabajan en jornadas de entre diez y once horas o las que trabajan trece horas o más, con colapsos a partir de jornadas superiores a ocho horas debido a la demanda física y mental pero los turnos de doce horas generalmente reportan tener horas de descanso intermedias ayudando a evitar estas insatisfacciones o percances.

2.3. Pruebas de complejidad NP-Duro

No se abordará a profundidad en la complejidad del NRP debido a que varios trabajos ya clasificaron este problema como NP-Duro, describiremos con ejemplos muy generales lo que se presenta en los siguientes trabajos, donde Diogo Santos, Pedro Fernandes, Henrique Lopes Cardoso & Eugénio Oliveirai, (2015) nos dice que los problemas clasificados como NP-Duro han necesitado de años de investigación para encontrar métodos que impacten en las soluciones y también afirman que es imposible tratar de resolver este problema en tiempo razonable con métodos exactos.

Steven den Hartog, J.A. Hoogeveen & H.L. Bodlaender, (2016) nos dicen que el problema puede separarse en varias subsecciones las cuales se modelan matemáticamente y durante esta categorización del NRP se comprueba que aunque algunas subsecciones son de tiempos polinomiales otras tienen una complejidad de NP-Duro, y al encontrar solo una subsección que sea de tiempos NP-Duro hace que el problema en general sea NP-Duro, describiendo que si en un problema de NRP se presentan dos restricciones duras de una misma naturaleza como cobertura de turnos estricta y cobertura de turnos solo pueden suceder dos cosas, contradicción o coexistencia aun si una de las restricciones duras pasa a ser suave este sigue siendo NP-Duro, en general la clasificación del problema se refiere que tan difícil es conseguir la mejor de las respuestas en tiempo razonable debido a que la mayor desventaja de algunos métodos es que consumen un tiempo exponencial al realizar una búsqueda exhaustiva dentro del conjunto de todas las soluciones potenciales.

Además M. Rajeswari, J. Amudhavel, Sujatha Pothula & P. Dhavachelvan, (2017) hablan de que en el NRP se deben considerar restricciones duras y suaves lo que en varios trabajos han comprobado es que debido a la naturaleza de estas restricciones lo hacen un problema de difícil optimización debido al inmenso espacio de soluciones y se necesita la investigación de métodos novedosos que ayuden a resolver por separado las distintas secciones en las que se puede descomponer este problema NP-Duro.

2.4. Ramificación y poda

También Steven den Hartog, J.A. Hoogeveen & H.L. Bodlaender, (2016) planeando utilizar a futuro herramientas como ramificación y poda primero diseñan un modelo matemático para un NRP que trata de la cobertura de veinticuatro horas divididas en tres turnos (temprano, tarde y noche) con cinco restricciones (Cobertura de turnos, asignaciones balanceadas, Secuencias prohibidas, peticiones, habilidades requeridas), hablando de que el problema es NP-Duro y que tres de las restricciones no pueden ser fácilmente satisfechas relacionadas con cobertura de turnos estricta, fines de semana idénticos y fines de semana libres además descomponen en tres puntos importantes el problema, las restricciones a las enfermeras, las restricciones de cobertura y el objetivo de optimización logrando un modelo matemático que sirva para futuras experimentaciones.

Otra investigación es la de Ahmed Ali El Adoly, Mohamed Gheith & M. Nashat Fors, (2017) se utilizó un algoritmo de ramificación y poda (implementada en Gurobi 7.0.1) para resolver diecisiete instancias con un máximo de cincuenta enfermeras donde el problema se componía de seis restricciones que deben ser satisfechas las cuales son, cada enfermera puede ser asignada a dos turnos por día, cada enfermera debe cubrir un número definido de turnos de noche mínimos, cada enfermera debe cubrir un número de turnos de tarde mínimos, cada enfermera debe trabajar entre lapsos mínimos y máximos de asignaciones de turnos durante toda la programación, si una enfermera tiene un turno en un día que se especifico debía estar libre no se deberá asignar los siguientes dos turnos, si tuvo dos turnos continuos deberá estar libre los siguientes tres turnos para después aplicar el modelo matemático propuesto en un caso real de

un hospital de Alejandría Egipto con las siguientes características, cuarenta y seis enfermeras, tres turnos por día, se inicia la semana en sábado, el hospital trabaja durante siete días pero a las enfermeras se les permite trabajar entre seis y ocho turnos cada semana, a las jefas de enfermera solo se les permite trabajar por las mañanas, sobre estos datos se experimentó con dos escenarios diferentes del hospital real y los resultados obtenidos demostraron una clara mejora contra la asignación manual, las ventajas obtenidas fueron calculadas por semana, las cuatro horas manuales de asignación pasaron a ser menos de un segundo, se logró equilibrar la carga de trabajo con el beneficio de reducir el costo de las horas extra en hasta un treinta y seis por ciento.

2.5. Programación lineal y entera

Dependiendo del enfoque utilizado en este método se pueden desarrollar varias estrategias gracias al poder de dividir el problema en varias fases de solución como ordenamientos de disponibilidad, reforzar con la información de cada empleado con la finalidad de tener una estructura de control fuerte como nos dicen en el trabajo de D. Michael Warner and Juan Prawda, (1972) consiste en primero calcular con un método mixto entero el número necesario de enfermeras de acuerdo con las características solicitadas cada día, con una función objetivo que tiene como meta minimizar la diferencia entre los límites inferiores dados a las enfermeras y las variables, al considerar la posibilidad de reemplazar los estándares establecidos por la organización y agregar personal con diferentes habilidades para así poder reducir los costos de no tener el personal suficiente para el problema.

El NRP de acuerdo con Abernathy, W. J., N. Baloff, J. C. Hershey, and S. Wandel, (1973) no debe ser tratado como los clásicos problemas de asignación de personal y la organización de las enfermeras debe ser dividida en tres niveles de decisión los cuales son decisiones políticas, planeación del personal y términos de asignación a corto plazo de acuerdo con los primeros dos niveles descritos, es entonces que dan un ejemplo hipotético donde se puede aplicar la asignación individual de cada enfermera estocásticamente cuando la demanda no está predeterminada.

Generalmente este problema se resuelve a mano por las enfermeras administrativas y para un caso en el que ya se tiene una solución dada Margarida Moz and Margarida Pato, (2004) desarrollaron un modelo de remplazos de guardias para poder ser aplicado en este caso real de hospital donde respetan las restricciones duras y minimizan los costos en relación a la programación original con la finalidad de respetar la vida privada de las enfermeras.

En opinión de Edmund K. Burke, Patrick De Causmaecker, Greet Vanden Berghe & Hendrik Van Landeghem, (2004) este método no tiene la capacidad de abordar todas las necesidades de un hospital debido a que una de sus mayores limitantes es que no se pueden incluir las necesidades de las enfermeras o emergencias sin incluir otros métodos o ajustes.

2.6. Optimización multi-objetivo

Los métodos de optimización por programación matemática se basan en encontrar soluciones óptimas basados en una sola función objetivo lo que en general los vuelve no aplicables a problemas muy grandes y complejos, estos algoritmos se adaptan a este tipo de problemas al utilizar la optimización multi-objetivo volviéndose tratables una gran cantidad de problemas altamente restringidos o la incorporación de algunas ideas de este método en otros métodos más modernos para conseguir respuestas en tiempo razonable.

También nos dice Edmund K. Burke, Patrick De Causmaecker, Greet Vanden Berghe & Hendrik Van Landeghem, (2004) que las ventajas de este enfoque radican en que se consideran varios o todos los criterios de acuerdo con un orden de prioridad como balancear el personal, la demanda y no exceder los tiempos de trabajo del personal todo esto con la finalidad de tener la posibilidad de alcanzar optimizar todos los criterios de evaluación pero al tener varias metas lo que se busca es tener una evaluación lo más cercana posible a cada objetivo final, para lograr este tipo de razonamiento en el NRP se ha necesitado de muchos años de investigación en problemas de asignación y este mismo razonamiento seguirá jugando un rol importante en el desarrollo futuro de más descubrimientos de métodos modernos para el NRP, un trabajo más reciente es el de Patricio Alejandro, Pablo Escalonar & Raúl Stegmaiter, (2016) y nos dicen que resolver este

problema es combinatorio debido a su complejidad de NP-Duro, en este enfoque se busca desarrollar un modelo de programación matemática para poder resolver el problema en tiempos polinomiales de forma cíclica o no cíclica pero este tipo de programación requiere un alto nivel de precisión y debido a que el tamaño real de las instancias del NRP es intratable, entonces como solución inicial se busca satisfacer todas las restricciones duras y después valorar la calidad de la solución agregando algunas restricciones suaves donde el problema que trataron fue un caso real de diez restricciones duras las cuales son requerimientos diarios de cobertura para cada tipo, las enfermeras no deben de tener más de un turno por día, no exceder el número máximo de días a trabajar durante toda la programación, no exceder el máximo número de fines de semana durante toda la programación, no exceder el máximo número de turnos nocturnos durante toda la programación, solo asignar turnos nocturnos entre turnos de noche, se debe asignar dos turnos libres después de una serie de turnos nocturnos, número máximo de turnos nocturnos consecutivos, número máximo de días de trabajo consecutivos y ningún turno de tarde debe ser asignado para una enfermera en particular, todas estas restricciones acompañadas con una demanda de veintiocho días, tres turnos diarios y tres tipos de enfermeras las cuales se dividían en treinta, dos y ocho dando un total de cuarenta enfermeras, para las cuales se generaron cincuenta problemas de requerimientos y se escogieron aleatoriamente diez de estas instancias, lo que lograron con el modelo propuesto fue una disminución de la insatisfacción de entre veintitrés a treinta y seis por ciento según el análisis de los resultados dejando para investigación futura el trabajo de agregar las restricciones suaves para su optimización, las cuales son fines de semana completos, evitar días de trabajo entre días libres, respetar un mínimo de días libres después de una serie de turnos, respetar un máximo y mínimo de asignaciones de turnos matutinos y nocturnos, respetar máximo y mínimo de días de trabajo semanales, respetar número máximo de días de trabajo consecutivo para las enfermeras de tiempo parcial, evitar cierto tipo de sucesiones de turno de que al tener varias metas de forma simultánea como objetivo, generalmente tendremos varios incumplimientos de contratos es por ello que se busca alcanzar cada meta en un orden de beneficios calculados o escalares, para poder minimizar la insatisfacción de las enfermeras, enfocando el estudio en la ponderación del peso y los efectos

que tiene sobre la función objetivo al cambiarlo además de que en este problema se encontró que los tiempos máximos de procesamiento fueron 24.406 segundos.

2.7. Inteligencia artificial y modelación matemática

La inteligencia artificial siendo sistemas basados en conocimientos no se emplean con el objetivo de obtener el mejor de los resultados por lo general se tratan de una estrategia o adaptación para mejorar las soluciones conocidas realizadas por un experto en hasta un diez por ciento, utilizando heurísticas, imitación de comportamientos de la naturaleza, aleatoriedad para navegar en una o varias secciones de soluciones posibles y es debido a esta naturaleza que no se cuenta con la posibilidad de tener un certificado de calidad de la solución, pero dependiendo del enfoque se puede tratar problemas comprobados como NP-Duros en tiempos razonables y así poder conseguir soluciones factibles a problemas difíciles ya que un algoritmo exacto o modelación matemática calcula el mejor global exacto consumiendo un tiempo de búsqueda varias veces mayor a la vida del universo, para solucionar esto Scott Steve & Ron Simpson, (1998) describen una idea en la que no se incorporan resultados pero nos dicen que si se imitan los métodos manuales de los administradores de programación de enfermeras e incorporando esto a las restricciones del caso base se lograría reducir el tamaño de la búsqueda global y así poder cumplir los tiempos límites mientras se encuentran soluciones de alta calidad.

Li H., A. Lim, & B. Rodrigues, (2003) con un método híbrido de búsqueda local y satisfacción de restricciones en los problemas de hospitales reales resolvieron un sub-problema que incluyó restricciones duras de cobertura y algunas suaves llamadas reglas de preferencias tomando como primer paso relajar la satisfacción de las restricciones duras para después aplicar una técnica de búsqueda local con la finalidad de satisfacer todas las restricciones suaves posibles.

Edmund K. Burke, Patrick De Causmaecker, Greet Vanden Berghe & Hendrik Van Landeghem, (2004) Comentan que la única desventaja de los modelos matemáticos en el NRP es debido a lo enorme y complejo que es el espacio de búsqueda y que estos se basan en optimizar el valor de una sola función objetivo y es por esto que se necesita de un estudio interdisciplinario de técnicas

de búsquedas matemáticas e inteligencia artificial o basadas en experiencia, tomando en cuenta las distintas fortalezas y debilidades de cada uno de estos métodos debido a que si se tienen los conocimientos necesarios se puede caracterizar y estructurar varios modelos matemáticos que pueden trabajar algunas secciones del problema en tiempos polinomiales y sumarle una o varias heurísticas para lograr desarrollar soluciones de alta calidad en tiempos razonables.

2.8. Búsqueda en vecindario variable

Un algoritmo de ramificación y poda podría darnos el resultado exacto buscado, pero se necesita tanto poder de computo que es mejor adaptar algunos algoritmos a una heurística o metaheurística que logre checar las opciones vecinas de la solución actual en busca de soluciones factibles con mejores óptimos y es así que se logra crear enfoques distintos creando o tomando herramientas de la literatura existente como se muestra en la opinión de Edmund K. Burke, Timothy Curtois, Gerhard Post, Rong Qu & Bart Veltman, (2007) que presentaron un proceso iterativo de cambios y reparación al utilizar una heurística de ordenamiento y seguimiento para la evaluación de la calidad de la solución la cual comparan contra un programa comercial desarrollado por ORTEC usado en hospitales, para esto implementaron una heurística de ordenamiento de demanda que se enfoca en procesar primero los turnos más restringidos dándolos a la enfermera que obtiene menos penalización ante el turno en proceso, al terminar este proceso de asignaciones se obtienen soluciones infactibles en algunos casos y el paso siguiente fue utilizar una búsqueda en dos vecindarios variables donde se asignaba un turno a una enfermera distinta o se intercambiaban asignaciones entre enfermeras dándose cuenta que solo el segundo vecindario era mayormente útil aun así la mayor desventaja radicaba en tener que evaluar la solución después de cada cambio haciendo que los tiempos de proceso se extendieran en periodos mayores a una hora hasta máximo una noche, dejando una invitación a quienes quisieran intentar con este método a probar tiempos de ejecución más largos.

Para Ziran Zheng, Xiyu Liu & Xiaoju Gong, (2015) la organización de los horarios de forma automática para el personal enfermero siempre cuenta con una gran cantidad de restricciones un avance logrado por ellos es que ya se cuenta con un gran apoyo de la literatura que ayuda a

abordar este problema desde varios ángulos proponiendo un algoritmo que se basa en una búsqueda iterativa con perturbaciones para escapar de los locales óptimos pobres, esto es sobre una solución inicial aleatoria y después un operador de combinación y afinación es desarrollado para hacer mejores búsquedas iterativas de la solución, después de una cantidad límite de iteraciones sin lograr una mejora se realizan perturbaciones en búsqueda de una mejor solución, en esta búsqueda iterativa se tiene dos operadores llamados grupo de cambio donde se intercambian todas las asignaciones y el grupo de movimiento dentro de un rango de días adyacentes entre dos enfermeras seleccionadas con la finalidad de explorar distintos bloques de vecindarios de esta solución, donde el grupo de cambios es seleccionado con una alta probabilidad de aceptación con un valor del parámetro $\lambda = 0.95$ y no será cambiado para ninguna instancia aceptando cualquier solución que sea mejor o igual a la actual utilizando un contador para terminar las iteraciones cuando estas no logran mejorar la solución un número determinado de veces, y si se logra mejorar el contador volverá a cero hasta cumplirse la función de paro y así comenzar con una nueva perturbación para volver a repetir este método de mejora, este método lo aplicaron sobre sesenta instancias para experimentar divididas en tres grupos enfermeras distintos de entre treinta y cincuenta enfermeras, otros parámetros fueron el nombre de sprint track, medium track y long track, cada uno de estos grupos tiene tres sub-grupos con su respectiva dificultad para encontrar una solución, la naturaleza del algoritmo utilizado es estocástica y corre entre cincuenta, veinte y cinco veces para cada instancia dependiendo de sus tiempos límites de cada instancia, teniendo dos parámetros de frenado un basado en el tiempo límite para detener el algoritmo y la otra una condición de frenado dependiente de las iteraciones que se basa respectivamente en el tamaño de la instancia que van de seiscientas hasta más de tres millones y medio de iteraciones, los resultados muestran que se obtuvo el mejor resultado conocido en veinticuatro de veinticinco instancias Sprint, siete de quince instancias Long para algunos casos se podría haber alcanzado el mejor resultado conocido si se relajara un poco el tiempo límite de espera de la solución, demostrando que este algoritmo es eficiente y estable.

2.9. Recocido simulado

Cuando se tiene un problema difícil se recomienda utilizar el análisis de alguna búsqueda local iterativa para encontrar una solución factible, una meta-heurística como el recocido simulado es adaptable y tiene la capacidad de realizar perturbaciones en vecindarios de una solución para buscar un máximo global lo más cercano posible a los mejores valores conocidos, de no contar con estos datos se puede comparar los resultados con los de algún otro método, en la experiencia de Jaszkiwicz Andrzej, (1997) que brindó soporte computacional para el NRP en los hospitales de Polonia se debe crear un balance al dividir los turnos entre el número de enfermeras disponibles en la fuerza laboral, aplicando a su estudio varios métodos que en su mayoría son menos flexibles pero resolviendo este problema con un método de dos fases aplicando un recocido simulado combinado con un algoritmo multi-objetivo con la finalidad de crear soluciones de alta calidad logrando crear por primera vez la posibilidad de aplicar en un hospital un sistema de planeación interactiva.

Método híbrido

Inicializador con recocido simulado

En el siguiente trabajo de Ruibin Bai, Edmund K. Burke, Graham Kendall, Jingpeng Li & Barry McCollum, (2010) hablan de la importancia de desarrollar conocimiento sobre el tema y específicamente para el problema que presentan tratan de improvisar las restricciones manejándolas con la habilidad de un enfoque estándar evolutivo usando un método de categorización estocástico y explican lo difícil que es desarrollar el conocimiento para tratar un problema real de varios hospitales en Reino Unido debido a la inmensa cantidad de limitaciones y cuestiones no tan obvias, este problema se formuló y generalizó por K. A. Dowsland como un modelo de programación de enteros para su uso en varios estudios, este consiste en desarrollar soluciones semanales en turnos de día y noche para una semana con treinta enfermeras donde las enfermeras tienen tres rangos distintos y las de rango más altos pueden cubrir los rangos menores, pero mientras menos rango tienen, menos horas les son asignadas y más limitaciones de trabajo tienen. Existe la restricción de que las enfermeras trabajen en patrones de turnos específicos y la última restricción es asegurarse de que se cuenta con el suficiente número de

enfermeras para cubrir cada turno y rango requerido. Bai et. al. nos dicen que este problema puede ser abstraído como vectores binarios en catorce paneles para representar siete días y siete noches donde se presenta una decisión booleana uno para la enfermera toma el turno y cero es que no toma el turno, es a partir de esto que el recocido simulado con una hiper heurística demostró su potencial al ser adaptable a varios escenarios del NRP hibridándolo con genéticos se necesita desarrollar los criterios de aceptación y mecanismos de selección donde para este caso se utilizó una población de mil individuos cruzados de forma simple en un vector que cambia el turno seleccionado por otro turno factible con rangos de posibilidad de cruza en 0.75, rangos de mutación en 0.02, un criterio de paro que se basa en no lograr mejorar durante treinta intentos la solución óptima, siendo el modo de selección un enfrentamiento de rango aleatorio de 0.25 más la mejora actual de la solución, después de que genéticos concluye su trabajo entrega una solución inicial que se usa para iniciar el proceso del recocido simulado empezando con una temperatura que se considera intermedia porque de ser muy alta el costo computacional sería extenso, cuando la respuesta dada por el proceso genético ya es baja y solo se necesita de afinar la solución inicial con los siguientes parámetros que son criterios de aceptación de un total de nueve simples heurísticas de bajo nivel que afectan dos funciones objetivo (f, φ) con los siguientes criterios:

- 1- Cambiar el patrón de turnos de una enfermera aleatoria por otro patrón aleatorio y factible.
- 2- Similar al criterio 1, exceptuando el criterio de aceptación es buscar la primera mejora del valor φ .
- 3- Similar al criterio 1, exceptuando que el criterio de aceptación es el primero en mejorar φ y no deteriore f .

- 4- Similar al criterio 1, exceptuando que el criterio de aceptación es el primero en mejorar f .
- 5- Similar al criterio 1, exceptuando que el criterio de aceptación es el primero en mejorar f y no deteriore φ .
- 6- Intercambia el tipo de turno (por ejemplo, de día a noche y viceversa) de una enfermera aleatoria cuando la solución no está balanceada.
- 7- Esta heurística intenta generar un balance en la solución intercambiando el tipo de los patrones de asignación entre día, noche y viceversa. Cuando tanto el turno diurno y nocturno no están balanceados intercambia los turnos de dos enfermeras que tengan asignaciones de turnos diferentes.
- 8- Esta heurística intenta encontrar el primer movimiento que mejore f mediante el cambio del patrón de turnos de una enfermera aleatoria n_1 asignando a otra enfermera n_2 el patrón de asignación abandonado.
- 9- Igual que el criterio 8 pero el criterio de aceptación consiste en encontrar el primero en mejorar f sin empeorar φ .

Aplicando esta propuesta a cincuenta y dos instancias que se resolvieron veinte veces con semillas independientes y aleatorias demostrando un comportamiento similar al de otros métodos, pero mejorando en veinticuatro instancias.

Método híbrido

Escalado de colina, búsqueda tabú, recocido simulado y aceptación tardía.

Cuando se trata del NRP Diogo Santos, Pedro Fernandes, Henrique Lopes Cardoso & Eugénio Oliveirai, (2015) nos dice que este es conocido por ser NP-Duro y se necesita tanto de no violar restricciones duras como de minimizar el costo total de restricciones violadas a lo que se le puede llamar un problema de optimización, para este caso el problema que trato fue el mismo de este documento el INRC-II usando una librería de OptaPlanner por las facilidades de acceso que aporta el código abierto para utilizar otros métodos los cuales ajustaron débilmente con escalado de colina y fuertemente ajustado con búsqueda tabú, recocido simulado y aceptación tardía para encontrar soluciones factibles mientras se probaban las soluciones infactibles también, logrando la factibilidad de las soluciones en los rangos 95-100% sin lograr alcanzar los mejores globales.

2.10. Búsqueda tabú

Se recomienda combinar la búsqueda tabú con un enfoque multi-objetivo que no solo busque cambiar los días activos e inactivos de soluciones factibles sino también tomar en cuenta las restricciones dadas por el hospital cada vez que se realiza un cambio en estas soluciones, donde Edmund Burke, Patrick De Causmaecker & Greet Vanden Berghe, (1998) hablan sobre PLANE un algoritmo desarrollado en 1993 con bases en una exhaustiva investigación, su primera versión de PLANE fue implementada en los hospitales en 1995 y está siguió evolucionando con los complicados problemas de la vida real, fue desarrollado por Impakt y GET para asistir la asignación del personal del hospital de acuerdo a la demanda tomando en cuenta una lista de restricciones que limitaban estas asignaciones basándose en grupos de veinte personas con diferentes habilidades y responsabilidades como jefa de enfermeras, enfermera regular, enfermera de curaciones, estudiante y algunas de estas pueden ser reemplazadas por las enfermeras de otras categorías siempre y cuando tengan la evaluación o el grado necesario para suplir a la enfermera de otra categoría, gracias a todo este desarrollo se pudo llegar a la conclusión de que los requerimientos de los hospitales en Bélgica requieren más de métodos

donde se puedan incluir sus necesidades y no solo ser tratadas con algoritmos cíclicos, para esto desarrollaron un método con búsqueda tabú con las características de que estas heurísticas deberán estar probadas y basadas en técnicas de asignación manual donde la velocidad del cálculo es tan importante como encontrar una solución cercana al óptimo, entonces ocuparon varias fases para desarrollar este método las cuales fueron, construcción de una solución factible inicial aleatoria que considere las emergencias y los periodos previamente planeados, seguido de la búsqueda tabu donde solo se considera el movimiento de un deber de una persona a otra el mismo día, prohibiendo movimientos cuando el cambio se hace sobre una persona que no está en la categoría adecuada, donde para cada categoría se considera cada posible movimiento y el movimiento que sea el más beneficioso será asignado. Se desarrollaron dos métodos para esta búsqueda tabú híbrida: una que consumiera poco tiempo de procesamiento y otra que requería más tiempo, pero con mejores soluciones. Las restricciones duras consideradas fueron la cobertura de las enfermeras en cada categoría de todo el periodo de programación, en el problema existen treinta restricciones suaves, pero no todas fueron consideradas en el trabajo, las se consideraron son las que tienen que ver con las preferencias del personal, en el artículo proporcionan una lista no detallada de las restricciones, pero dan un link del problema para quienes estén interesados conocer cada restricción.

Unos años más tarde Edmund Burke, Peter Cowling, Patrick De Causmaecker & Greet Vanden Berghe, (2001) trataron este mismo problema con más detalle en una búsqueda tabú con genéticos y explicando que las restricciones suaves a considerar son tiempo mínimo entre dos asignaciones de acuerdo a las tareas involucradas, horas de trabajo mínimas y máximas para todo el periodo de la programación, número máximo de asignaciones para todo el periodo de la programación, trabajar fines de semana completos, trabajar de acuerdo a un patrón preferido, número máximo de asignaciones de cada tipo de tarea durante cada semana y para todo el periodo de la programación, restringir el orden en el que los turnos y días libres no serán asignados, distribuir de forma uniforme las tareas entre las personas con las mismas regulaciones de trabajo y las preferencias personales (Ad-hoc) consideradas como la requisición de días libres, celebraciones, requisición de trabajar cierto día y turno, no asistir por enfermedad, no asistir por embarazo y asignaciones secundarias a otra guardia, para esto en la solución inicial se tomaron

en cuenta varias posibilidades por individuo donde los genes seleccionados de los dos padres para el primer hijo son las mejores filas y para el segundo hijo son las mejores programaciones estos padres se generaron con las mismas técnicas aleatorias del trabajo anterior con la finalidad de poder hacer recombinaciones o diversificación, donde después de evaluar cada posible movimiento en el vecindario el mejor será tomado. La calidad de la solución es la suma de las penalizaciones en los candidatos pero la desventaja es que hacer este cálculo consume significativamente más tiempo, aun así lograron resolver cuatro instancias del mundo real de las que no se comparte la información de estas por problemas de confidencialidad pero comparten tablas de tiempos no superiores a cinco horas y todas sus mejores soluciones encontradas, recomiendan que si se utiliza este método se deberán hacer varias adaptaciones de acuerdo con las características del hospital en el que se esté trabajando.

En el trabajo de Florian Mischek & Nysret Musliu, (2016) se puede encontrar un modelo matemático detallado de cada restricción dura y suave del INRC-II, problema para el cual se describieron las características con anterioridad en este documento, ellos abordan este problema implementando una búsqueda tabú combinado con una heurística de Min-Conflict empezando desde una solución inicial distinta cada vez que no se logra improvisar la mejor solución actual esta se genera mediante asignaciones aleatorias avariciosas debido a que no encontraban mejoras resaltables si se utilizaban ordenamientos para este tipo de asignaciones las cuatro restricciones duras fueron tratadas como suaves con altos costos de violación aun así esto no aseguraba que la solución final fuera factible, utilizaron dos tipos de movimientos para mejorar la solución:

- 1- Cambios: Establecer el día libre de una enfermera en un turno y habilidad legal.
- 2- Intercambio: Entre dos enfermeras que no generen conflictos de acuerdo con sus habilidades y se hace un intercambio de las asignaciones seleccionadas en un rango (a,b) .

Mientras una solución inicial no sea encontrada solo se utilizará la opción uno por motivos de reducción de tiempo.

Para perfeccionar los parámetros que determinan el siguiente movimiento usaron un software de nombre IRACE, para implementar todos los algoritmos usaron JAVA 7 y para resolver toda la programación entera usaron el software IBM ILOG CPLEX solver, versión 12.6.3 con lo que lograron resolver este problema en los tiempos establecidos en el INRC-II logrando resultados similares a los mejores conocidos.

2.11. Búsqueda local

La búsqueda local es el pilar de otros métodos más complejos como búsqueda tabu, recocido simulado o mejoras a través de aleatoriedad avariciosa y en el trabajo de Haroldo G. Santos, Túlio A. M. Toffolo, Rafael A. M. Gomes, Sabir Ribas, (2014) donde desarrollan un método de programación entera mixta de dos fases las cuales son fase de construcción avariciosa y fase de búsqueda local utilizando como metaheurística el método de ramificación y poda para una búsqueda en vecindarios, para las soluciones iniciales el método es construir una solución a partir de la satisfacción de cada día uno por uno repitiendo este método hasta satisfacer la demanda, para la segunda parte se usó una búsqueda local que explora el espacio generado en varios vecindarios distintos pero reportando que solo dos de estos vecindarios aportan reducciones significativas a la solución, también se utilizó un software de planos cortantes (COIN-OR Foundation, CBC) para confirmar límites mínimos y máximos, con este método trataron el problema que se presentó en el INRC-I resolviendo óptimamente e improvisando las mejores soluciones conocidas con nuevas cotas mínimas de hasta un 23.7%.

Ziran Z., et al, Ziran Zheng, Xiyu Liu & Xiaoju Gong, (2015) tratan el problema presentado en the First International Nurse Rostering Competition (INRC-I), desarrollando una hiper-heurística con un intercambio avaricioso este es un algoritmo que puede considerarse una variación de un método de búsqueda de vecindarios para el cual se necesitaran tres operadores y una búsqueda local aplicando perturbaciones para tener diversidad en el espacio de búsqueda con lo que se logra escapar de los resultados locales pobres, describen el problema tratado como la cobertura de demandas con dos restricciones duras y catorce restricciones suaves listadas a continuación:

Restricciones duras

- 1- Solo se puede asignar un día a una enfermera por día.
- 2- Para cada día del periodo de programación se requieren un cierto número de turnos relacionados a la carga de trabajo.

Restricciones suaves

- 1- Máximo y mínimo número de turnos asignados a una enfermera.
- 2- Máximo y mínimo número de días consecutivos de trabajo asignados a una enfermera.
- 3- Máximo y mínimo número de días consecutivos libres.
- 4- Máximo y mínimo número de fines de semana trabajados consecutivamente.
- 5- Máximo número de fines de semana trabajados.
- 6- Asignación a fines de semana completos.
- 7- Asignación de fines de semana idénticos.
- 8- Asignación de dos días libres después de un turno nocturno.
- 9- No asignar turnos nocturnos después de un fin de semana libre.
- 10- Fines de semana completos.
- 11- Respetar los patrones de asignación no deseados.
- 12- Respetar los días libres que piden las enfermeras.
- 13- Asignar habilidades distintas
- 14- Respetar los días en los que las enfermeras piden trabajar.

El benchmark consistió de sesenta instancias en las que se logró obtener cuarenta y dos de los mejores resultados conocidos y en el resto se consiguieron cercanos al óptimo.

Sobre la implementación de métodos complejos Antoine Legrain, Hocine Bouarab & Nadia Lahrichi, (2015) nos dicen que en lo que se refiere a las asignaciones de enfermeras es una labor muy contradictoria en la que se tiene que minimizar costos de penalización mientras se maximiza la satisfacción del personal, teniendo en manos un caso de ciclos entre unidades de cuidado de planta y unidades de respaldo lo complejo fue primero entender el tema donde tuvieron que

hablar a dos hospitales de nombres Montreal el Notre-Dame y Sainte-Justine los cuales cumplen con las características de ser grandes con fuerza laboral de treinta empleados y un pequeño equipo de apoyo que absorbe las variaciones en la demanda. Después de entender el problema se tiene en cuenta con que se cuenta para la resolución de administración de tiempos de enfermeras que como ejemplos nos dan programación matemática, programación restringida, heurísticas, metaheurísticas, métodos híbridos y simulación. Teniendo en cuenta que se tienen varios turnos que se trabajaran de forma continua con descansos mínimos entre un grupo de turnos con ajustes de acuerdo a las preferencias de las enfermeras los procesos de asignación, el valor de las tareas, penalizaciones, la introducción de un modelo matemático y una heurística de todo esto se debe contar con el suficiente entendimiento para poder apegarlo a una asignación normal de turnos sin que las asignaciones afecten demasiado al problema para finalmente guardar todo en un mapa.

Para entender lo variante que es la carga de trabajo se puede usar la siguiente fórmula matemática:

$$(Demanda\ promedio) * \left(\frac{Turnos\ disponibles}{Demanda\ total} \right) * \left(\frac{Varianza}{Varianza\ promedio} \right)$$

Esto se ajusta fácilmente a otros problemas y solo se usa como un parámetro.

2.12. Algoritmos Voraces

Los algoritmos voraces, glotones o avaros son métodos veloces que cumplen un criterio a la vez minimizar o maximizar los valores de acuerdo con Cheang, H. Li, A. Lim & B. Rodrigues, (2002) la velocidad este tipo de métodos son ideales para los hospitales debido a que estos necesitan generar rápidamente los cambios en su programación sin importar si su duración es semanal, quincenal o mensual argumentando que este tipo de métodos pueden ser mejorados con la aplicación de otras heurísticas como las de la búsqueda tabú donde la calidad de las soluciones depende de la permanencia de las soluciones aceptadas en memoria teniendo en cuenta que la memoria que ocupan estas soluciones no es ilimitada y entre más soluciones se tienen en la lista

tabú de un problema difícil más crecen los tiempos de procesamiento, entonces esta técnica se debe usar con una buena estrategia o parametrizaciones para poder abordar los problemas de hospitales grandes oscilando entre las asignaciones factibles y las que sólo disminuyen el costo de la solución, donde técnicas de ordenamiento en la información del problema nos dan grandes ventajas en la velocidad de búsqueda, también comparten la modelación, distintas metodologías para resolver el NRP y la función objetivo para este caso que trata restricciones duras solo descritas como las que generalmente tienen que ver con cobertura estricta de la demanda y las restricciones suaves listadas en la sección de búsqueda local en el trabajo de Zheng et al, (2015) el objetivo de estos métodos generalmente es distribuir los recursos para satisfacer las restricciones duras con prioridad y dejar para segundo término la satisfacción de las restricciones suaves para conseguir soluciones que generalmente superan en hasta un diez por ciento las soluciones de los profesionales.

2.13. Enjambres

Al diseñar una la búsqueda de enjambres se busca mimetizar en una metaheurística el comportamiento colectivo de algunas especies de insectos, técnicas que desarrollaron a lo largo del tiempo para hacer la búsqueda colectiva de sus alimentos, este algoritmo pertenece al campo de la inteligencia de enjambre para ser precisos se utiliza ampliamente en meta-heurísticas e inteligencia computacional como herramienta de optimización para problemas de combinatoria continua y un trabajo que lleva el comportamiento de las abejas a solucionar el NRP que se presenta en el INRC-I descrito con anterioridad en la sección de búsqueda local es el de Rajeswari, J. Amudhavel, Sujatha Pothula & P. Dhavachelvan, (2017) la colmena envía exploradores estocásticamente para localizar caminos de flores y si las abejas realizan un hallazgo vuelven y reclutan a otras abejas que harán la recolección local hasta agotar y abandonar este camino de flores, mientras otro pequeño número de abejas siguen buscando otros caminos globalmente, con la siguiente descripción general del procedimiento:

- 1- El reclutamiento que genera una población de agentes para muestrear el espacio de soluciones vacías.
- 2- La búsqueda global es mandar una fracción de la población exploradora usando el método de ruleta rusa para el posicionamiento en distintas regiones donde se evalúan todas las posibles construcciones con una búsqueda local, repitiendo este paso se repite hasta encontrar una solución aceptable o se cumpla la condición de paro.
- 3- La búsqueda local que se realiza dentro de la búsqueda global toma los exploradores más exitosos de acuerdo a la función objetivo y reclutan un número variable de la población inactiva para hacer una búsqueda en la proximidad de las soluciones más adecuadas.
- 4- Durante la duración de la búsqueda local se lleva a cabo una reducción progresiva del vecindario siempre que se falle en la improvisación de una búsqueda local entera.
- 5- El abandono de una región es la interrupción de la búsqueda después de un número de estancamientos consecutivos para reiniciar la búsqueda en un sitio elegido al azar.

Teniendo como objetivo principal al desarrollar la función multi-objetivo satisfacer lo siguiente:

- a- Minimizar el costo total del NRP.
- b- Satisfacer todas las restricciones duras.
- c- Satisfacer tantas restricciones suaves como se pueda.
- d- Fortalecer recursos utilizados.
- e- Distribuir equitativamente las cargas de trabajo.

Con este método lograron alcanzar los mejores valores en 34 de 69 instancias.

2.14. Coloreo de grafos

Si vemos este problema de la forma en la que lo ve un profesional de la salud Masoud Rabbani y Mehrdad Niyazi, (2017) nos dicen que desde 1966 al 2011 se necesitó la formulación de problemas reales a través de aprendizaje y desarrollo de técnicas para recolección de datos muy sencillos que puedan ser utilizados en este problema, gracias a esto se comprendió que la complejidad radica en la toma de decisiones como quien tomará el turno activo, quien descansará, cuales son las políticas del hospital y las reglas de la región, toda esta información generalmente se transforma en un problema de optimización en el cual el objetivo es mejorar los tiempos o atractivo de una solución, donde como ejemplo podemos tener una solución no económica pero que respete todas las restricciones de penalización o en casos duros encontrar la factibilidad minimizando o maximizando violaciones suaves, donde al repetir estos ciclos de retroalimentación y adaptación de un problema real es más fácil agregar casos generales como las emergencias o pedidos exclusivos de algún hospital como la forma de convivir profesionalmente del personal. En el modelo para este problema tenemos tres turnos por día, cargas de trabajo por enfermeras de cuarenta y cinco horas, exceptuando a la jefa de enfermeras las demás enfermeras solo deberán trabajar un turno por día, las enfermeras deberán trabajar al menos cinco noches durante toda la solución de no ser así deberán trabajar al menos dos noches con tres tardes por semana, ninguna enfermera trabaja por las mañanas después de haber trabajado dos turnos de tarde seguidos, las enfermeras tienen al menos un día libre a la semana y si trabajo las cinco noches seguidas serán dos. Con el objetivo de minimizar la cantidad de outsourcing contratado tomando un enfoque de coloreo de grafos (grafos bipartitos) para improvisar el resultado de la relación entre enfermeras. La asignación de enfermeras es conocida por ser NP-Duro y los experimentos fueron desde treinta hasta cincuenta enfermeras, obteniendo resultados eficientes al buscar no asignar enfermeras con malas relaciones entre sí a los mismos turnos mientras las enfermeras en entrenamiento solo eran asignadas a turnos con jefas de enfermeras en ellos.

2.15. Sistemas expertos y conocimiento basado en sistemas

Estos intentan solucionar el problema ofreciendo una solución en tiempos razonables y de buena calidad, pero se necesita desarrollar con el conocimiento de poder mimetizar el trabajo manual de una enfermera experta en asignación de turnos de enfermeras.

Método híbrido

Pre-procesamiento con búsqueda tabú y avaricioso.

Sanja Petrovic & Greet Vanden Berghe, (2010) debido a que solo se ha tenido un grado de asertividad en algunos problemas del NRP y a pesar de las décadas de investigación desarrollados para este problema es que las instituciones de la salud prefieren seguir su trabajo de asignación manualmente debido a que los ambientes suelen ser muy distintos entre hospitales y cada año que pasa tiene aún más restricciones, es por ello que pensaron que la consistencia de todo lo que se sabían hasta ese año era baja debido a que solo se contaban con poco más de dos enfoques para resolver el NRP desarrollados siempre por los mismos autores, es por esto que investigaciones más recientes ofreciendo datos de instancias reales, estas pueden encontrarse en <http://www.cs.nott.ac.uk/~tec/NRP/index.html> adquiriendo los permisos necesarios para acceder a ellos estos ofrecen un par de solucionadores con una interface y graficas muy interesantes además incluye los mejores resultados conocidos para esas instancias.

Otra página es <http://www.projectmanagement.ugent.be/nsp.php> donde permiten el análisis de diferentes enfoques del NRP, presentándonos un método que se basa en un problema del hospital Ophthalmology ward at Queen's Medical Centre (QMC) y para este caso nos dicen que el componente principal del enfoque desarrollado es el de una meta-heurística basada en búsqueda tabú donde iterativamente se busca en muchos vecindarios, formando el núcleo de este método, donde antes de la fase de optimización se necesita construir un evaluador de la factibilidad del vecindario que se está generando para así lograr la solución de las restricciones duras, con la opción de congelar partes del tablero se previene que algunas vecindades puedan ser modificadas aunque todo esto vuelve más compleja la parte de la optimización siempre se tiene como meta que las soluciones sean factibles entonces el siguiente paso es dividir el problema en sub-problemas donde se buscará tanto la cobertura mínima como la deseada y si al

final se obtiene una solución factible esta será tomada, un ejemplo de cómo deben ser los vecindarios es:

- 1.- El intercambio de algunos días consecutivos se lleva entre la enfermera con el peor horario y alguna aleatoria.
- 2.- El intercambio avaricioso en el vecindario se aplica a cada pareja de enfermeras.
- 3.- El vecindario que considera los fines de semana tendrá menos prioridad de ser seleccionado en comparación de los demás vecindarios.

La exploración en el espacio de búsqueda se moverá de un vecindario a otro cuando se alcance el criterio de parada del vecindario actual. Para después del pre-procesamiento utilizar una búsqueda tabú en estos problemas que se tienen de treinta a treinta y cinco enfermeras en guardia, con un tablero no cíclico se utiliza un método totalmente nuevo para el que se ocupó de una rigurosa experimentación y conocimientos que sirvió para el modelado de vecindarios a partir de la experiencia de administradores profesionales en el que las enfermeras pueden expresar sus necesidades, este proceso consiste en la reparación de los tableros con una lista de parámetros requeridos donde se listan las violaciones causadas durante el proceso de reparación, pero que este equipo de profesionales al desarrollar la solución del problema manualmente violaron muy pocas restricciones, para lo cual describen tres tipos de reparaciones a utilizar:

- Reasignación: Asignar un turno a una enfermera.
- Intercambio: Intercambiar turnos entre enfermeras en un día en particular.
- Cambio: intercambiar la asignación de dos enfermeras en dos días.

Esto funciona con base en el almacenamiento de casos similares donde se comete una violación y la reparación de esta se hace en dos fases, en la fase uno se generaliza la violación para un caso similar en términos de violación y parámetros de casos base, después se pasa a la segunda fase donde se busca la distancia entre las características las cuales se dividen en tres grupos:

- 1.- Las características dadas proveen información estadística sobre el tablero.
- 2.- Las características que dan la información de cobertura.
- 3.- El turno modela las características de las asignaciones de las enfermeras vecinas del día que envuelve la restricción.

Para el equipo de Martin Stølevik, Tomas Eric Nordlander, Atle Riise & Helle Frøyseth, (2011) es importante producir programaciones de alta calidad, trabajo que no es sencillo debido a la complejidad del problema que engloban las demandas cambiantes, políticas del hospital, leyes de trabajo y regulaciones de la región, la satisfacción de empleados, empleadores y pacientes, el problema que resolvieron contiene siete restricciones duras y ocho suaves las cuales son:

Duras

- 1- Asignar máximo un turno cada día a cada empleado.
- 2- Cubrir toda la demanda laboral especificada.
- 3- Las horas de trabajo que sumen cada empleado no deben superar el contrato asignado.
- 4- Los empleados solo pueden ser asignados a horarios que demanden sus habilidades.
- 5- Respetar el tiempo mínimo de descanso en días trabajados consecutivamente.
- 6- Cada semana debe tener un periodo mínimo libre.
- 7- Respetar el tiempo máximo permitido de trabajo semanal.

Suaves

- 1- Se debe evitar asignar demasiados turnos consecutivos del mismo tipo.
- 2- Se debe evitar asignar demasiados días consecutivos de trabajo.
- 3- Se debe evitar asignar pocos turnos consecutivos del mismo tipo.
- 4- Se debe evitar asignar pocos días consecutivos de trabajo.
- 5- Minimizar la desviación del mínimo y máximo número de turnos en cada categoría.
- 6- Minimizar la desviación del tiempo contratado.
- 7- Agrupar lo más posible, los días de descanso.
- 8- Maximizar el número de patrones deseados.
- 9- Minimizar el número de patrones no deseados.

Utilizaron un software para resolver problemas de NRP su nombre es SINTEFs SCOOP library que trabaja con modelos genéticos y sirve para resolver varios tipos de este contiene la librería CP/CSP library que sirve para hacer búsquedas locales para primero satisfacer las restricciones duras iterando entre vecindarios variables. En la segunda fase se agrega la destrucción de partes de la solución para volver a construirla con el objetivo de disminuir por secciones las violaciones, los cambios permitidos están en la siguiente lista:

- a- No permitir movimientos que violaran las restricciones duras.
- b- La cobertura debe ser obedecida permitiendo solo intercambios verticales entre enfermeras.
- c- En cada iteración se aplica un movimiento que cause un decremento en la penalización.
- d- Buscar los vecindarios en secuencia.
- e- Concentración en el área problemática de la solución actual.

Los resultados que lograron fue resolver problemas reales grandes en tiempo razonable, estos contenían ochenta empleados con programaciones que duraban ciento sesenta y ocho días con hasta nueve tipos de turnos distintos en tres categorías distintas.

Capítulo 3.

3. Origen del INRC-II

La organización que creó los concursos del INRC-I en 2010 y el INRC-II del 2014 es el Practice and Theory of Automated Timetabling (PATAT) esta información se obtuvo del siguiente URL: <https://www.patatconference.org/communityService.html>.

El problema que abordamos en esta tesis es el INRC-II que inicio el 17 de octubre del 2014 donde los mejores resultados y sus métodos de solución se dieron a conocer el 1 de junio 2015. El INRC-II es un problema que se reconoce y está registrado con una complejidad NP-Duro, el concurso asociado a este problema tuvo dos fases una de resolver un conjunto A de instancias, en la siguiente se utilizó otro Benchmark de instancias seleccionadas por los organizadores. Las instancias utilizadas en este concurso será el Benchmark que utilizaremos en este trabajo y su resolución consiste en satisfacer la demanda total las cuales están restringidas con cuatro restricciones duras y se debe optimizar la penalización total de siete restricciones suaves tomando en cuenta el historial y el escenario.

3.1. Descripción general de escenario, historial y demanda.

El escenario contiene la información de la fuerza laboral que se tiene para satisfacer la demanda de una instancia incluyendo características como duración del horizonte de la instancia (de 4 a 8 semanas), número de enfermeras, contratos, habilidades, límites de las asignaciones de turnos y sus prohibiciones.

La demanda se compone de varias semanas que se pueden dividir en requerimientos mínimos y requerimientos óptimos de cobertura para un día, si no se logra satisfacer la demanda mínima la solución será infactible.

El historial contiene toda la información previa a la semana que se esté solucionando con el objetivo de poder conservar la factibilidad entre semanas debido a que si la semana empieza en

un lunes y no se toma en cuenta lo que paso el domingo de la semana pasada puede surgir el caso en el que la asignación sin memoria puede ser factible cada semana, pero infactible al concatenar las semanas restantes, debido a la restricción dura de sucesiones de turnos prohibidas, que se explica más adelante. Debido a esto es importante tomar en cuenta el historial generado por cada semana para su uso futuro durante las asignaciones de las semanas faltantes.

3.2. Restricciones duras

H1 Cada enfermera puede cubrir máximo un turno por día:

Un día esta compuesto de cuatro turnos distintos para cada enfermera, estas solo pueden ser asignadas una vez cada día donde en caso de asignar dos turnos o más a una enfermera durante el mismo día esto volvería infactible la solución que se esta calculando.

Para el siguiente ejemplo se usará 1 para la asignación de un turno en un día y 0 para los turnos libres.

3.1 Ejemplo de asignación de una enfermera en una semana, cuando se asigna más de un turno al día estan marcadas en un recuadro rojo para indicar que se está violando la restricción

Turno	L	Ma	Mi	J	V	S	D
Matutino	1	0	0	0	1	1	0
Diurno	1	1	0	0	0	0	0
Vespertino	0	0	0	1	0	0	1
Nocturno	0	0	1	0	1	0	0

H2 Cubrir la demanda mínima de enfermeras en la instancia:

Una instancia esta compuesta de cuatro u ocho semanas con demandas de asignaciones distintas o similares, esta se divide en el día que se solicita la demanda y cada día se divide en una cantidad de turnos que se determina en el archivo escenario donde para cada uno de estos turnos se tendrá una demanda mínima que necesita ser cubierta .La demanda mínima sea cubierta en su totalidad de lo contrario la solución final sera infactible.

3.2.A Ejemplo de demanda mínima

Turno	Demanda mínima						
	L	Ma	Mi	J	V	S	D
Matutino	1	1	3	1	2	1	3
Diurno	1	2	2	1	1	1	2
Vespertino	1	1	2	2	2	1	1
Nocturno	1	3	3	1	1	1	2

3.2.B Ejemplo de violación de la demanda mínima, los días de demanda que violan la asignación mínima estan marcadas en un recuadro rojo.

Turno	Asignación de enfermeras						
	L	Ma	Mi	J	V	S	D
Matutino	1	1	2	1	2	1	2
Diurno	1	2	2	1	1	1	2
Vespertino	1	1	2	2	2	1	1
Nocturno	1	1	1	1	1	1	2

H3 Las asignaciones continuas deberán respetar las condiciones de asignación legales.

La asignación de algunos turnos tienen distintos tipos de sucesiones prohibidas, es decir, cada instancia puede tener un conjunto diferente de sucesiones de turnos prohibidos, además esta restricción se tiene que tener en cuenta no solo en cada semana, si no

tambien cuando se concatenan todas las semanas de la instancia, esto hace que las asignaciones que suceden en domingo afecten las asignaciones que se permiten los lunes.

3.3 Tabla de sucesiones prohibidas en una instancia

Sucesiones prohibidas				
Matutino	0	-	-	-
Diurno	1	Matutino	-	-
Vespertino	2	Matutino	Diurno	-
Nocturno	3	Matutino	Diurno	Vespertino

3.4 Ejemplo de asignaciones de una enfermera, los turnos que violan las sucesiones de turnos estan marcadas en un recuadro rojo.

Turno	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D
Matutino	1	1	0	0	0	0	0	1	0	0	0	1	1	0
Diurno	0	0	0	0	0	0	0	0	1	0	0	0	0	0
Vespertino	0	0	0	1	0	0	1	0	0	0	0	0	0	1
Nocturno	0	0	1	0	0	0	0	0	0	0	0	0	0	0

En la tabla del ejemplo 3.4 se puede apreciar que después de la programación de una enfermera la asignación del primer jueves en el turno Vespertino viola esta restricción ya que el primer miercoles hay una asignación en el turno Nocturno y este tiene prohibidas las sucesiones Matutino, Diurno y Vespertino de igual forma el segundo lunes se tiene una asignación en el turno Matutino que esta violando las sucesiones permitidas para el turno asignado en el primer domingo volviendo infactible este ejemplo de programación de enfermeras.

H4 La enfermera asignada deberá contar con la habilidad solicitada en la demanda.

Los turnos de la demanda tienen requerimientos de cobertura se describe la habilidad que la enfermera debe tener para cubrir el turno solicitado y las enfermeras pueden tener un mínimo de una habilidad hasta un máximo de tres habilidades.

Matutino Jefa de enfermeras, Diurno Enfermera, Vespertino Cuidados, Nocturno Entrenamiento.

3.3. Restricciones suaves

S1 Cobertura óptima de la demanda (30).

La restricción H1 tiene las mismas características de esta restricción pero la diferencia principal es que en la S1 no se busca una cobertura estricta, esto da la flexibilidad de tomar decisiones entre buscar factibilidad o una mejor solución.

3.5.A Ejemplo con de demanda mínima y óptima

Turno	Demanda – Inicial													
	L		Ma		Mi		J		V		S		D	
	Mín	Opt	Mín	Opt	Mín	Opt	Mín	Opt	Mín	Opt	Mín	Opt	Mín	Opt
Matutino	1	3	1	1	3	4	1	3	2	3	1	3	3	4
Diurno	1	3	2	0	2	1	1	5	1	0	1	2	2	4
Vespertino	1	0	1	4	2	0	2	3	2	1	1	1	1	1
Nocturno	1	2	3	4	3	4	1	0	1	4	1	2	2	3

3.5.B Asignación de enfermeras cubriendo la demanda mínima y violando en algunos casos la demanda óptima, donde se viola la demanda óptima se marca en recuadro rojo.

Turno	Asignación de enfermeras						
	L	Ma	Mi	J	V	S	D
Matutino	1	1	3	1	2	1	3
Diurno	1	2	2	1	1	1	2
Vespertino	1	1	2	2	2	1	1
Nocturno	1	3	3	1	1	1	2

En la tabla del ejemplo 3.5.B se puede apreciar que después de la programación de las enfermera el turno Matutino viola las asignaciones óptimas en miércoles y domingo en un total de dos ocasiones, el turno Diurno y Vespertino satisfacen la demanda óptima y el turno Nocturno no satisface lo óptimo en los días martes y miércoles cada uno en una ocasión, el total de violaciones acumuladas es 4 con un peso de 30 cada violación se obtiene que el peso de esta programación es 120.

S2 S2a. Respetar límites mínimos y máximos de turnos consecutivos asignados (15).

Este dato se encuentra en el archivo escenario en la sección de tipos de turno, este se aplica a las asignaciones de cada enfermera y se viola por cada turno consecutivo mínimo faltante y cada máximo excedido como se ilustra en la siguiente tabla.

3.6 Tabla escenario sección SHIFT_TYPE.

Tipo de turnos = 4		
-	Mín	Máx
Matutino	2	5
Diurno	2	28
Vespertino	3	5
Nocturno	4	5

3.7 Ejemplo de tabla de asignación de turnos de una enfermera, las violaciones están marcadas en un recuadro rojo.

Turno	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D
Matutino	1	0	0	0	0	0	0	0	0	0	0	0	0	0
Diurno	0	0	1	1	1	0	0	0	0	0	0	0	0	0
Vespertino	0	0	0	0	0	1	1	1	1	1	1	1	0	0
Nocturno	0	0	0	0	0	0	0	0	0	0	0	0	0	0

En la tabla 3.7 se tiene una relación con los datos de la tabla 3.6 se puede apreciar que se viola el número mínimo de turnos Matutino en uno, ya que sólo hay un día asignado en ese turno y deberían ser mínimo dos. Para el turno Vespertino se viola dos veces el máximo número de turnos permitidos, ya que sólo se permiten máximo cinco días seguidos con ese turno y hay siete. El turno Diurno respeta el número de asignaciones máximas y mínimas, de igual forma el turno nocturno no presenta violaciones al no ser utilizado, el total de violaciones acumuladas es 3 con un peso de 15 cada violación se obtiene que el peso de esta programación es 45.

S2b Respetar límites mínimos y máximos de días consecutivos de trabajo asignados (30).

Existe información en las instancias que nos entrega el archivo escenario en la sección de los contratos este se aplica a cada enfermera en esta sección podemos encontrar el dato de días consecutivos de trabajo mínimos y máximos que nos penalizará por cada faltante o cada vez que se excede el máximo en las asignaciones, hay dos tipos de duración cuatro y ocho semanas cada uno de estos tipos tiene dos instancias del mismo tamaño, sus respectivos tamaños son 30, 40, 50, 60, 80, 100 y 120 enfermeras esto da un total de veintiocho instancias y catorce escenario con su respectivo conjunto de contratos, los cuales tienen un tamaño mínimo de tres contratos y un tamaño máximo de cuatro contratos, la tabla 3.8 muestra un ejemplo de la información de contratos.

3.8 Tabla escenario sección CONTRACTS.

Contratos=4	Turnos globales		Días consecutivos de trabajo		Días consecutivos de descanso		Asignación fines de semana	Fines de semana juntos
	Mín	Máx	Mín	Máx	Mín	Máx	Máx	
Tiempo completo	15	22	3	5	2	3	2	1
Tiempo parcial	7	15	4	7	2	5	2	1
Medio tiempo	7	11	4	6	3	5	1	0
Veinte porciento	5	8	3	4	4	7	1	1

3.9 Ejemplo de tabla de asignación de días de trabajo para enfermeras con contratos distintos, las violaciones están marcadas en un recuadro rojo.

Turno	Contratos	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D
HN_1	Tiempo completo	1	1	0	0	0	1	1	1	1	1	1	0	0	0
NU_1	Tiempo parcial	1	1	1	1	1	1	1	1	0	0	1	1	1	1
CT_1	Medio tiempo	1	1	1	1	0	0	0	1	1	1	1	1	1	0
TR_1	Veinte porciento	1	1	1	0	0	0	0	1	1	1	1	0	0	0
NU_2	Tiempo completo	0	0	0	1	1	1	1	1	0	0	0	1	1	1
CT_2	Medio tiempo	0	0	0	0	0	1	1	1	1	1	0	0	0	0
TR_2	Medio tiempo	1	1	1	0	0	0	0	1	1	1	0	0	0	0

En la tabla 3.9 se tiene una relación con los datos de la tabla 3.8 con esta información se puede apreciar en la enfermera HN_1 con contrato Tiempo completo que al tener dos asignaciones seguidas de trabajo viola el mínimo de tres días de trabajo continuos en una ocasión. El máximo de días de trabajo continuo tambien se viola en las asignaciones de la enfermera HN_1 al tener seis asignaciones continuas se viola el máximo de cinco días de trabajo consecutivo en una ocasión. La enfermera NU_1 con contrato Tiempo parcial solo tiene una violación al exceder el límite máximo de días continuos del contrato en uno. Las enfermeras CT_1 con contrato Medio tiempo, TR_1 con contrato de Veinte porciento, UN_2 con contrato Tiempo completo y CT_2 con contrato Medio tiempo tienen asignaciones continuas que respetan los rangos mínimos y máximos, la enfermera TR_2 con contrato

Medio tiempo tiene dos asignaciones de tres días las cuales violan el mínimo de cuatro días continuos de trabajo, el total de violaciones acumuladas es 5 con un peso de 30 cada violación se obtiene que el peso de esta programación es 150.

S3 Respetar los límites mínimos y máximos de días consecutivos de descanso (30).

De forma similar a la restricción S2, esta información existe en las instancias que nos entrega el archivo escenario en la sección de los contratos como se observa en la tabla 3.8, y se viola por cada día faltante de descanso u cada máximo excedido.

3.10 Ejemplo de tabla de asignación de días de descanso de enfermeras con contratos distintos, las violaciones están marcadas en un recuadro rojo.

Turno	Contratos	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D
HN_1	Tiempo completo	1	1	1	1	0	1	1	1	1	1	0	0	0	0
NU_1	Tiempo parcial	1	1	1	1	1	1	1	0	0	0	1	1	1	1
CT_1	Medio tiempo	1	1	1	1	0	0	0	1	1	1	1	1	1	0
TR_1	Veinte por ciento	1	1	1	0	0	0	0	1	1	1	1	0	0	0

En la tabla 3.10 se tiene una relación con los datos de la tabla 3.8 con esta información se puede apreciar que en la enfermera HN_1 con contrato Tiempo completo al tener un día de descanso se viola el mínimo de dos días consecutivos en una ocasión. El máximo de días de descanso consecutivo también se viola en la enfermera HN_1 al tener cuatro asignaciones seguidas viola el máximo de tres días de descanso consecutivo en una ocasión. Las enfermeras NU_1 con contrato Tiempo parcial, CT_1 con contrato Medio tiempo y la enfermera TR_1 con contrato de Veinte por ciento tienen asignaciones continuas que respetan los rangos mínimos y máximos, el total de violaciones acumuladas es 2 con un peso de 30 cada violación se obtiene que el peso de esta programación es 60.

S4 Preferencia de día o turno libre (10).

Esta información se encuentra en el archivo datos semanales en la sección de los requisición de turnos libres como se observa en la tabla 3.11, y se viola sólo cuando la

enfermera es asignada a uno de los turnos que seleccionó como libres, el turno ANY se refiere a un día completamente libre de asignaciones.

3.11 Tabla datos semanales sección SHIFT_OFF_REQUESTS.

Requisición de días libres = 11		
CT_19	Any	Mar
CT_21	Vespertino	Mar
HN_2	Matutino	Mar
NU_13	Any	Mie
CT_22	Vespertino	Jue
CT_24	Any	Jue
NU_11	Any	Jue
NU_14	Any	Jue
NU_16	Diurno	Jue
NU_7	Diurno	Jue
NU_9	Nocturno	Vie

3.12 Tabla de definición de los turnos en la programación.

Turnos	
Matutino	M
Diurno	D
Vespertino	V
Nocturno	N
Libre	-

3.13 Ejemplo de tabla de asignación de turnos de enfermeras para requisición de día o turno libre en una programación de cuatro semanas, las violaciones están marcadas en un recuadro rojo.

	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D
CT_19	M	M	M	D	D	-	V	V	V	-	-	-	N	N	-	-	M	-	M	-	-	-	M	-	-	-	D	-
CT_21	D	D	D	D	V	V	V	-	-	-	-	-	-	N	N	N	N	-	-	-	-	-	-	-	-	M	M	M
HN_2	V	V	-	-	-	-	-	M	M	M	N	N	-	-	D	D	D	V	V	-	-	V	V	N	N	N	-	-
NU_13	N	N	-	-	-	-	M	D	-	-	-	M	M	D	D	-	-	-	-	-	-	V	N	-	M	D	V	-

En la tabla 3.13 se tiene una relación con los datos de la tabla 3.11 con esta información se puede apreciar en la enfermera CT_19 tres violaciones debido a que pidió que los martes no se le asignará ningún turno. La enfermera CT_21 no tiene ninguna violación de su requisición de días libres. La enfermera HN_2 tiene una violación de su contrato al haberle asignado un martes en un turno Matutino y por último la enfermera UN_13 tiene una violación al haberle asignado cualquier turno un miércoles, el total de violaciones acumuladas es 5 con un peso de 10 cada violación se obtiene que el peso de esta programación es 50.

S5 Trabajar completo el fin de semana (sábado y domingo) si esta restricción esta activa (30). De forma similar a la restricción S2, esta información se encuentra en el archivo escenario en la sección de los contratos-fines de semana juntos, y se viola solo cuando esta booleana esta activa en uno, cero cuando no esta activa, la siguiente tabla muestra esta violación.

3.14 Ejemplo de tabla de asignación de fines de semana completos de enfermeras con contratos distintos, las violaciones están marcadas en un recuadro rojo.

Turno	Contratos	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D
HN_1	Tiempo completo	0	1	1	1	0	0	1	1	1	1	0	0	1	1
NU_1	Tiempo parcial	0	1	1	1	1	1	1	0	0	1	1	1	0	0
CT_1	Medio tiempo	0	0	1	1	1	1	0	0	0	1	1	1	1	0
TR_1	Veinte porciento	1	1	1	0	0	0	0	1	1	1	1	0	0	0

En la tabla 3.14 se tiene una relación con los datos de la tabla 3.8 con esta información se puede apreciar que en la enfermera HN_1 con contrato Tiempo completo los fines de semana

completos estan activos debido a esto esta restricci3n se viola en una ocasi3n. En la enfermera NU_1 con contrato Tiempo parcial los fines de semana completos estan activos pero no se viola esta restricci3n al tener un fin de semana asignado por completo en s3bado y domingo y el otro fin de semana sin ninguna asignaci3n en s3bado y domingo. En la enfermera CT_1 con contrato Medio tiempo no se viola esta restricci3n al estar desactivada, En la enfermera TR_1 con contrato Veinte porciento esta restricci3n esta activa pero no se viola al tener ambos fines de semana no asignados, el total de violaciones acumuladas es 1 con un peso de 30 cada violaci3n se optiene que el peso de esta programaci3n es 30.

S6 Respetar los l3mites minimos y m3ximos de d3as trabajados de forma global (20).

De forma similar a la restricci3n S2, esta informaci3n existe en las instancias que nos entrega el archivo escenario en la secci3n de los contratos-d3as globales como se observa en la tabla 3.8, y por cada d3a faltante de asignaci3n m3nima u cada m3ximo excedido globalmente se viola una ocasi3n, esta restricci3n evalua solo cuando se conoce la soluci3n de toda la programaci3n de una demanda.

3.15.A Ejemplo de tabla de contratos distintos para la tabla 3.15.B.

Turno	Contratos
HN_1	Tiempo completo
NU_1	Tiempo parcial
CT_1	Medio tiempo
TR_1	Veinte porciento
NU_2	Tiempo completo
CT_2	Medio tiempo
TR_2	Medio tiempo

3.15.B Ejemplo de tabla de asignación de turnos de enfermeras para días globales de una programación de cuatro semanas, las violaciones están marcadas en un recuadro rojo.

	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D	Total de asignaciones
HN_1	1	1	1	1	1	0	1	1	1	0	0	0	1	1	0	0	1	0	1	0	0	0	1	0	0	1	0	14	
NU_1	1	1	1	1	1	1	1	0	0	0	0	0	0	1	1	1	1	1	0	0	0	0	0	0	1	1	1	14	
CT_1	1	1	0	0	0	0	1	1	1	1	1	1	0	0	1	1	1	1	1	0	0	1	1	1	1	0	0	17	
TR_1	0	0	0	0	0	1	1	0	0	0	1	1	1	1	0	0	0	0	0	1	1	0	1	1	1	0	0	11	
NU_2	0	0	0	1	1	1	1	1	0	0	0	1	1	1	0	0	0	1	1	1	1	1	0	0	0	1	1	16	
CT_2	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	0	0	10	
TR_2	1	1	1	0	0	0	1	1	1	0	0	0	0	1	1	1	0	0	0	0	0	1	1	1	0	0	0	12	

En las tablas 3.15.A y 3.15.B se tiene una relación con los datos de la tabla 3.8 con esta información se puede apreciar en la enfermera HN_1 con contrato Tiempo completo que se tiene el límite de quince días de trabajo mínimos durante toda la programación y se asignaron catorce entonces se viola en una ocasión la restricción. La enfermera NU_1 con contrato Tiempo parcial respeta los límites mínimos y máximos globales de la programación de cuatro semanas. La enfermera CT_1 con contrato Medio tiempo tiene límites de once días de trabajo máximos durante toda la programación y se asignaron diecisiete lo que viola en seis ocasiones la restricción. La enfermera TR_1 con contrato Veinte por ciento tiene límite de ocho días de trabajo máximos durante toda la programación y se asignaron once lo que viola en tres ocasiones la restricción. Las enfermeras NU_2 con contrato Tiempo completo y CT_2 con contrato Medio tiempo respetan los límites mínimos y máximos de asignaciones globales. La enfermera TR_2 con contrato Medio tiempo viola el máximo de asignaciones globales es una ocasión el total de violaciones acumuladas es 11 con un peso de 20 cada violación se obtiene que el peso de esta programación es 220.

S7 Respetar el límite máximo de fines de semana trabajados (30).

De forma similar a la restricción S2, esta información existe en las instancias que nos entrega el archivo escenario en la sección de los contratos-asignación máxima de fines de

semana como se observa en la tabla 3.8, y se viola por cada tupla de fin de semana que sobrepase el máximo de asignaciones.

3.16 Ejemplo de tabla de asignación de días de enfermeras para fines de semana mínimos y máximos en una programación de cuatro semanas, las violaciones están marcadas en un recuadro rojo.

	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D	L	Ma	Mi	J	V	S	D
HN_1	1	1	1	1	1	0	1	1	1	0	0	0	1	1	0	0	1	0	1	0	0	0	1	0	0	0	1	0
NU_1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	0	0	0	0	0	0	1	1	1	1
CT_1	1	1	0	0	0	0	0	1	1	1	1	1	0	0	1	1	1	1	1	0	0	1	1	1	1	0	0	0
TR_1	0	0	0	0	0	0	1	1	0	0	0	1	1	1	0	0	0	0	0	1	1	0	1	1	0	0	0	

En la tabla 3.16 se tiene una relación con los datos de la tabla 3.8 esta información nos describe que cuando se asigna una enfermera a alguno o a los dos días del fin de semana se cuenta una sola vez y como se puede apreciar en la enfermera HN_1 con contrato Tiempo completo que se tiene el límite de dos fines de semana máximos para trabajar durante toda la programación y se asignaron tres lo que viola en una ocasión la restricción. La enfermera NU_1 con contrato Tiempo parcial respeta el límite máximo de dos asignaciones a fines de semana de la programación de cuatro semanas. La enfermera CT_1 con contrato Medio tiempo respeta el límite máximo de un fin de semana de trabajo durante toda la programación. La enfermera TR_1 con contrato Veinte por ciento tiene límite de un fin de semana de trabajo durante toda la programación y se asignaron tres lo que viola en dos ocasiones la restricción, el total de violaciones acumuladas es 3 con un peso de 30 cada violación se obtiene que el peso de esta programación es 90.

3.4. El formato de los archivos

3.4.1. Escenario

En la primera línea el nombre del archivo, se basa en la cantidad de enfermeras y número de semanas que componen esta instancia (Ejemplo SCENARIO = n030w4).

En la segunda línea se tiene el encabezado WEEKS = 4 donde hay que seleccionar de la cadena el número **W** (4 en este caso).

En la siguiente línea se tiene En la segunda línea se tiene el encabezado SKILLS = 4 donde hay que seleccionar de la cadena el número **SK** de habilidades diferentes que estarán presentes en esta instancia, en este caso 4, luego en las siguientes **SK** líneas estará el nombre de las habilidades (en este ejemplo: HeadNurse, Nurse, Caretaker, Trainee).

En la siguiente línea se tiene el encabezado SHIFT_TYPES = 4 donde hay que seleccionar de la cadena el número **T** de turnos diferentes que estarán presentes en esta instancia (para todas las instancias de este problema son cuatro los turnos), luego en las siguientes **T** líneas estará el nombre de los turnos con el siguiente formato:

$$\begin{aligned} &turno_1 (t_{min_1}, t_{max_1}) \\ &turno_2 (t_{min_2}, t_{max_2}) \\ &\dots \\ &turno_t (t_{min_t}, t_{max_t}) \end{aligned}$$

En la siguiente línea se tiene el encabezado FORBIDDEN_SHIFT_TYPES_SUCCESSIONS y luego habrá **T** líneas con el siguiente formato:

$$\begin{aligned} &turno_{u_1} t_{u_1} Uturno_{1_1} Uturno_{1_2} \dots Uturno_{1_{t,u}} \\ &turno_{u_1} t_{u_1} Uturno_{2_1} Uturno_{2_2} \dots Uturno_{2_{t,u}} \\ &\dots \\ &turno_{u_t} t_{u_t} Uturno_{t_1} Uturno_{t_2} \dots Uturno_{t_{t,u}} \end{aligned}$$

En la siguiente línea se tiene el encabezado CONTRACTS = 4 donde hay que seleccionar de la cadena el número **E** de contratos diferentes que estarán presentes en esta instancia y luego habrá **E** líneas con el siguiente formato:

$$\begin{aligned} &contrato_1 (a_{min_1}, a_{max_1})(t_{min_1}, t_{max_1})(d_{min_1}, d_{max_1}) finde_{max_1} finde_{b_1} \\ &contrato_2 (a_{min_2}, a_{max_2})(t_{min_2}, t_{max_2})(d_{min_2}, d_{max_2}) finde_{max_2} finde_{b_2} \\ &\dots \\ &contrato_e (a_{min_e}, a_{max_e})(t_{min_e}, t_{max_e})(d_{min_e}, d_{max_e}) finde_{max_e} finde_{b_e} \end{aligned}$$

nombre del contrato, (mínimo, máximo) número de asignaciones totales en toda la programación, (mínimo, máximo) días trabajados consecutivamente, (mínimo, máximo) días

descansados consecutivamente, máximo número de fines de semana, 0 – 1 booleana que nos muestra si la restricción de fin de semana completo esta activa (1 cuando la booleana esta activa).

En la siguiente línea se tiene el encabezado NURSE = 30 donde hay que seleccionar de la cadena el número **N** de enfermeras diferentes que estarán presentes en esta instancia (en este caso 30) y luego habrá **N** líneas con el siguiente formato:

$$\begin{aligned}
 &enfermera_1 Ncontrato_1 sk_1 habilidad_{1_1} habilidad_{1_2} \dots habilidad_{1_{sk}} \\
 &enfermera_2 Ncontrato_2 sk_2 habilidad_{2_1} habilidad_{2_2} \dots habilidad_{2_{sk}} \\
 &\dots \\
 &enfermera_n Ncontrato_n sk_n habilidad_{n_1} habilidad_{n_2} \dots habilidad_{n_{sk}}
 \end{aligned}$$

La figura 3.1 es un ejemplo del formato de los archivos de escenario que se presentaran durante todo el benchmark de este problema.

```

SCENARIO = n030w4

WEEKS = 4

SKILLS = 4
HeadNurse
Nurse
Caretaker
Trainee

SHIFT_TYPES = 4
Early (2,5)
Day (2,28)
Late (2,5)
Night (3,5)

FORBIDDEN_SHIFT_TYPES_SUCCESIONS
Early 0
Day 1 Early
    
```

Late 2 Early Day
Night 3 Early Day Late

CONTRACTS = 3

FullTime (15,22) (3,5) (2,3) 2 1

PartTime (7,15) (3,5) (2,4) 2 1

HalfTime (5,11) (2,4) (2,5) 1 1

NURSES = 30

HN_0 FullTime 3 HeadNurse Nurse Caretaker

HN_1 HalfTime 3 HeadNurse Nurse Caretaker

HN_2 HalfTime 2 HeadNurse Nurse

HN_3 PartTime 2 HeadNurse Nurse

NU_4 HalfTime 2 Nurse Caretaker

NU_5 HalfTime 2 Nurse Caretaker

NU_6 FullTime 2 Nurse Caretaker

NU_7 PartTime 2 Nurse Caretaker

NU_8 FullTime 2 Nurse Caretaker

NU_9 HalfTime 2 Nurse Caretaker

NU_10 PartTime 2 Nurse Caretaker

NU_11 FullTime 2 Nurse Caretaker

NU_12 PartTime 2 Nurse Caretaker

NU_13 HalfTime 2 Nurse Caretaker

NU_14 FullTime 2 Nurse Caretaker

NU_15 FullTime 2 Nurse Caretaker

NU_16 HalfTime 2 Nurse Caretaker

CT_17 FullTime 1 Caretaker

CT_18 HalfTime 1 Caretaker

CT_19 FullTime 1 Caretaker

CT_20 PartTime 1 Caretaker

CT_21 PartTime 1 Caretaker

CT_22 FullTime 1 Caretaker

CT_23 FullTime 1 Caretaker

CT_24 FullTime 1 Caretaker

TR_25 PartTime 1 Trainee

TR_26 PartTime 1 Trainee

TR_27 HalfTime 1 Trainee

TR_28 FullTime 1 Trainee

TR_29 HalfTime 1 Trainee

Figura 3.1 Archivo Sc-n030w4.

3.4.2. Archivo datos semanales (Demanda)

En la primera línea está el nombre del archivo WEEK_DATA.

En la segunda línea está el nombre interno de la instancia, se basa en la cantidad de enfermeras y número de semanas que componen esta instancia (Ejemplo: WD-n030w4).

En la siguiente línea se tiene el encabezado REQUIREMENTS luego comenzarán las líneas de demanda donde en cada T turno se tiene el nombre del turno y una adición para cada una de las SK habilidades lo que es igual al producto de $T \cdot SK$ este resultado es la cantidad siguiente de líneas que estarán presentes en la demanda de esta instancia con el siguiente formato donde D es igual a 7 que representa los días que componen a las demandas semanales (Ejemplo: Early HeadNurse (1,1) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0)):

$$\begin{aligned}
 &DTnombre_1 DSKnombre_{1_1} (Dn_{1_{1_1}}, Dx_{1_{1_1}}) (Dn_{1_{1_2}}, Dx_{1_{1_2}}) \dots (Dn_{1_{1_d}}, Dx_{1_{1_d}}) \\
 &DTnombre_1 DSKnombre_{1_2} (Dn_{1_{2_1}}, Dx_{1_{2_1}}) (Dn_{1_{2_2}}, Dx_{1_{2_2}}) \dots (Dn_{1_{2_d}}, Dx_{1_{2_d}}) \\
 &\dots \\
 &DTnombre_1 DSKnombre_{1_{sk}} (Dn_{1_{sk_1}}, Dx_{1_{sk_1}}) (Dn_{1_{sk_2}}, Dx_{1_{sk_2}}) \dots (Dn_{1_{sk_d}}, Dx_{1_{sk_d}}) \\
 &\dots \\
 &DTnombre_t DSKnombre_{t_{sk}} (Dn_{t_{sk_d}}, Dx_{t_{sk_d}}) (Dn_{t_{sk_d}}, Dx_{t_{sk_d}}) \dots (Dn_{t_{sk_d}}, Dx_{t_{sk_d}})
 \end{aligned}$$

Requerimientos contiene para cada turno en todas las habilidades requeridas el mínimo y óptimo de enfermeras para cada uno de los siete días que componen la demanda semanal (cuando el número identificador en el nombre del archivo cambia la demanda a resolver de la semana actual cambia).

En la siguiente línea se tiene el encabezado SHIFT_OFF_REQUESTS = 8 donde hay que seleccionar de la cadena el número P (en este caso el número 8) que son las requisiciones de turnos y luego habrá P líneas con el siguiente formato (Ejemplo: HN_1 Night Mon):

$$\begin{aligned}
 &P_enfermera_1 P_turno_1 P_dia_1 \\
 &P_enfermera_2 P_turno_2 P_dia_2 \\
 &\dots \\
 &P_enfermera_p P_turno_p P_dia_p
 \end{aligned}$$

La figura 3.2 es un ejemplo del formato de los archivos de datos semanales (demanda semanal) que se presentaran durante todo el benchmark de este problema.

```
WEEK_DATA
n030w4

REQUIREMENTS
Early HeadNurse (1,1) (0,0) (0,0) (0,0) (0,0) (0,0) (0,0)
Early Nurse (1,2) (1,1) (1,1) (1,2) (1,2) (1,1) (1,1)
Early Caretaker (2,4) (2,2) (1,3) (2,3) (2,3) (0,0) (1,1)
Early Trainee (1,1) (1,1) (1,1) (0,0) (0,1) (1,1) (0,0)
Day HeadNurse (0,0) (0,0) (0,0) (0,0) (0,0) (0,0) (1,1)
Day Nurse (1,1) (1,1) (1,1) (1,1) (1,1) (1,1) (1,1)
Day Caretaker (3,4) (1,3) (2,3) (3,4) (2,2) (1,1) (0,2)
Day Trainee (0,0) (1,1) (1,1) (0,0) (0,1) (1,1) (1,1)
Late HeadNurse (0,0) (0,0) (0,0) (1,1) (1,1) (1,1) (0,0)
Late Nurse (1,1) (1,1) (1,1) (1,1) (1,1) (1,2) (1,1)
Late Caretaker (2,3) (2,3) (2,2) (2,3) (1,3) (1,1) (0,1)
Late Trainee (1,1) (0,1) (0,0) (0,0) (0,0) (1,1) (1,1)
Night HeadNurse (0,0) (0,0) (0,0) (0,0) (0,0) (1,1) (0,0)
Night Nurse (1,1) (1,1) (1,1) (1,2) (1,2) (1,1) (1,2)
Night Caretaker (2,3) (2,3) (2,2) (2,3) (3,3) (0,2) (0,2)
Night Trainee (0,0) (1,1) (0,0) (1,1) (0,0) (0,0) (1,1)

SHIFT_OFF_REQUESTS = 8
HN_1 Night Mon
NU_4 Any Mon
CT_23 Day Tue
NU_16 Late Wed
NU_12 Any Fri
TR_25 Late Fri
NU_9 Any Sat
TR_28 Any Sat
```

Figura 3.2 Archivo WD-n030w4-0.

3.4.3. Archivo historial

En la primera línea está el nombre del archivo HISTORY.

En la segunda línea está el número identificador de que historial se está usando y el nombre interno de la instancia, se basa en la cantidad de enfermeras y número de semanas que componen esta instancia (Ejemplo: 0 n030w4).

En la siguiente línea se encuentra el encabezado NURSE_HISTORY seguido de las **N** líneas del archivo escenario las cuales describen el historial con el que inician las enfermeras en esta instancia con el formato siguiente (Ejemplo: HN_0 0 0 Night 4 4 0):

historial₁ a_global₁ finde_globales₁ ultimo_t₁ a_ultimo₁ a_continuas₁ d_continuos₁
historial₂ a_global₂ finde_globales₂ ultimo_t₂ a_ultimo₂ a_continuas₂ d_continuos₂
 ...
historial_n a_global_n finde_globales_n ultimo_t_n a_ultimo_n a_continuas_n d_continuos_n

El historial de las enfermeras debe registrar lo sucedido la semana anterior también se debe tomar en cuenta para procesar la primera demanda, este contiene nombre de la enfermera, asignaciones durante toda la programación (cero al procesar la primer semana), fines de semana trabajados durante toda la programación (cero al procesar la primer semana), nombre del último turno trabajado el domingo (None si no trabajo el domingo), número de asignaciones continuas del mismo turno si trabajado el domingo, número de días trabajados consecutivamente, días consecutivos de trabajo y días descansados continuamente.

La figura 3.3 es un ejemplo del formato de los archivos de historial que se presentaran durante todo el benchmark de este problema.

```
HISTORY
0 n030w4

NURSE_HISTORY
HN_0 0 0 Night 4 4 0
HN_1 0 0 None 0 0 5
HN_2 0 0 None 0 0 2
HN_3 0 0 Day 2 2 0
NU_4 0 0 None 0 0 2
NU_5 0 0 None 0 0 2
NU_6 0 0 Late 5 5 0
NU_7 0 0 Early 2 2 0
NU_8 0 0 Night 2 4 0
NU_9 0 0 None 0 0 4
```

```
NU_10 0 0 Late 1 5 0
NU_11 0 0 Early 3 3 0
NU_12 0 0 None 0 0 3
NU_13 0 0 None 0 0 5
NU_14 0 0 Early 2 2 0
NU_15 0 0 None 0 0 2
NU_16 0 0 None 0 0 2
CT_17 0 0 None 0 0 2
CT_18 0 0 None 0 0 2
CT_19 0 0 Night 2 2 0
CT_20 0 0 None 0 0 3
CT_21 0 0 None 0 0 2
CT_22 0 0 Late 2 2 0
CT_23 0 0 None 0 0 2
CT_24 0 0 None 0 0 2
TR_25 0 0 None 0 0 4
TR_26 0 0 Early 2 2 0
TR_27 0 0 None 0 0 2
TR_28 0 0 Late 1 5 0
TR_29 0 0 Night 1 3 0
```

Figura 3.3 Archivo H0-n030w1-0.

3.4.4. Archivo solución

El nombre del archivo contiene número de enfermeras, el total de semanas que abarcarán las soluciones, el identificador del historial que se usa en esta instancia y el número de la demanda que se solucionó en el archivo actual, la solución de cada semana se guarda en un archivo distinto (Ejemplo: Sol-n030w4-1-0).

En la primera línea está el nombre del archivo SOLUTION.

En la segunda línea está el número identificador de que semana es la solución y el nombre interno de la instancia, se basa en la cantidad de enfermeras y número de semanas que componen esta instancia (Ejemplo: 0 n030w4).

En la siguiente línea se tiene el encabezado ASSIGNMENTS = 104 donde hay que seleccionar de la cadena el número **A** (en este caso el número 104) que son las asignaciones totales de enfermeras que se ocuparon para solucionar esta instancia y luego habrá **A** líneas con el siguiente formato (Ejemplo: TR_28 Mon Late Trainee):

A_enfermera₁ D_enfermera₁ T_enfermera₁ SK_enfermera₁
A_enfermera₂ D_enfermera₂ T_enfermera₂ SK_enfermera₂
 ...
A_enfermera_a D_enfermera_a T_enfermera_a SK_enfermera_a

La figura 3.4 es un ejemplo del formato de los archivos de solución que se presentaran durante todo el benchmark de este problema.

Sol-n030w4-1-0

SOLUTION
0 n030w4

ASSIGNMENTS = 104
TR_29 Tue Early Trainee
TR_29 Wed Early Trainee
TR_29 Thu Day Trainee
...
HN_2 Sun Late Nurse
HN_3 Thu Late HeadNurse
HN_1 Mon Night HeadNurse

Sol-n030w4-1-1

SOLUTION
0 n030w4

ASSIGNMENTS = 104
TR_29 Tue Early Trainee
TR_29 Wed Early Trainee
TR_29 Thu Day Trainee
...
HN_2 Sun Late Nurse
HN_3 Thu Late HeadNurse
HN_1 Mon Night HeadNurse

Sol-n030w4-1-2

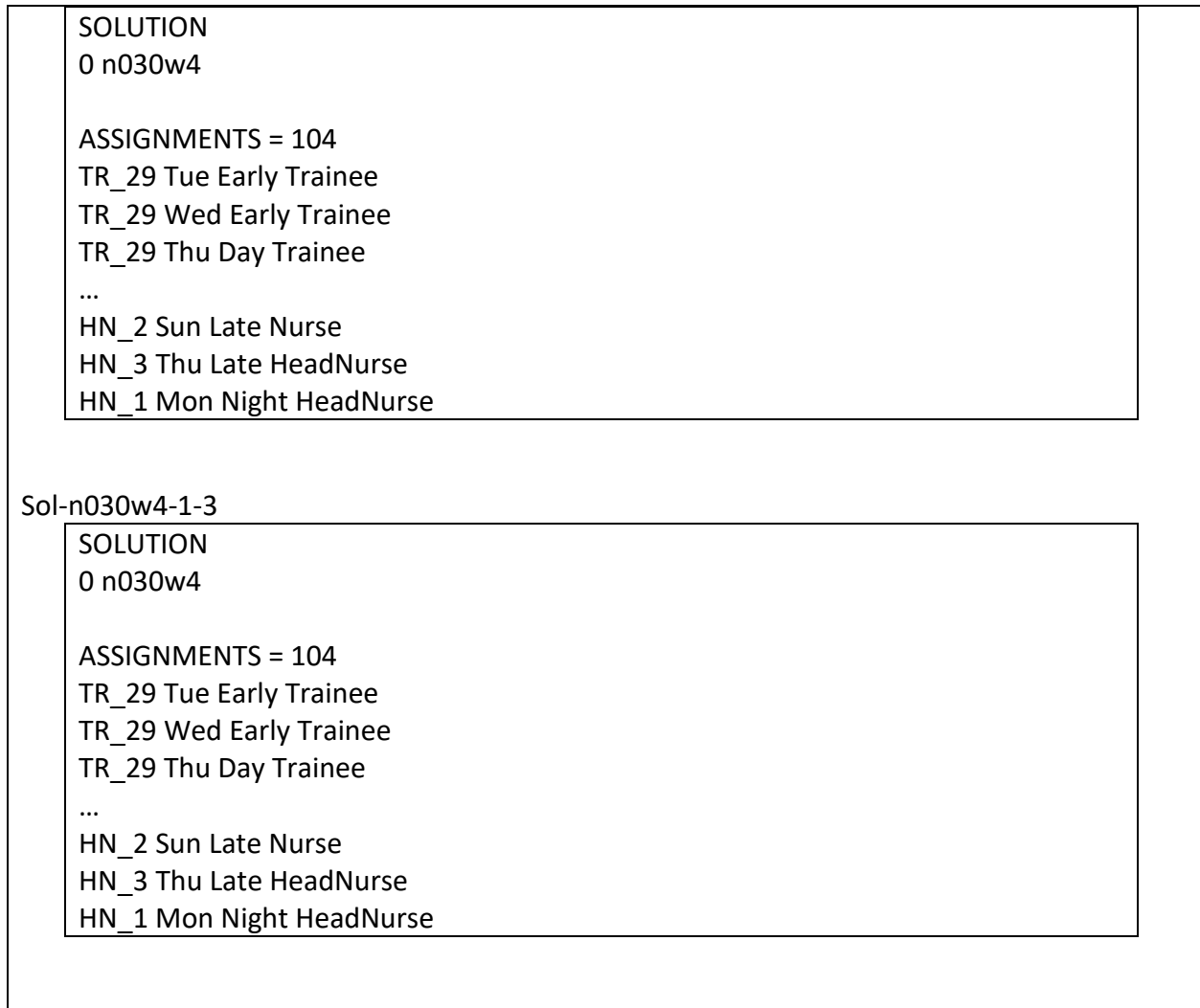


Figura 3.4 Archivos Sol-n030w4-1-0, Sol-n030w4-1-1, Sol-n030w4-1-2, Sol-n030w4-1-3.

Los parámetros generales para las distintas instancias son los siguientes:

- Habilidades:
 - Jefa de enfermeras
 - Enfermera
 - Cuidados
 - Practicante
- Preferencias personales
 - Turnos libres
 - Días libres

- Turnos:
 - Matutino
 - Diurno
 - Vespertino
 - Nocturno

- Contratos:
 - Tiempo-completo
 - Tiempo-parcial
 - Medio-tiempo
 - Veinte-por-ciento

- Sucesiones sin prohibiciones:
 - Matutino

- Sucesiones con posibilidad de tener prohibiciones:
 - Diurno - prohibiciones: Matutino.

- Sucesiones que siempre tienen prohibiciones:
 - Vespertino - prohibiciones: Matutino y Diurno.
 - Nocturno - prohibiciones: Matutino, Diurno y Vespertino.

3.4.5. Evaluador de factibilidad

Este fue creado para prevenir que los participantes gasten tiempo en la búsqueda de soluciones factibles cuando estas no existen y calificar la calidad de la solución final en cuanto a factibilidad y violación de restricciones suaves.

La siguiente línea de comando es un ejemplo del formato necesario para activar el evaluador:

```
Validador>java -jar validator.jar --sce Sc-n030w4.txt --his H0-n030w4-1.txt --weeks WD-n030w4-6.txt WD-n030w4-2.txt WD-n030w4-9.txt WD-n030w4-1.txt --sols Sol-n030w4-1-0.txt Sol-n030w4-1-1.txt Sol-n030w4-1-2.txt Sol-n030w4-1-3.txt
```

En la figura 3.5 se presenta el ejemplo de la evaluación de una solución, en este caso de la programación de una instancia de cuatro semanas para treinta enfermeras donde se puede observar los resultados de la factibilidad de esta solución debajo de la línea *Hard constraint violations* con un listado de las cuatro restricciones duras (*Minimal coverage constraints*, *Required skill constraints*, *Illegal shift type succession constraints*, *Single assignment per day*) que se deben respetar en todo el benchmark y seguido a esta sección se encuentra lo que se debe optimizar, debajo de la línea *Cost per constraint type* donde se observa el peso final de cada una de las siete restricciones suaves (*Total assignment constraints*, *Consecutive constraints*, *Non working days constraints*, *Preferences*, *Max working weekend*, *Complete weekends*, *Optimal coverage constraints*) y el peso total de la sumatoria de las siete violaciones (*Total cost*).

	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S	M	T	W	T	F	S	S		
HN_0	-	-	-	-	-	-	-	-	-	E	D	-	-	-	-	-	N	-	-	-	-	-	D	-	-	D	-	L		
HN_1	N	-	-	-	-	-	-	-	-	N	-	-	-	N	-	-	-	-	-	-	-	E	-	-	-	N	-	N	-	
HN_2	E	E	E	E	D	-	L	L	L	-	-	-	L	L	-	-	L	-	L	-	-	-	E	-	-	-	-	D	-	
HN_3	-	-	-	L	-	-	-	L	-	-	-	-	L	-	-	D	D	D	-	-	-	-	L	-	-	-	-	-	-	
NU_4	E	E	E	E	E	E	E	-	-	-	-	-	-	-	E	E	E	E	-	-	-	-	-	-	-	-	E	E	E	
NU_5	D	D	D	D	D	D	D	-	-	-	-	-	-	N	N	N	N	N	-	-	-	-	L	-	-	L	-	L	-	
NU_6	L	L	L	L	L	-	-	-	-	-	-	-	L	-	D	D	D	D	D	-	-	-	-	-	-	D	D	D	D	
NU_7	-	D	-	D	D	D	L	-	-	-	-	-	D	L	L	L	L	L	L	-	-	-	L	-	L	L	L	-	-	
NU_8	N	N	N	N	N	-	N	-	-	-	E	-	D	D	D	D	D	L	-	E	-	-	-	-	E	E	E	E	E	
NU_9	-	-	-	-	-	L	-	E	E	E	E	E	-	-	-	-	-	-	-	-	L	-	E	E	E	-	-	-	-	
NU_10	N	N	N	N	N	N	-	N	N	-	-	-	N	N	N	N	N	N	N	N	-	-	N	-	-	-	N	N	N	N
NU_11	-	N	-	N	N	-	E	E	E	D	D	D	-	-	-	-	-	-	-	-	D	L	L	L	L	L	L	-	-	
NU_12	-	-	-	-	D	-	-	D	D	L	L	L	-	-	-	-	-	-	N	N	N	N	N	N	N	N	-	-	-	
NU_13	-	-	-	-	-	-	-	L	L	-	-	E	E	E	-	-	-	-	E	D	D	D	D	D	-	-	-	-	-	
NU_14	-	-	-	-	-	-	-	N	N	N	N	N	-	-	-	L	-	-	-	E	E	-	L	L	-	L	L	L	L	
NU_15	-	-	-	-	-	-	-	D	D	D	D	L	-	-	N	-	-	-	N	N	N	N	N	N	-	N	N	N	-	
NU_16	-	-	-	-	-	-	-	N	N	N	N	N	-	-	N	N	N	-	N	-	-	-	D	D	D	-	-	-	-	
CT_17	L	L	-	-	-	-	-	N	N	N	N	N	-	-	D	D	D	L	L	-	-	N	N	N	N	N	-	-	-	
CT_18	L	L	L	-	L	-	L	L	L	-	L	L	L	-	L	-	L	-	-	-	-	L	L	L	L	-	-	-	-	
CT_19	L	L	L	-	L	L	-	L	-	-	-	-	E	E	L	L	L	-	L	L	-	D	D	D	L	L	-	-	-	
CT_20	D	D	L	L	L	L	L	L	-	L	L	-	-	-	E	E	E	E	E	-	-	D	D	D	L	-	-	-	-	
CT_21	E	E	E	E	E	D	D	D	L	L	L	L	-	-	D	D	L	L	L	-	-	E	D	D	D	-	-	-	-	
CT_22	E	E	E	E	E	-	-	D	D	D	D	D	-	-	D	D	D	D	D	-	-	E	E	E	E	E	-	-	-	
CT_23	D	D	D	D	D	-	-	E	D	D	D	D	-	-	E	E	D	D	D	-	-	E	E	E	D	D	-	-	-	
CT_24	E	E	E	E	E	-	-	E	E	E	E	E	-	-	E	E	E	E	E	-	-	E	E	E	E	E	-	-	-	
TR_25	-	-	-	-	-	-	D	D	-	-	-	L	L	L	L	-	-	-	-	-	D	D	-	D	D	D	-	-	-	
TR_26	-	-	-	-	-	E	E	-	-	-	L	-	-	-	-	-	-	E	E	E	-	-	N	-	-	-	-	-	-	
TR_27	-	N	N	N	-	-	-	N	-	-	N	N	N	-	D	-	-	-	-	-	-	-	E	-	-	E	-	E	-	
TR_28	-	L	L	L	-	-	-	E	E	D	D	D	-	-	E	-	L	-	L	-	L	L	L	-	-	-	-	-	L	
TR_29	-	E	E	D	-	L	-	-	-	-	-	E	D	D	-	N	N	N	N	-	-	E	-	-	E	L	-	-	-	

Hard constraint violations

Minimal coverage constraints: 0
 Required skill constraints: 0
 Illegal shift type succession constraints: 0
 Single assignment per day: 0

Cost per constraint type

Total assignment constraints: 920
Consecutive constraints: 5430
Non working days constraints: 3390
Preferences: 140
Max working weekend: 540
Complete weekends: 900
Optimal coverage constraints: 2370

Total cost: 13690

Figura 3.5 Evaluación de una solución factible.

3.4.6. Instancias del problema

En este caso el problema a resolver es el presentado en el INRC-II la composición del Benchmark de la tabla 3.17.

3.17 Tabla de instancias del benchmarck del INRC-II.

Nombre de la Instancia	Enfermeras	Semanas	Habilidades	Turnos	Contratos
n030w4_1_6-2-9-1	30	4	4	4	3
n030w4_1_6-7-5-3	30	4	4	4	3
n030w8_1_2-7-0-9-3-6-0-6	30	8	4	4	3
n030w8_1_6-7-5-3-5-6-2-9	30	8	4	4	3
n040w4_0_2-0-6-1	40	4	4	4	3
n040w4_2_6-1-0-6	40	4	4	4	3
n040w8_0_0-6-8-9-2-6-6-4	40	8	4	4	3
n040w8_2_5-0-4-8-7-1-7-2	40	8	4	4	3
n050w4_0_0-4-8-7	50	4	4	4	3
n050w4_0_7-2-7-2	50	4	4	4	3
n050w8_1_1-7-8-5-7-4-1-8	50	8	4	4	3
n050w8_1_9-7-5-3-8-8-3-1	50	8	4	4	3
n060w4_1_6-1-1-5	60	4	4	4	4

n060w4_1_9-6-3-8	60	4	4	4	4
n060w8_0_6-2-9-9-0-8-1-3	60	8	4	4	3
n060w8_2_1-0-3-4-0-3-9-1	60	8	4	4	3
n080w4_2_4-3-3-3	80	4	4	4	4
n080w4_2_6-0-4-8	80	4	4	4	4
n080w8_1_4-4-9-9-3-6-0-5	80	8	4	4	4
n080w8_2_0-4-0-9-1-9-6-2	80	8	4	4	4
n100w4_0_1-1-0-8	100	4	4	4	4
n100w4_2_0-6-4-6	100	4	4	4	4
n100w8_0_0-1-7-8-9-1-5-4	100	8	4	4	4
n100w8_1_2-4-7-9-3-9-2-8	100	8	4	4	4
n120w4_1_4-6-2-6	120	4	4	4	3
n120w4_1_5-6-9-8	120	4	4	4	3
n120w8_0_0-9-9-4-5-1-0-3	120	8	4	4	3
n120w8_1_7-2-6-4-5-2-0-2	120	8	4	4	3

Cada instancia tiene un historial, escenario y demanda con lo que se debe generar un archivo solución para continuar en cada semana que componga la demanda de la instancia actual y al final en el validador se evaluarán todas las soluciones generadas para todo el benchmark.

3.5. Modelo matemático

Los siguientes parámetros se encuentran en el archivo escenario y modelan los valores para la planeación de toda la programación:

N Conjunto de enfermeras

T Conjunto de turnos

SK Conjunto de habilidades

E Conjunto de contratos

$|W|$ Número de semanas

$|D|$ Número de días

$a_n^{[+/-]}$ Máximas y mínimas asignaciones para una enfermera n durante la planeación del horizonte

$b_n^{[+/-]}$ Máximas y mínimas asignaciones de trabajo consecutivos para una enfermera n

$f_n^{[+/-]}$ Máximos y mínimos descansos consecutivos para una enfermera n

g_n^+ Máximos número de fines de semana para una enfermera n durante la planeación del horizonte

h_n Boleana, 1 si ambos días del fin de semana deben ser trabajados, 0 si no

$\kappa_{n,sk}$ Boleana, 1 si la enfermera n tiene la habilidad sk

$o_t^{[+/-]}$ Máximas y mínimas asignaciones consecutivas del turno t

$u_{sk,t}$ Boleana, 1 si el turno t puede ser asignado de acuerdo con la asignación de t'

Los siguientes parámetros son definidos para cada semana:

w Número de la semana actual

$\widehat{\Delta}_{t,sk}^d$ Cobertura de demanda mínima

$\overline{\Delta}_{t,sk}^d$ Cobertura de demanda optima

$p_{n,t}^d$ Boleana, 1 si la enfermera pidió no trabajar en el turno t en el día d ($t=0$ si pidió el día libre).

Los siguientes parámetros son para especificar los valores de la semana previa los cuales son dados en el archivo historial y deben trasladarse en cada semana:

q_n^{id} id es para el último turno en el que la enfermera n trabajo previamente (0 si fue un día libre)

$q_{n,t}$ Los turnos consecutivos del tipo t trabajados por la enfermera n si caen a final de semana (0 si $t \neq q_n^{id}$)

q_n^w Los días trabajados consecutivamente por la enfermera n si caen a final de semana (0 si $q_n^{id} = 0$)

q_n^r Los días descansados consecutivamente por la enfermera n si caen a final de semana (0 si $q_n^{id} \neq 0$)

x_n^z Total, de asignaciones para la enfermera n hasta el momento

y_n^z Total, de semanas trabajadas por la enfermera n hasta el momento

VARIABLES DE DECISIÓN:

$\alpha_{n,d,t,sk} \in \{0, 1\} \quad \forall n \in N, \forall d \in D, \forall t \in T, \forall sk \in SK$ Booleana, 1 si la enfermera n fue asignada en el turno t al día d con una de sus habilidades sk , 0 de otra manera

$w_n \in \{0, 1\} \quad \forall n \in N$ Indica si la enfermera n trabajó al menos un día de la semana, 0 si no

Las siguientes inecuaciones modelan las restricciones duras anteriormente descritas:

Asignación de un turno por día.

$$H1 \quad \forall n \in N, d \in D\{1 \dots 7\}$$

$$\sum_{\substack{s \in S \\ k \in K}} \alpha_{n,t,sk}^d \leq 1$$

Demanda mínima.

$$H2 \quad \forall t \in T, sk \in SK, d \in D\{1 \dots 7\}$$

$$\sum_{n \in N} \alpha_{n,t,sk}^d \geq c_{sk}^d$$

Asignaciones legales.

$$H3 \quad \forall n \in N, t, t' \in T, sk \in SK, d \in D\{1 \dots 7\}: u_{t,t'} = 0$$

$$\sum_{sk \in SK} \alpha_{n,t,sk}^d + \sum_{sk \in SK} \alpha_{n,t',sk}^{d+1} \leq 1$$

La enfermera debe tener la habilidad requerida.

$$H4 \quad \forall n \in N, t \in T, d \in D\{1 \dots 7\}, sk \in SK: \kappa_{n,sk} = 0$$

$$\alpha_{n,t,sk}^d = 0$$

La violación de las restricciones se mide usando variables no negativas o booleanas:

$C_{d,t,sk}^{S1} \geq 0$ Las enfermeras faltantes para cubrir óptimamente la demanda de día d , turno t y habilidad sk .

$C_{n,t,d}^{S2a} \geq 0$ Días faltantes en el bloque de turnos t que comienzan el día d , para la enfermera n .

$C_{n,t,d}^{S2b} \geq 0$ días excedidos para la enfermera n el turno t en el día d viola los turnos consecutivos máximos.

$C_{n,d}^{S2c} \geq 0$ Días faltantes en el bloque de trabajo que comienzan el día d , para la enfermera n .

$C_{n,d}^{S2d} \geq 0$ días excedidos que trabajo la enfermera n en el día d viola los días consecutivos máximos.

$C_{n,d}^{S3a} \geq 0$ Días faltantes en el bloque de días libres que comienzan el día d , para la enfermera n .

$C_{n,d}^{S3b} \geq 0$ Para los días d de la enfermera n violan los días consecutivos máximos de descanso.

$C_{n,d}^{S4} \in \{0, 1\}$ 1 si la asignación del día d viola las preferencias de la enfermera n .

$C_n^{S5} \geq 0$ Número de fines de semana en el que la enfermera n tiene alguna violación de fines de semana completos.

$C_n^{S6} \geq 0$ Número de turnos totales fuera de los límites permitidos en la enfermera n .

$C_n^{S7} \geq 0$ Número de fines de semana arriba del máximo permitido en la enfermera n .

La función objetivo es la suma de todas las violaciones de las restricciones suaves:

$$\text{Minimizar } f = 30 * \sum_{\substack{d \in D\{1...7\} \\ t \in T \\ sk \in SK}} C_{d,t,sk}^{S1}$$

- S1. Cubrir el número óptimo de enfermeras para cada día y turno.

$$+ 15 * \sum_{\substack{d \in D\{1...7\} \\ t \in T \\ sk \in SK}} (C_{n,t,d}^{S2a} + C_{n,t,d}^{S2b})$$

- S2. Respetar asignaciones consecutivos mínimos y máximos para turnos y valores globales.

$$+ 30 * \sum_{\substack{n \in N \\ d \in D\{1...7\}}} (C_{n,d}^{S2c} + C_{n,d}^{S2d})$$

$$+ 30 * \sum_{\substack{n \in N \\ d \in D\{1...7\}}} (C_{n,d}^{S3a} + C_{n,d}^{S3b})$$

- S3. Respetar días libres consecutivos mínimos y máximos.

$$+ 10 * \sum_{\substack{n \in N \\ d \in D\{1...7\}}} C_{n,d}^{S4}$$

- S4. No asignar una enfermera a un turno no deseado.

$$+ 30 * \sum_{n \in N} C_n^{S5}$$

- S5. Si se asigna al fin de semana se debe cubrir sábado y domingo

$$+ 20 * \sum_{n \in N} C_n^{S6}$$

- S6. Respetar asignaciones totales mínimas y máximas para los días trabajados.

$$+ 30 * \sum_{n \in N} C_n^{S7}$$

- S7. Respetar mínimos y máximo de fines de semana trabajados.

$$\text{Sujeto a } H \{H1, H2, H3, H4\} = 0$$

Capítulo 4.

4. Solución

4.1. Esquema general

Toda la información anteriormente expuesta de este problema puede clasificarse de forma general en distintos conjuntos específicamente en cuatro conjuntos distintos como se muestra en la figura 4.1, donde se debe satisfacer todos los turnos de cada día con la asignación de estos a las enfermeras disponibles y que tengan la habilidad solicitada donde lo complicado es encontrar un óptimo global para cada enfermera debido a que calcularlo consume un tiempo no razonable y además este óptimo global no debe violar las restricciones duras y se debe optimizar el peso de la violación de las restricciones suaves.



Figura 4.1 Problema de asignación de turnos de enfermeras dividido en conjuntos(NRP).

El esquema general del problema se muestra en la figura 4.2 donde deben introducirse los datos de todos los archivos de la instancia seleccionada para posteriormente programar la solución de la demanda solicitada durante cada semana y ya que la factibilidad de la solución depende de lo

sucedido en el archivo historial de la semana posterior a la demanda principal se deberá contar con la memoria de este historial para semanas posteriores, es así al final de cada demanda se debe generar un archivo solución txt y una estructura historial dentro del programa con los datos que se ocuparán para resolver la demanda de la semana siguiente repitiendo este paso hasta haber resuelto todas las demandas de la instancia, en la sección 3.4.3 Archivo historial se describe con detalle la información necesaria para conservar la factibilidad entre semanas cuando estas se concatenan, al final del esquema general tendremos una cantidad de archivos de solución equivalente a la duración en semanas de la instancia, este conjunto de archivos finales es el que se introducirá en el evaluador para conocer la factibilidad y calidad de la solución de acuerdo con el parámetro Total cost.

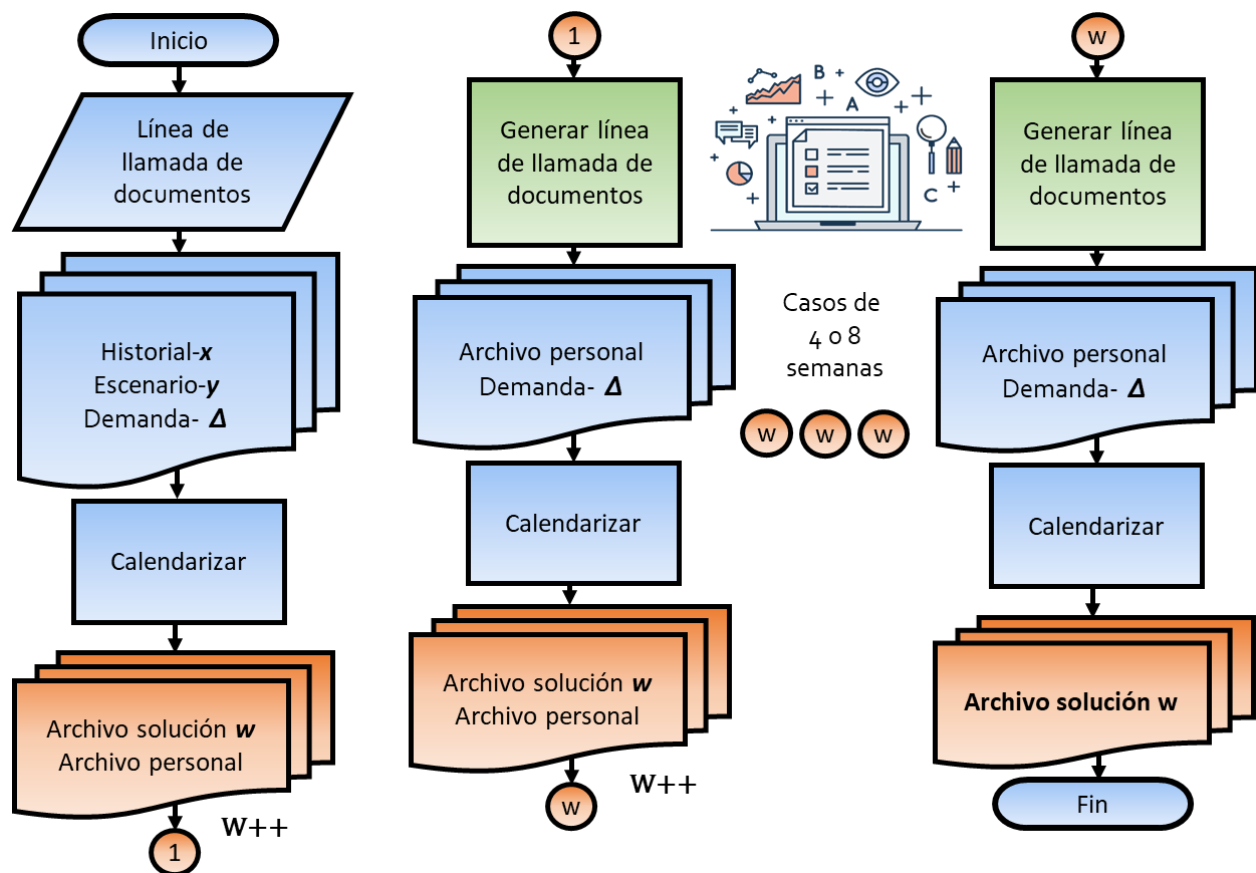


Figura 4.2 Esquema de solución general para el problema NRP.

4.2. Métodos desarrollados

En este capítulo se presentarán los generadores de soluciones factibles determinista y otro no determinista que fueron desarrollado con la misma finalidad de encontrar las soluciones factibles de todo el benchmark en un tiempo considerablemente menor al tiempo límite que se muestra en el INRC-II, además de incluir algunos vecindarios aplicados en una búsqueda voraz para mejorar las soluciones iniciales del método que se consideró el mejor esquema de generación de soluciones iniciales de esta experimentación.

4.2.1. Algoritmo determinista

Primero se decidió que la información debería tratarse en dos fases una de factibilidad y otra de optimización, la segmentación se muestra en la figura 4.3 donde las asignaciones solo contemplarían las restricciones duras del problema.

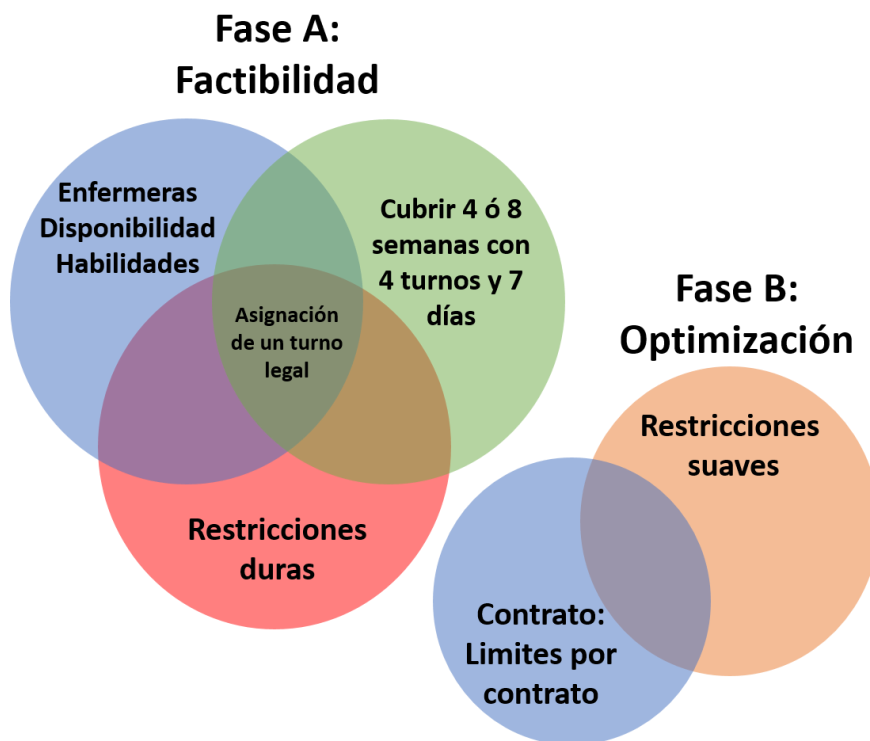


Figura 4.3 Segmentación y separación del problema NR en fases A y B.

El algoritmo determinista fue el primero en desarrollarse con la finalidad de encontrar rápidamente soluciones iniciales factibles aunque no con mucha calidad en cuanto a la violación de restricciones suaves, en la figura 4.4 se observa cuales son conjuntos que se consideran importantes para desarrollar un algoritmo ad-hoc determinista que funcione en tiempos razonables y siempre encuentre una solución factible cuando esta existe.



Figura 4.4 Solución de una instancia de NRP con restricciones duras.

Existe un conjunto de demanda mínima $\widehat{\Delta} = \{\widehat{\delta}_{d_1 t_1 s k_1}, \dots, \widehat{\delta}_{d_7 t_4 s k_4}\}$

Existe un conjunto de demanda optima $\bar{\Delta} = \{\bar{\delta}_{d_1 t_1 s k_1}, \dots, \bar{\delta}_{d_7 t_4 s k_4}\}$

Se crea el conjunto $A = \{\alpha_{n_1 d_1 t_1 s k_1}, \dots, \alpha_{n_{120} d_7 t_4 s k_4}\}$ donde $\alpha_{n_i d_j t_k s k_l}$ valdrá uno si la enfermera $n_i \in N$ con una de sus habilidades $s k_l \in SK$ satisface la demanda $\widehat{\delta}_{d_j t_k s k_l}$ del día $d_j \in D$ y turno $t_k \in T$, cero en caso contrario.

$$\sum_{n=1}^{|N|} \alpha_{n,d,t,s} \geq \widehat{\delta}_{d,t,s} \quad \forall d \in D; \forall t \in T; \forall s \in S$$

En el algoritmo **Hard – Restricted Assignment Greedy Solver** se realizarán las asignaciones si estas no violan las restricciones **H1**, **H2**, **H3** y si al final de las asignaciones $|A| - |\hat{\Delta}| \geq 0$ se considera que la restricción **H4** esta satisfecha.

Se tomaron varios criterios de acuerdo con cada archivo esto se debe a que se proporciona una gran cantidad de información que se considera útil para tomar estrategias que nos ayuden con el problema. Principalmente se consideró implementar la idea de un generador de soluciones totalmente aleatorio pero surgieron algunos inconvenientes con la conservación de la factibilidad relacionados con el poder laboral del equipo de enfermeras para resolver cada instancia y la naturaleza del tipo de solución inicial esperada como factible, entonces se tomo la desición de aprovechar toda la información aplicando ideas no recomendadas en la literatura como ordenar la información bajo ciertos criterios y ademas implementar asignaciones en cadenas lo más continuas posibles bajo la restricción de conservación de la factibilidad. Estos criterios consisten en diferenciar la fuerza laboral más restringida de la más flexible de acuerdo con el historial actual haciendo ordenamientos con la toma de desiciones como, ¿Que enfermeras han trabajado más días consecutivos?, ¿Quienes han tenido menos días libres consecutivos? y ¿Que turnos tienen prohibidos cada enfermera los lunes?. El algoritmo de la figura 4.5 fue el primer logro de esta investigación al encontrar que bajo estos criterios y cuando se tiene al menos el número necesario de enfermeras para resolver la demanda mínima en este problema entonces se puede solucionar todo el benchmark del INRC-II con tiempos que van desde 1.2 segundos para las instancias de 30 enfermeras hasta 4.2 segundos para las instancias de 120 enfermeras. El verdadero inconveniente fue el peso de las restricciones suaves violadas, el cuál es muy alto en estas soluciones y el algoritmo cuenta con la lectura de los datos para cada instancia los cuales son N número de enfermeras, $Sk_{n,sk}$ número de habilidades por enfermera, T cantidad tipos turnos que se ocuparan para la matriz según el número de habilidades, Δ_{dtsk} tamaño de la demanda mínima, W cantidad de semanas a solucionar, leer la demanda mínima w al empezar una semana nueva, se guardan los datos del historial actual en el archivo personal tras cada nueva semana, se usan los criterios de ordenamiento según la información disponible, se evaluan que turnos no pueden ser asignados los lunes según la influencia del turno asignado el domingo

de la semana anterior, es entonces que para cada enfermera $n_1 \in N$, para cada día $d \in D$, para cada turno $t \in T$ y para cada habilidad de la enfermera $sk_i \in n_1$ se verifica si la demanda actual en Δ_{dtsk} es mayor que cero, si la enfermera n_1 tiene la habilida SK_{nsk} solicitada en Δ_{dtsk} y si las sucesiones de turnos de la enfermera n_1 le permite atender esa demanda en el turno solicitado. Si la enfermera cumple las características solicitadas y no hay restricción que le impida atender la demanda, asignamos el día d a la enfermera n_1 con la habilidad sk en el turno t , incrementamos los turnos C_n (para implementaciones en las que solo se está buscando una solución factible de cualquier calidad este contador C_n puede ser reestructurado como una variable booleana que registre directamente si la enfermera n trabajó en domingo), asignados a la enfermera n decrementamos la demanda en Δ_{dtsk} en uno y terminamos los ciclos T y $SK_{n,sk}$, para continuar con la demanda del día d siguiente, esto se hace hasta terminar el ciclo FOR de enfermeras n para al final crear el archivo solución de la semana w . Si la enfermera no pudiera atender la demanda o no tiene la habilidad requerida se considera otra combinación de casos hasta que todos hayan sido examinados.

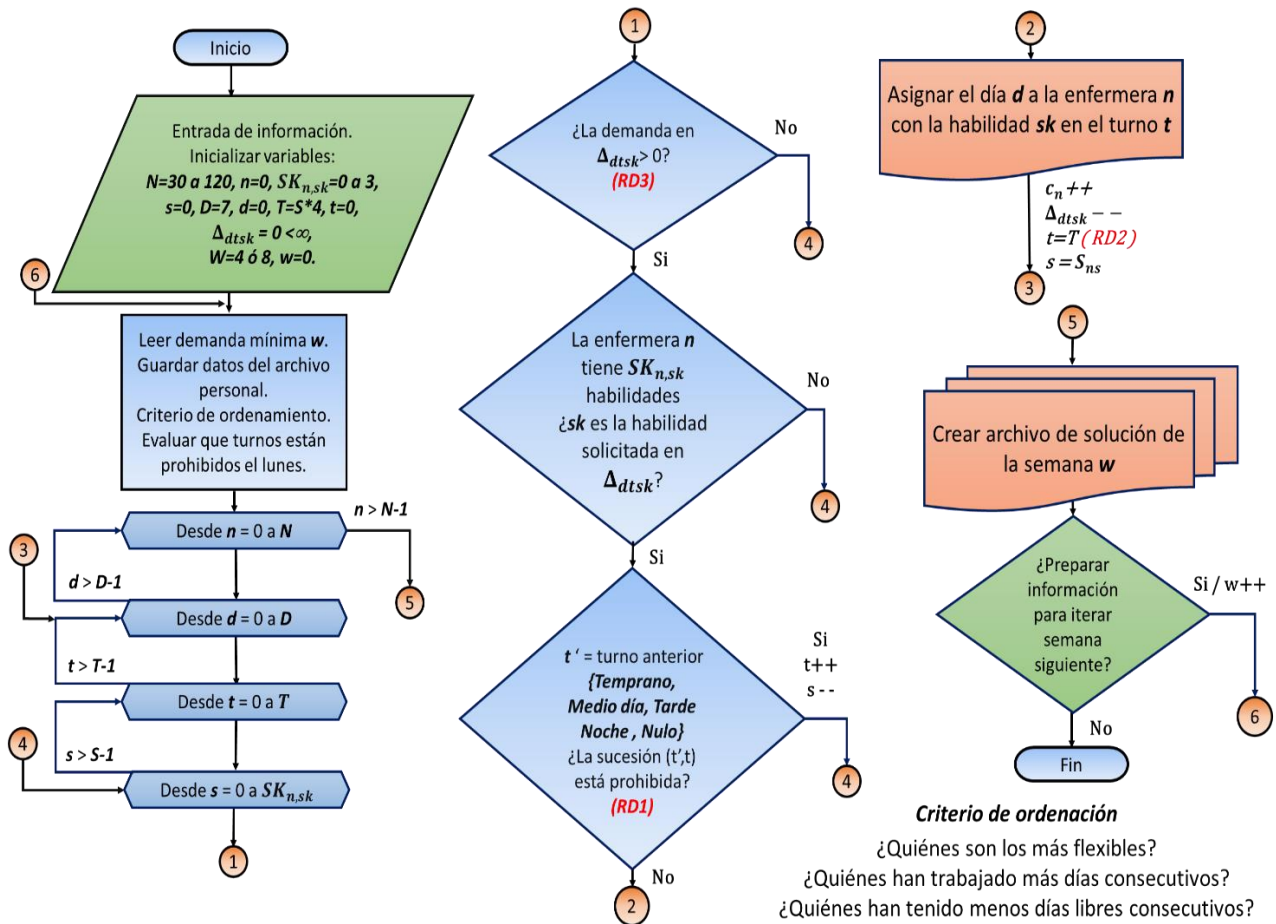


Figura 4.5 Solución de una instancia de NRP tomando en cuenta sólo las restricciones duras con el algoritmo Hard – Restricted Assignment Greedy Solver.

4.2.2. Método no determinista

El algoritmo no determinista fue el segundo en desarrollarse con la finalidad de encontrar rápidamente soluciones iniciales factibles pero incluyendo decisiones más complejas y adaptables para que las soluciones iniciales tuvieran un peso mejorado de al menos una reducción del peso de un 50% que el algoritmo determinista conservando el consumo de tiempo razonable y menor a los tiempos límite del INRC-II.

Lo aprendido del algoritmo determinista es la importancia de tomar en cuenta toda la información que nos proveen para solucionar el problema y no sólo pensar en esta como una limitación sino como un apoyo para crear criterios como el siguiente, si se tiene al menos el

número necesario de enfermeras para cubrir la demanda mínima solicitada en la instancia, entonces en una asignación aleatoria de los días de la demanda a las enfermeras se podrá cubrir toda la demanda mínima y además una parte de la demanda óptima, pero como anteriormente se había observado esto no garantiza la factibilidad de la solución final, entonces se mejoraron los criterios de ordenamiento de la información con criterios de límites por enfermeras, un ordenamiento de las enfermeras y una separación de las demandas en mínima y óptima para que la demanda óptima pueda ser considerada en lugar de ser ignorada, esto sería parte del nuevo método generador de soluciones factibles donde tomar en cuenta las asignaciones óptimas vuelve infactible todas las soluciones pero al agregar límites a las enfermeras para cubrir estas asignaciones y con criterios de aleatoriedad sobre que demanda se debe considera a cubrir vuelve una solución factible de mala calidad en una solución factible de mejor calidad si se respetan todas las limitaciones del problema actual, pero si en esté tratáramos de solucionarlo solo desde lo óptimo sin tomar en cuenta los mínimos factibles podríamos agotar todo el personal antes de encontrar una solución factible que cubra toda la demanda mínima volviendo a este método infactible en un caso hipotético como el siguiente, donde tenemos el suficiente personal para cubrir exactamente lo mínimo pero no contamos con suficiente personal para cubrir la demanda óptima. Entonces en la figura 4.6 se puede observar que para el segundo generador de soluciones factibles se considero agregar las restricciones suaves como duras con el criterio de que las enfermeras estaran disponibles en principio con los límites mínimos de sus contratos más parametros adaptables que crecieran en un rango conforme se busque una solución factible sin éxito y este parametro crecientemente adaptable se asignara aleatoriamente entre cero y su valor actual.



Figura 4.6 Solución de una instancia de NRP con restricciones duras y restricciones suaves aleatorias.

Desarrollo Fase A (Factibilidad H&RS).

En el método **Hard & Random Soft - Restricted Assignments Greedy Solver** se realizarán las asignaciones si estas no violan las restricciones **H1, H3, H4** y las suaves **S1, S2, S3, S4, S5, S6** y **S7**, que se consideran restricciones duras aleatoriamente activas en cada enfermera con distintos rangos aleatorios y crecientes tras cada iteración que no satisfaga **H2** en varias posibles combinaciones para no ser violadas.

Si al final de las asignaciones $|A| - |\hat{\Delta}| \geq 0$ se considera que la restricción **H2** esta satisfecha y si $|A| > |\hat{\Delta}|$ se cubrieron días óptimos.

El método de la figura 4.7 fue el segundo logro de esta investigación al encontrar que bajo los criterios del algoritmo determinista y otras varias modificaciones se logra un método que en unos pocos segundos extra encuentra mejores soluciones iniciales, el cual se describe a continuación.

Lectura de los datos del archivo escenario, inicializar las variables N número de enfermeras en la instancia, SK_{nsk} número de habilidades en cada enfermera n , D días que componen un ciclo de trabajo, T turnos que componen cada día, el parametro e de conteo de días para poder compararlo con el límite máximo de días trabajados del contrato más EXT , las matrices de demanda se representan con $\Delta_{dtsk[0]}$ sea la demanda mínima y $\Delta_{dtsk[1]}$ sea la demanda máxima, E_n es la cantidad de días máximos que una enfermera puede trabajar, W es el tamaño en semanas de la instancia, la variable RNG puede iniciar con cualquier valor, EXT es el parametro aleatorio entre cero y el valor actual con la función de agregar a los días mínimos del contrato de una enfermera días extra aleatorios que no asignarán más allá del límite natural de la duración de una semana de acuerdo con el ciclo FOR limitado en D , $RESET$ sea retomar la información respaldada para repetición al no encontrar una solución factible con los parametros actuales, $REST_n$ sean los días mínimos que una enfermera n puede descansar, desición booleana L sea el máximo número de intentos de encontrar una solución factible. Lectura de Historial y Demanda w , entonces tenemos la primer desición ¿ L excede 20000 intentos de buscar una solución de la demanda en la semana w ? para lo cual tenemos dos posibles respuestas.

- Respuesta Sí (se exceden 20000 intentos), se repite la búsqueda de toda la instancia.
- Respuesta No, continuamos con la siguiente desición la cual es ¿ $RESET$ es igual a cero? Para la cual tenemos dos posibles respuestas.
- Respuesta No, $RESET$ no es igual a cero entonces $RESET$ se iguala con cero se retoman todos los valores iniciales de la semana w anteriormente almacenados.
- Respuesta Si, $RESET$ es igual a cero el cual siempre comienza con este valor y asegura el almacenamiento de la información inicial, entonces se continua con las instrucciones de criterio de ordenamiento, evaluar que turnos estan prohibidos el lunes según los turnos asignado el

domingo del archivo historial w , D_{min} es una variable que guarda el número de asignaciones para satisfacer la demanda mínima en $\Delta_{dtsk[0]}$, luego se respalda toda la información de la semana w para restauración de la información en caso de no encontrar una solución a la demanda w con los parametros actuales. Es entonces que para cada enfermera $n_1 \in N$, para cada día $d \in D$, para cada turno $t \in T$ y para cada habilidad de la enfermera $sk_i \in n_1$, donde despues del primer ciclo FOR se tiene el parametro EXT que tendra un número aleatorio entre cero y el valor actual del parametro L dividido entre 1200, el parametro e se iguala a cero, r toma el valor de la cantidad de días mínimos en $REST_n$ que la enfermera n puede descansar y sirve para volver a asignar días de trabajo cuando se terminan los días de descanso, el parametro RNG es la demanda que se tomara en cuenta para asignación con una probabilidad de 0.75% de que sea escogida la demanda mínima, entonces comienza la pregunta de ¿La demanda $\Delta_{dtsk[RNG]}$ es mayor que cero? Donde RNG podra tener valores 0 ó 1 y dependiendo de la demanda seleccionada aleatoriamente tendremos dos posibles respuestas.

- Respuesta No, cambiamos al siguiente tipo de demanda que se esta solicitando.
- Respuesta Sí, pasamos a la siguiente pregunta ¿El valor actual de e los días trabajados continuamente son menores que los parametros $E_n + EXT$ ó $5 + \frac{L}{10000}$? Tenemos dos posibles respuestas.

(Donde el parametro $5 + \frac{L}{10000}$ representa el límite natural de días continuos de trabajo para que una persona no colapse pero entre más repeticiones son necesarias para encontrar una solución factible este límite crecerá en un máximo de 7 días con la finalidad de que al solucionar una semana regrese a su valor inicial) .

- Respuesta No, restamos un día a r días de descanso continuos y hacemos otra pregunta ¿ r es igual a cero? Con dos respuestas posibles las cuales son.
 - Respuesta Sí, las asignaciones e de trabajo continuo regresan a cero y los descansos r que la enfermera n puede tomar regresan a tener el valor en $REST_n$.
 - Respuesta No, se regresa al ciclo FOR de D .

- Respuesta Sí, se continua con la pregunta ¿la habilidad sk de la enfermera n es la solicitada en la demanda $\Delta_{dtsk[RNG]}$? Para lo cual tenemos dos posibles respuestas.
- Respuesta No, regresamos al ciclo FOR de $SK_{n,sk}$.
- Respuesta Sí, se continua con la pregunta ¿La sucesión t' y t esta prohibida? Puede ocurrir una de las dos siguientes posibles respuestas.
- Respuesta Sí, la sucesión esta prohibida incrementamos en uno el turno t , decrementamos en uno el contador del ciclo FOR s que volvera a su estado anterior por el proceso del ciclo FOR y continuará con la comparación de la misma habilidad sk pero en otro t turno.
- Respuesta No, la sucesión no esta prohibida y asignamos el día d a la enfermera n con la habilidad sk en el turno t y se compara el valor actual de RNG con cero para saber que demanda se esta satisfaciendo ¿ RNG es igual a cero? Para lo cual tenemos dos posibles respuestas.
- Respuesta Sí, se esta satisfaciendo la demanda mínima $\Delta_{dtsk[0]}$ entonces incrementamos los turnos e asignados a la enfermera n decrementamos la demanda $\Delta_{dtsk[0]}$ mínima en uno y tambien la $\Delta_{dtsk[1]}$ demanda óptima si es mayor que cero, decrementamos el contador $Demin$ de la demanda mínima total en uno y terminamos los ciclos T y $SK_{n,sk}$, para continuar con la demanda del día d siguiente.
- Respuesta No, se esta satisfaciendo la demanda óptima $\Delta_{dtsk[1]}$ entonces incrementamos los turnos e asignados a la enfermera n decrementamos la demanda óptima $\Delta_{dtsk[1]}$ en uno y tambien la $\Delta_{dtsk[0]}$ demanda mínima si es mayor que cero, terminamos los ciclos T y $SK_{n,sk}$, para continuar con la demanda del día d siguiente. Esto se hace hasta terminar el ciclo FOR de enfermeras n donde se responde la última pregunta ¿ $Demin$ es igual a cero? Para lo cual se tienen dos respuestas
- Respuesta No, se iguala el valor de $RESET$ con 1 y cantidad de intentos L se incrementa en 1 para regresar a la pregunta de cantidad de intentos L donde se repetirá este procedimiento hasta haber encontrado la solución inicial de la instancia actual en caso contrario los parametros se actualizaran tras cada repetición fallida para permitir asignaciones extra a las permitidas por los límites mínimos de los contratos de las enfermeras.

- Respuesta Sí, se crea el archivo solución de la semana w , y el historial $w+1$ y se pregunta el valor de w es menor que W ? Para lo que tenemos dos posibles respuestas.

- Respuesta Sí, se incrementa el valor de w en uno y se regresa a leer las siguientes entradas de Historial y Demanda w .

- Respuesta No, termina la ejecución del generador de soluciones factibles no determinista (Hard & Random Soft - Restricted Assignments Greedy Solver).

Entonces cuando se tiene al menos el número necesario de enfermeras para resolver la demanda mínima en este problema con este método se puede solucionar todo el benchmark del INRC-II.

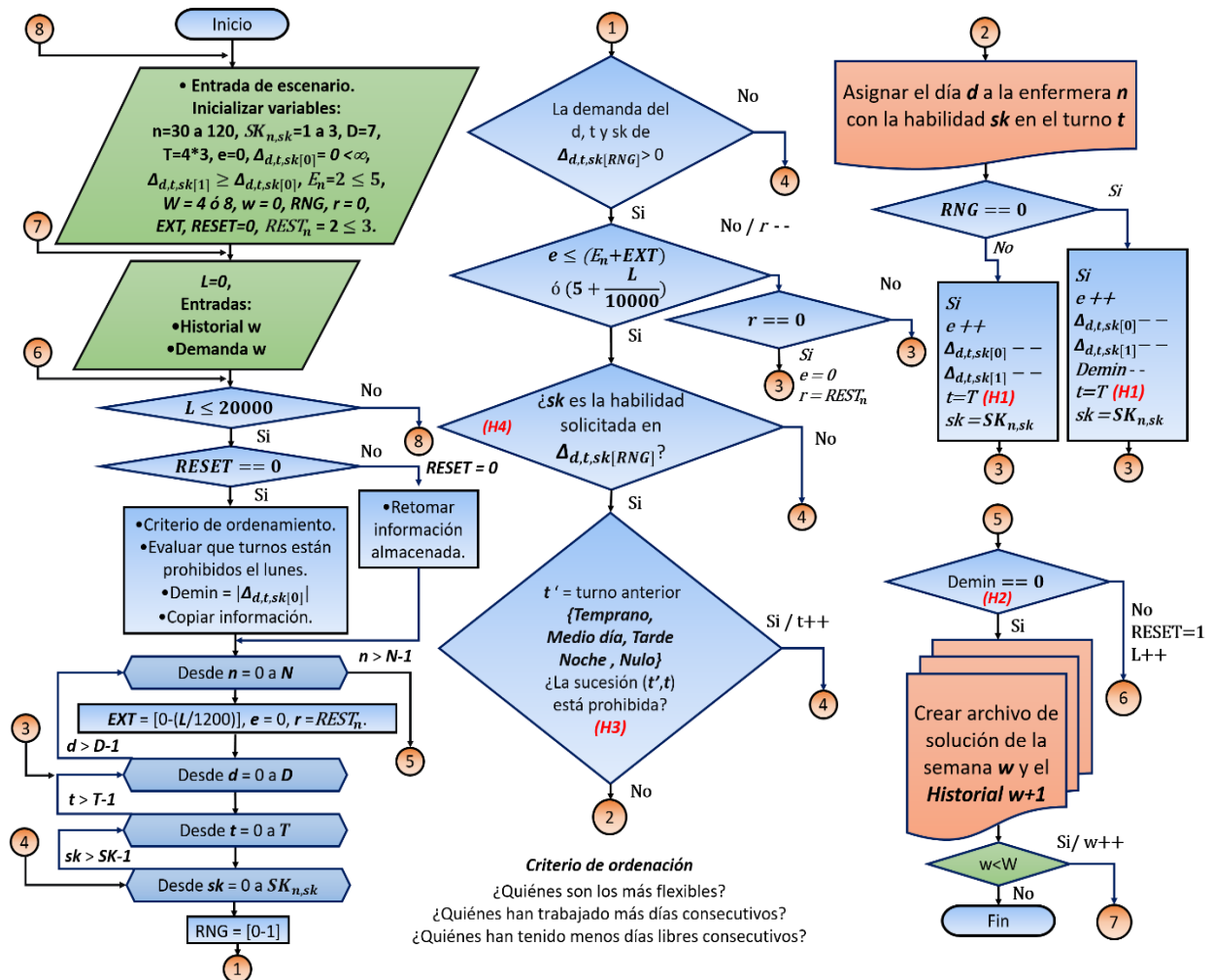


Figura 4.7 Hard & Random Soft - Restricted Assignments Greedy Solver, diagrama de flujo.

El desarrollo de una buena caracterización a profundidad de un problema es fundamental para encontrar el diseño y características que diferencian a un problema similar de otro, además de poder plantear correctamente un algoritmo exacto polinomial o desarrollar los atajos que componen una heurística o metaheurística, este método se desarrolló gracias a este pensamiento y al haber encontrado una buena caracterización que encontrara todas las respuestas al INRC-II de forma determinista, lo cual asegura que la misma caracterización podría funcionar de forma no determinista con el objetivo de optimizar los costos alcanzados de forma determinista.

Procedimiento de la heurística Hard & Random Soft - Restricted Assignments Greedy Solver

Análisis y planeación para el reclutamiento

Se mide la fuerza laboral disponible en cada enfermera para ordenarlas según la flexibilidad de sus habilidades, contratos y asignaciones hasta el momento.

Búsqueda local estricta

Comenzando por la enfermera con más flexibilidad, los parámetros de los límites principales se establecen en lo mínimo que se le puede exigir de acuerdo a los contratos de cada enfermera, de esta forma se le asignan días seleccionados aleatoriamente entre los dos tipos de demanda óptima y mínima de la semana actual, hasta encontrar una solución aceptable para esta semana.

Búsqueda local flexibilizada

Cuando las enfermeras no tienen éxito para cubrir toda la demandan mínima con los parámetros limitados se flexibilizan cada mil doscientas iteraciones asignando uno más a todos los límites mínimos de cada enfermera donde ahora se asignarán turnos mínimos y óptimos en un rango aleatorio entre el límite mínimo que marcan sus contratos y la n-ésima cantidad de veces dividida entre mil doscientas repeticiones en las que las enfermeras no logren satisfacer la demanda mínima de la semana actual.

Búsqueda local semanal

Se regresan los parámetros al estado original de la búsqueda local estricta y los de la búsqueda local flexibilizada regresan a cero, para realizar una búsqueda local estricta o flexibilizada según el caso en cada nueva semana que se analice.

Repetición global

Si las iteraciones totales de las repeticiones fallidas para una semana suman veinte mil intentos se repetirá la búsqueda desde la primera semana.

En la tabla 4.1 se puede observar los resultados alcanzados con los métodos que se desarrollaron para encontrar las soluciones iniciales del INRC-II, donde podemos ver en segundos que el

algoritmo determinista tiene los mejores tiempos, pero el método H&RS – RAGS mejora todos los pesos de las soluciones iniciales desde un 28.56% hasta un 53.60% por el costo de 3.8 segundos en las instancias más chicas y hasta 20.9 segundos en las instancias más grandes.

4.1. Tabla de resultados en costo y tiempo promedio del algoritmo determinista y el método H&RS – RAGS.

Instancia	Resultados promedio				
	Determinista	Tiempo	H&RS – RAGS	Tiempo	Mejora en peso
n030w4_1_6-2-9-1	26400	1.2 s	13195	7.0 s	50.02%
n030w4_1_6-7-5-3	25125	1.3 s	12375	7.4 s	50.75%
n030w8_1_2-7-0-9-3-6-0-6	56480	1.7 s	28825	9.9 s	48.96%
n030w8_1_6-7-5-3-5-6-2-9	55700	1.8 s	26695	11.6 s	52.07%
n040w4_0_2-0-6-1	39515	1.5 s	22655	5.6 s	42.67%
n040w4_2_6-1-0-6	32515	1.4 s	23230	5.2 s	28.56%
n040w8_0_0-6-8-9-2-6-6-4	82510	2.4 s	45725	7.8 s	44.58%
n040w8_2_5-0-4-8-7-1-7-2	85155	2.1 s	48590	7.0 s	42.94%
n050w4_0_0-4-8-7	48025	1.7 s	24950	8.4 s	48.05%
n050w4_0_7-2-7-2	47565	1.4 s	25290	7.3 s	46.83%
n050w8_1_1-7-8-5-7-4-1-8	99500	2.4 s	52750	11.7 s	46.98%
n050w8_1_9-7-5-3-8-8-3-1	100890	2.2 s	54395	11.5 s	46.08%
n060w4_1_6-1-1-5	59970	1.7 s	34190	5.4 s	42.99%
n060w4_1_9-6-3-8	61125	1.5 s	35705	6.2 s	41.59%
n060w8_0_6-2-9-9-0-8-1-3	108415	2.5 s	54265	13.6 s	49.95%

n060w8_2_1-0-3-4-0-3-9-1	110825	2.6 s	57670	16.7 s	47.96%
n080w4_2_4-3-3-3	80285	1.8 s	44470	8.0 s	44.61%
n080w4_2_6-0-4-8	79375	1.7 s	43705	8.3 s	44.94%
n080w8_1_4-4-9-9-3-6-0-5	164055	2.8 s	85515	7.9 s	47.87%
n080w8_2_0-4-0-9-1-9-6-2	167945	2.7 s	84520	7.0 s	49.67%
n100w4_0_1-1-0-8	93570	1.9 s	45130	11.9 s	51.77%
n100w4_2_0-6-4-6	94980	1.8 s	45055	12.7 s	52.56%
n100w8_0_0-1-7-8-9-1-5-4	190475	2.9 s	89435	13.7 s	53.05%
n100w8_1_2-4-7-9-3-9-2-8	193830	2.9 s	90090	12.8 s	53.52%
n120w4_1_4-6-2-6	104320	2.4 s	50150	14.2 s	51.93%
n120w4_1_5-6-9-8	101735	2.7 s	50660	12.5 s	50.20%
n120w8_0_0-9-9-4-5-1-0-3	217360	4.2 s	109275	25.1 s	49.73%
n120w8_1_7-2-6-4-5-2-0-2	216645	4.2 s	100515	25.0 s	53.60%

Para saber los tiempos limites de ejecución para cada solución del benchmark en el INRC-II se entrega un software llamado Benchmark_INRC2-Windows-x86-64.exe el cual mide los componentes del ordenador en el que el solucionador será ejecutado y entrega una lista como la presentada en la tabla 4.2 donde podemos observar los tiempos limites con un ordenador de las siguientes características:

System Information

Time of this report: 9/6/2020, 23:30:27

Operating System: Windows 10 Pro 64-bit (10.0, Build 19041) (19041.vb_release.191206-1406)

Language: English (Regional Setting: English)

System Model: TM1701

BIOS: XMAKB5R0P0502 (type: UEFI)

Processor: Intel(R) Core(TM) i7-8550U CPU @ 1.80GHz (8 CPUs), ~2.0GHz

Memory: 16384MB RAM

Available OS Memory: 16268MB RAM

Page File: 5853MB used, 12846MB available

DirectX Version: DirectX 12

DX Setup Parameters: Not found

User DPI Setting: 120 DPI (125 percent)

System DPI Setting: 120 DPI (125 percent)

Miracast: Available, with HDCP

Microsoft Graphics Hybrid: Supported

DirectX Database Version: 1.1.5

DxDiag Version: 10.00.19041.0084 64bit Unicode

Figura 4.8 Características de la computadora utilizada.




4.2. Tabla de tiempo límite para cada instancia en el equipo de computo anteriormente mencionado.

Tiempo limite de las instancia según tamaño			
30 enfermeras	104 s	60 enfermeras	339 s
40 enfermeras	182 s	80 enfermeras	496 s
50 enfermeras	261 s	100 enfermeras	653 s
60 enfermeras	339 s	120 enfermeras	809 s

4.2.3. Fase B optimización mediante vecindarios

El método de la figura 4.17 fue el tercer logro de esta investigación en el que se estudio el impacto que tiene la búsqueda de soluciones vecinas en el peso de las soluciones iniciales, este estudio fue una aplicación en los resultados individuales y en conjunto de los vecindarios, como principio se tomó en cuenta el tiempo restante para terminar la investigación entonces se propuso su implementación en un algoritmo greedy que se observa en el diagrama de flujo de la figura 4.20, pero estos vecindarios pueden aplicarse en meta-herísticas más complejas en las que también se puede estudiar su impacto, este pudo desarrollarse principalmente por la implementación de la función objetivo de la figura 4.19 anteriormente presentada en el modelo matemático y también la implementación de memoria para la conservación de las soluciones iniciales a las cuales se les evaluó en esta herramienta para poder conocer el peso inicial y los resultados obtenidos al aplicar uno o varios vecindario en una o varias repeticiones dependiendo del criterio de paro del vecindario utilizado.

Notación en las figuras

-  Asignación original
-  Modificación de la asignación
-  Intercambio de asignaciones

En la figura 4.9 podemos observar el tablero original y un ejemplos de asignación individual y en la figura 4.10 podemos observar el tablero original y un ejemplo de asignación por bloques con esto se ilustra que tipos de cambios puede recibir una enfermera n_1 donde cada uno de estos tendrá un peso de evaluación distinto según los contratos de esta, y donde se evaluarán todos los resultados de los vecindarios utilizados en el algoritmo greedy (los vecindarios con texto color rojo son los vecindarios que aportaron un mayor veneficio a la búsqueda de una mejor solución).

•Enfermera n_1

L	M	M	J	V	S	D
M						

•Asignación

L	M	M	J	V	S	D
M	M					

Figura 4.9 Ejemplo del tablero original de asignaciones de una enfermera n_1 y asignación de un solo día.

•Enfermera n_1

L	M	M	J	V	S	D
M						

• Bloque de asignación

L	M	M	J	V	S	D
M			M	M	M	

Figura 4.10 Ejemplo del tablero original de asignaciones de una enfermera n_1 y asignación de varios días.

En la figura 4.11 podemos observar ejemplos de asignación por intercambio que puede recibir la enfermera n_1 con una enfermera n_2 distinta de n_1 donde este intercambio de asignaciones tendrá un peso de evaluación distinto en cada enfermera según los contratos de estas, estos ejemplos servirán para ilustrar con claridad los vecindarios utilizados en el algoritmo greedy.

•Intercambio

-Antes

•Enfermera random n_1

L	M	M	J	V	S	D
					M	M

•Enfermera random n_2 distinta de n_1

L	M	M	J	V	S	D
M						

-Después

•Enfermera random n_1

L	M	M	J	V	S	D
					M	M



•Enfermera random n_2 distinta de n_1

L	M	M	J	V	S	D
M						



Figura 4.11 Ejemplo de intercambios entre una enfermera aleatoria n_1 y otra enfermera aleatoria n_2 (*Enfermera random n_1 y enfermera random n_2*).

En la figura 4.12 al observar el contrato de la enfermera n_1 y que días tiene libres de la semana 1 a la semana w se puede observar varios días que pueden ser cubiertos por la demanda óptima y al aplicar el vecindario tenemos un ejemplo de una nueva cobertura en la semana 1 hasta la semana w para la enfermera n_1 que será evaluada por la función objetivo.

- Enfermera n_1 , contrato descansar 3 días, semana 1 a la semana w

L	M	M	J	V	S	D
M						

- Asignación semana 2

L	M	M	J	V	S	D

⋮

- Asignación semana w

L	M	M	J	V	S	D
	M					

- Una enfermera que no trabaje el día d cuando la demanda $\hat{\Delta} > 0$ agregarla a la planeación.

- Asignación semana 1

L	M	M	J	V	S	D
M	M	M		M	M	

- Asignación semana 2

L	M	M	J	V	S	D
		M	M	M		

⋮

- Asignación semana w

L	M	M	J	V	S	D
	M	M	M	M		

Figura 4.12 Vecindario de asignaciones continuas para satisfacer la demanda óptima.

En la figura 4.13 al observar el contrato de la enfermera n_1 y que días tiene libres para la semana 1 a la semana w se puede observar varios días que pueden ser cubiertos continuamente según la duración del contrato de la enfermera n_1 y al aplicar el vecindario tenemos un ejemplo de una nueva cobertura en la semana 1 a la semana w para la enfermera n_1 que será evaluada por la función objetivo.

- Enfermera n_1 , contrato descansar 3 días, semana 1 a la semana w

L	M	M	J	V	S	D
M						

- Asignación semana 2

L	M	M	J	V	S	D

⋮

- Asignación semana w

L	M	M	J	V	S	D
						M

- Una enfermera n_1 que no trabaje el día d y aún no está saturada, agregar un día

- Asignación semana 1

L	M	M	J	V	S	D
M	M	M	M			

- Asignación semana 2

L	M	M	J	V	S	D
M	M	M	M			

⋮

- Asignación semana w

L	M	M	J	V	S	D
M	M	M				M

Figura 4.13 Vecindario de asignaciones continuas para satisfacer el contrato de la enfermera n_1 .

En la figura 4.14 podemos observar el ejemplo de una nueva asignación para la enfermera n_1 al observar su tablero original en la figura 4.12 podemos ver que días tiene libres desde la semana 1 hasta la semana w donde se puede observar varios días que pueden ser cubiertos por bloques de asignación según la duración del contrato de la enfermera n_1 y al aplicar el vecindario tenemos un ejemplo de una nueva cobertura desde la semana 1 hasta la semana w para la enfermera n_1 que será evaluada por la función objetivo.

- Enfermera n_1 , contrato descansar 3 días, semana 1 a la semana w

L	M	M	J	V	S	D
M						

- Asignación semana 2

L	M	M	J	V	S	D

⋮

- Asignación semana w

L	M	M	J	V	S	D
						M

- Asignar un bloque de días d para una enfermera n_1 no saturada

- Asignación semana 1

L	M	M	J	V	S	D
M				M	M	M

- Asignación semana 2

L	M	M	J	V	S	D
			M	M	M	M

⋮

- Asignación semana w

L	M	M	J	V	S	D
			M	M	M	M

Figura 4.14 Vecindario de asignaciones en bloque para satisfacer el contrato de la enfermera n .

En la figura 4.15 podemos observar el ejemplo de una nueva asignación para la enfermera n_1 al observar su tablero original en la parte superior podemos ver que días tiene libres en la semana w donde se puede observar varios días que pueden ser cubiertos por al algún día aleatoriamente según las condiciones del contrato de la enfermera n_1 y al aplicar el vecindario tenemos un ejemplo de una nueva cobertura en la semana w para la enfermera n_1 que será evaluada por la función objetivo.

- Una enfermera que no trabaje el día d se le asigna 1 día aleatorio en una semana w aleatoria.

- Enfermera n_1 , contrato descansar 2 días, w

L	M	M	J	V	S	D
M						M

- Asignación semana w

L	M	M	J	V	S	D
M		M				M

Figura 4.15 Vecindario de soló una asignación aleatoria.

En la figura 4.16 podemos observar el ejemplo de una nueva asignación para la enfermera n en la que al observar su tablero original en la la parte superior podemos ver que días tiene libres en la semana w donde se puede observar varios días que pueden ser cubiertos por alguna d cantidad de días aleatoriamente según las condiciones del contrato de la enfermera n_1 y al aplicar el vecindario tenemos un ejemplo de una nueva cobertura en la semana w para la enfermera n_1 que será evaluada por la función objetivo.

- Enfermera n_1 , contrato descansar 2 días, semana w

L	M	M	J	V	S	D
M						M

- Una enfermera n_1 se le asignan días aleatorios de 1 a d en los que no trabaje el día d en una semana w aleatoria.

- Asignación semana w

L	M	M	J	V	S	D
M		M			M	M

Figura 4.16 Vecindario de varias asignaciones aleatorias.

En la figura 4.17 podemos observar el ejemplo de una nueva asignación para la enfermera n en la que al observar su tablero original en la parte superior podemos ver que días cumplen con la condición de estar entre dos días ya asignados en la semana w donde se puede observar un día que cumple estas condiciones y según las condiciones del contrato de la enfermera n_1 al aplicar el vecindario tenemos un ejemplo de una nueva cobertura en la semana w para la enfermera n_1 que será evaluada por la función objetivo.

- Asignar un día d libre, rodeado por el día $d-1$ ocupado y día $d+1$ ocupado para una enfermera n_1 no saturada

- Enfermera n_1 , contrato descansar 2 días, w

L	M	M	J	V	S	D
M		M				
L	M	M	J	V	S	D
M	D	M				

Figura 4.17 Vecindario de asignaciones intermedias.

En la figura 4.18 podemos observar la herramienta necesaria para poder evaluar el peso de las soluciones vecinas, donde bajo otros criterios de aceptación se puede hacer una búsqueda de mejoras en la solución original, es así que mejorar el peso de la solución puede hacerse en criterios como el de evaluar el peso tras cada cambio o al hacer varios cambios en la solución actual y evaluar al final, debemos tomar en cuenta el tiempo que toma una evaluación ya que su aplicación al finalizar cada cambio consumirá tiempos más grandes que aplicarla después de varios cambios, es por ello que las estrategias y parámetros de cada vecindario pasos importantes en el ahorro de tiempo y mejora de costos en los pesos.

$$\begin{aligned}
\text{Minimizar } f = & 30 * \sum_{d \in D\{1...7\}} C_{d,t,sk}^{S1} \\
& + 15 * \sum_{d \in D\{1...7\}} (C_{n,t,d}^{S2a} + C_{n,t,d}^{S2b}) \\
& + 30 * \sum_{\substack{n \in N \\ d \in D\{1...7\}}} (C_{n,d}^{S2c} + C_{n,d}^{S2d}) \\
& + 30 * \sum_{\substack{n \in N \\ d \in D\{1...7\}}} (C_{n,d}^{S3a} + C_{n,d}^{S3b}) \\
& + 10 * \sum_{\substack{n \in N \\ d \in D\{1...7\}}} C_{n,d}^{S4} \\
& + 30 * \sum_{n \in N} C_n^{S5} \\
& + 20 * \sum_{n \in N} C_n^{S6} \\
& + 30 * \sum_{n \in N} C_n^{S7} \\
\text{Sujeto a } & H \{H1, H2, H3, H4\} = 0
\end{aligned}$$

Figura 4.18 Función objetivo.

De acuerdo con la función objetivo la función de evaluación para aceptar una solución de mejor peso será la siguiente:

La función F(A) nos regresará el costo de penalización de la solución actual

$$F(A) = C^{S1} + C^{S2a} + C^{S2b} + C^{S2c} + C^{S2d} + C^{S3a} + C^{S3b} + C^{S4} + C^{S5} + C^{S6} + C^{S7}$$

Por lo tanto existen dos casos para el resultado de la booleana $F(A) > F(E)$

Caso 0: El costo de F(A) no es aceptado.

Caso 1: El costo de F(A) es aceptado y las variables en F(E) toman el valor de las variables en F(A)

$$C^{E1} = C^{S1}, C^{E2a} = C^{S2a}, C^{E2b} = C^{S2b}, C^{E2c} = C^{S2c}, C^{E2d} = C^{S2d}, C^{E3a} = C^{S3a}, C^{E3b} = C^{S3b}, C^{E4} = C^{S4},$$

$$C^{E5} = C^{S5}, C^{E6} = C^{S6}, C^{E7} = C^{S7}$$

$$F(E) = C^{E1} + C^{E2a} + C^{E2b} + C^{E2c} + C^{E2d} + C^{E3a} + C^{E3b} + C^{E4} + C^{E5} + C^{E6} + C^{E7}$$

El método de la figura 4.19 fue el tercer logro de esta investigación al aplicar un método greedy para el estudio del impacto de los vecindarios en las soluciones iniciales el cual se compone de la variable B inicializada en cero, variable CostoA inicializada con un peso de un millón y de igual forma la variable CostoE, seguido de todas las variables que almacenarán los pesos de la solución original actual y la solución vecina inicializados en cero los cuales son C^{S1} , C^{S2a} , C^{S2b} , C^{S2c} , C^{S2d} , C^{S3a} , C^{S3b} , C^{S4} , C^{S5} , C^{S6} , C^{S7} y C^{E1+} , C^{E2a+} , C^{E2b+} , C^{E2c+} , C^{E2d+} , C^{E3a+} , C^{E3b+} , C^{E4+} , C^{E5+} , C^{E6+} , C^{E7} seguido de las variables Limite, Exit y R inicializadas en cero. Como paso siguiente se necesita aplicar una evaluación de la solución original/actual donde la función $F(A)$ tendrá el mejor peso actual y le seguirá la pregunta ¿ $F(A)$ es menor que $F(E)$? Para la que tenemos dos posibilidades.

- Respuesta Sí, respaldar archivo solución/actual de w semanas como la solución estrella. Donde se copiarán todos los valores en la función $F(A)$ a la función $F(E)$.
- Respuesta No, restaurar solución estrella e incrementar en uno el contador Exit.

A esta pregunta le seguirá una aplicación del vecindario bloques del cual se obtuvo los mejores resultados durante la experimentación la pregunta es ¿B es igual a cero? Para lo que tenemos dos posibilidades.

- Respuesta Sí, se aplica el vecindario bloques y se iguala el valor de B a 1.
- Respuesta No, se escoge un vecindario aleatoriamente entre 1 y 6 después se aplica el vecindario seleccionado a la solución actual e incrementamos en uno la variable Limite que representa el criterio de paro en caso de que no se deje de encontrar soluciones que mejoren la solución actual. Seguido de la pregunta ¿Limite es mayor que 500,000 o Exit es mayor que 15000? Para lo que tenemos dos posibilidades.
- Respuesta Sí, imprimimos la mejor solución estrella y se termina la ejecución del método de vecindarios.
- Respuesta No, repetimos desde la evaluación del costo de la solución actual.

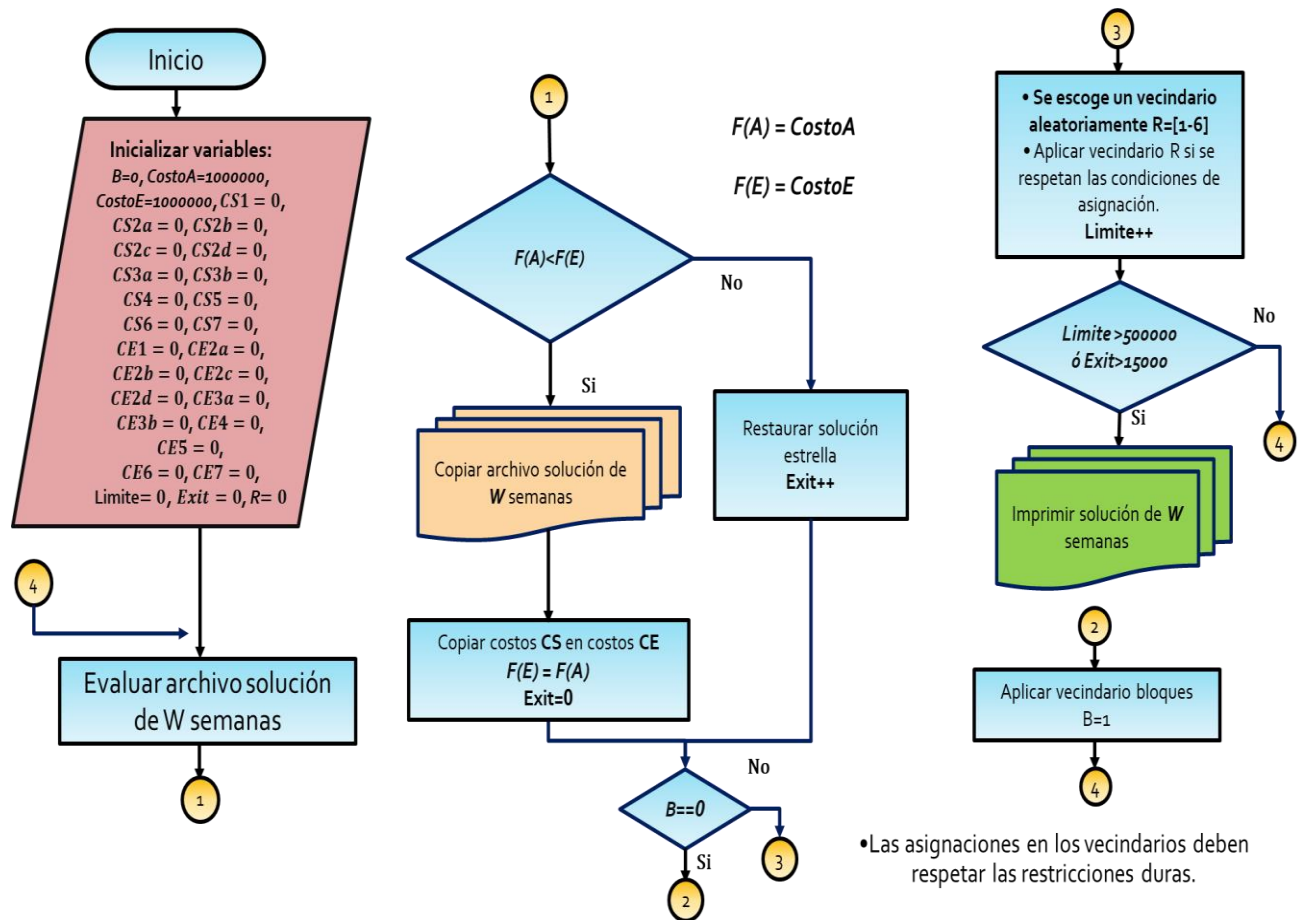


Figura 4.19 Vecindarios aplicados a la solución inicial de Hard & Random Soft - Restricted Assignments Greedy Solver, diagrama de flujo.

4.2.4. Tablas de resultados

La graficación de los resultados alcanzados por el método Hard & Random Soft - Restricted Assignments Greedy Solver se pueden observar en la figura 4.20 y muestran la dispersión de pesos de cada una de las restricciones suaves antes de la aplicación de vecindarios y para lo cual es necesario desarrollar una función objetivo que nos ayude a evaluar el peso de las nuevas soluciones, teniendo en cuenta que muchas veces durante este estudio la disminución de algunas restricciones suaves conlleva a que otras restricciones suaves sean violadas y es con esta información que se pueden crear criterios y cuando se deben tomar estrategias sobre cuales

restricciones suaves aportan más beneficios durante la optimización experimentando con diferentes vecindarios de la solución factible.

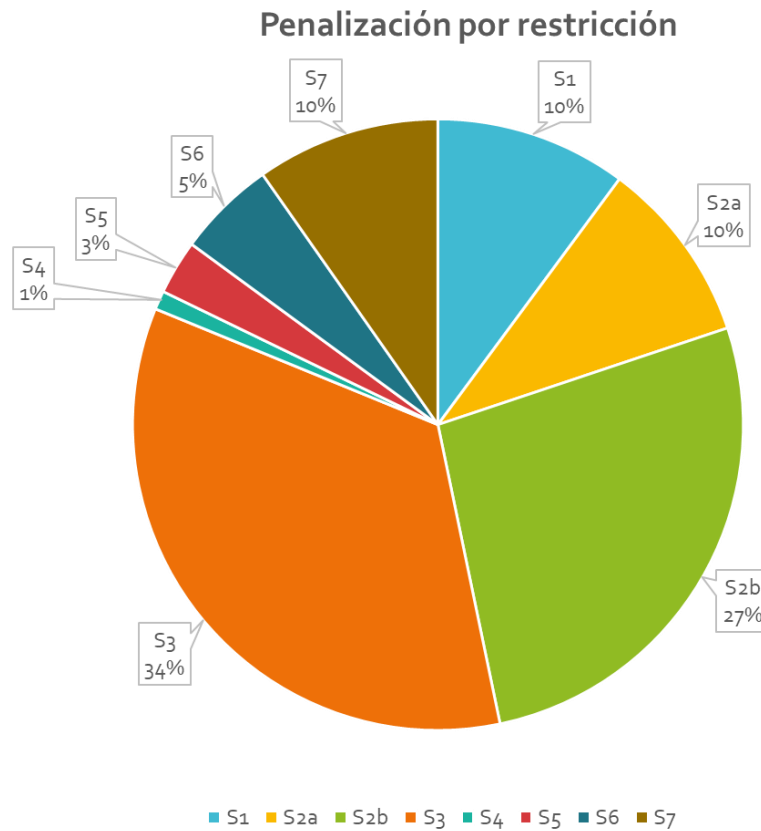


Figura 4.20 Distribución de la suma del peso de las violaciones de las restricciones suaves en la solución inicial en Hard & Random Soft - Restricted Assignments Greedy Solver antes de aplicar el método de vecindarios greedy en la instancia de 120 enfermeras n120w8_1_7-2-6-4-5-2-0-2.

En la tabla 4.3 se puede observar los mejores resultados alcanzados con los métodos H&RS – RAGS y vecindarios, donde podemos ver la mejora de los pesos que el método de vecindarios suma al método H&RS – RAGS, mejorando todos los pesos de las soluciones iniciales desde un 8.66% hasta un 39.11%.

4.3. Tabla de mejores resultados de los métodos H&RS – RAGS y vecindarios.

Instancia	Mejores resultados		
	H&RS – RAGS	Vecindarios	Mejora
n030w4_1_6-2-9-1	12245	10255	16.25%
n030w4_1_6-7-5-3	11610	10605	8.66%
n030w8_1_2-7-0-9-3-6-0-6	25035	18080	27.78%
n030w8_1_6-7-5-3-5-6-2-9	25230	17170	31.95%
n040w4_0_2-0-6-1	18975	15690	17.31%
n040w4_2_6-1-0-6	20520	16335	20.39%
n040w8_0_0-6-8-9-2-6-6-4	41365	27770	32.87%
n040w8_2_5-0-4-8-7-1-7-2	42990	26905	37.42%
n050w4_0_0-4-8-7	23345	16540	29.15%
n050w4_0_7-2-7-2	23025	16570	28.03%
n050w8_1_1-7-8-5-7-4-1-8	49240	35385	28.14%
n050w8_1_9-7-5-3-8-8-3-1	50220	36990	26.34%
n060w4_1_6-1-1-5	32755	26145	20.18%
n060w4_1_9-6-3-8	34070	27435	19.47%
n060w8_0_6-2-9-9-0-8-1-3	52930	35035	33.81%
n060w8_2_1-0-3-4-0-3-9-1	54765	37465	31.59%
n080w4_2_4-3-3-3	39410	28760	27.02%

n080w4_2_6-0-4-8	40125	29735	25.89%
n080w8_1_4-4-9-9-3-6-0-5	79655	58125	27.03%
n080w8_2_0-4-0-9-1-9-6-2	81940	59185	27.77%
n100w4_0_1-1-0-8	42265	28255	33.15%
n100w4_2_0-6-4-6	44300	27415	38.12%
n100w8_0_0-1-7-8-9-1-5-4	84985	53500	37.05%
n100w8_1_2-4-7-9-3-9-2-8	87710	53410	39.11%
n120w4_1_4-6-2-6	48080	37935	21.10%
n120w4_1_5-6-9-8	48910	36255	25.87%
n120w8_0_0-9-9-4-5-1-0-3	106080	70770	33.29%
n120w8_1_7-2-6-4-5-2-0-2	98460	73355	25.50%

La graficación de los resultados alcanzados por el método de vecindarios se pueden observar en la figura 4.21 y muestran una nueva dispersión de pesos de cada una las restricciones suaves violadas después de la aplicación de vecindarios donde el peso disminuye en todos los casos como se puede observar esto hace que los pesos de algunas restricciones tengan un decremento y otras un incremento, esto puede ser utilizado como información con la que se pueden crear nuevos criterios y nuevas estrategias sobre cuales restricciones suaves aportan más beneficios durante la optimización.

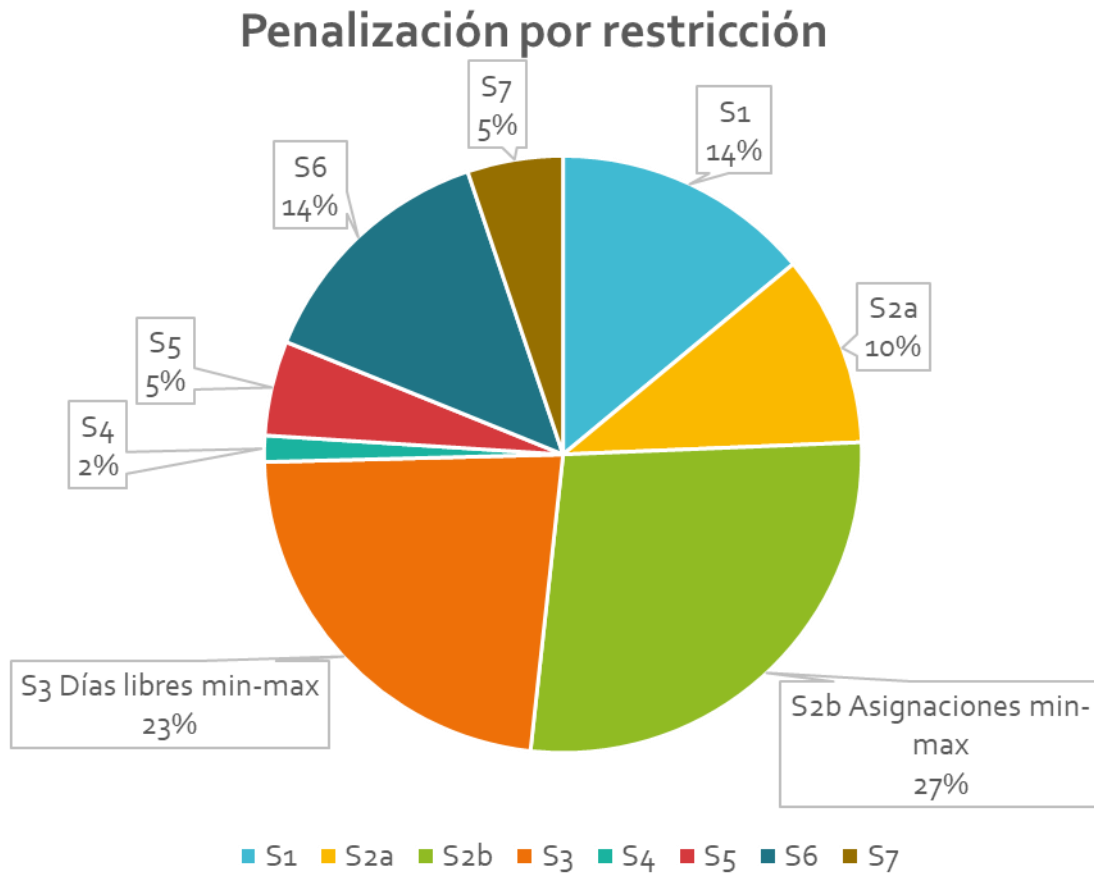


Figura 4.21 Peso de las restricciones suaves en Hard & Random Soft - Restricted Assignments Greedy Solver después de aplicar el método de vecindarios greedy en la instancia de 120 enfermeras n120w8_1_7-2-6-4-5-2-0-2.

En la tabla 4.3 se puede observar los resultados alcanzados con los métodos H&RS – RAGS y vecindarios, donde podemos ver la mejora de los pesos que el método de vecindarios suma al método H&RS – RAGS, mejorando todos los pesos de las soluciones iniciales desde un 8.66% hasta un 39.11% con tiempos de 11 segundos en las instancias más chicas y hasta 211 segundos en las instancias más grandes.

4.4. Tabla de comparación con los mejores resultados y tiempos límite.

Instancia	Peso de las soluciones			
	Literatura		Experimentación	
	Solución	Tiempo	Vecindarios	Tiempo promedio en segundos
n030w4_1_6-2-9-1	1755	104	10255	15
n030w4_1_6-7-5-3	1935	104	10605	15
n030w8_1_2-7-0-9-3-6-0-6	2340	104	18080	29
n030w8_1_6-7-5-3-5-6-2-9	1900	104	17170	21
n040w4_0_2-0-6-1	1730	182	15690	12
n040w4_2_6-1-0-6	1880	182	16335	11
n040w8_0_0-6-8-9-2-6-6-4	3310	182	27770	29
n040w8_2_5-0-4-8-7-1-7-2	3080	182	26905	31
n050w4_0_0-4-8-7	1490	261	16540	16
n050w4_0_7-2-7-2	1480	261	16570	15
n050w8_1_1-7-8-5-7-4-1-8	5410	261	35385	43
n050w8_1_9-7-5-3-8-8-3-1	5435	261	36990	48
n060w4_1_6-1-1-5	2815	339	26145	23
n060w4_1_9-6-3-8	3005	339	27435	23
n060w8_0_6-2-9-9-0-8-1-3	2765	339	35035	59
n060w8_2_1-0-3-4-0-3-9-1	3065	339	37465	56
n080w4_2_4-3-3-3	3535	496	28760	34

n080w4_2_6-0-4-8	3570	496	29735	38
n080w8_1_4-4-9-9-3-6-0-5	4995	496	58125	107
n080w8_2_0-4-0-9-1-9-6-2	5030	496	59185	93
n100w4_0_1-1-0-8	1530	653	28255	49
n100w4_2_0-6-4-6	2155	653	27415	56
n100w8_0_0-1-7-8-9-1-5-4	3195	653	53500	195
n100w8_1_2-4-7-9-3-9-2-8	3055	653	53410	211
n120w4_1_4-6-2-6	2435	809	37935	75
n120w4_1_5-6-9-8	2485	809	36255	77
n120w8_0_0-9-9-4-5-1-0-3	3615	809	70770	200
n120w8_1_7-2-6-4-5-2-0-2	3510	809	73355	116

Capítulo 5.

5. Conclusiones y trabajo futuro

5.1. Conclusiones

Se logró implementar una heurística ad-hoc de calendarización de enfermeras tomando en cuenta todas las restricciones duras, restricciones suaves, contratos, fuerza laboral para cada instancia, ordenamientos y la implementación de un criterio donde se consideran las restricciones suaves como duras a las que se les aplica una relajación creciente y aleatoria en los límites de los contratos esto permitió encontrar soluciones factibles de mejor calidad comparadas con un método determinista, estas soluciones iniciales también permiten la aplicación de vecindarios deterministas o aleatorios respetando las cuatro restricciones duras y las siete restricciones suaves es así que se logra mejorar las soluciones iniciales del método no determinista desarrollado en esta tesis para todas las instancias que componen el benchmark del INRC-II en tiempos razonables.

Con el método utilizado, entre más grande es la instancia se observa con base en los resultados obtenidos hay más probabilidad de violar las restricciones suaves y de alejarse de la mejor solución conocida en la literatura. En cuanto al tiempo de ejecución del método propuesto se logró solucionar todas las instancias del benchmark en tiempos menores a los establecidos como máximos del benchmark.

5.2. Trabajo futuro

Para mejorar la calidad de las soluciones se propone lo siguiente:

- Diseñar e implementar más vecindarios para disminuir la violación de restricciones suaves y mejorar la calidad de la solución
- Implementar la metaheurística Recocido Simulado y Búsqueda Tabú para mejorar la calidad de las soluciones
- Permitir soluciones infactibles en la búsqueda para mejorar la calidad de la solución en cuanto a las restricciones suaves violadas
- Implementar un método exacto para intentar resolver las instancias más pequeñas del problema

Referencias

- Arratia F., A. *INVESTIGACIÓN Y DOCUMENTACIÓN HISTORICA EN ENFERMERÍA*. Obtenido de [www.scielo.br: http://www.scielo.br/pdf/tce/v14n4/a14v14n4.pdf](http://www.scielo.br/pdf/tce/v14n4/a14v14n4.pdf), (2005). ISSN: 0104-0707
- Ceschia, S., Thanh Dang, N. T., De Causmaecker, P., Haspeslagh, S., & Schaerf, A. *Second International Nurse Rostering Competition (INRC-II)*. Obtenido de https://www.researchgate.net/https://www.researchgate.net/publication/271140471_Second_International_Nurse_Rostering_Competition_INRC-II_---_Problem_Description_and_Rules_---, (Enero / 2015). DOI: <https://doi.org/10.1007/s10479-018-2816-0>
- K., E. B., De Causmaecker, P., Berghe, G. V., & Landeghem, H. V. THE STATE OF THE ART OF NURSE ROSTERING. *Journal of Scheduling*, 7: 441–499, (2004). DOI: 10.1023/B:JOSH.0000046076.75950.0b
- Legrain, A., Bouarab, H., & N. L. *The nurse scheduling problem in real-life*. Obtenido de [www.researchgate.net: https://www.researchgate.net/publication/269820442_The_Nurse_Scheduling_Problem_in_Real-Life](https://www.researchgate.net/https://www.researchgate.net/publication/269820442_The_Nurse_Scheduling_Problem_in_Real-Life), (17 / Enero / 2015). DOI: <https://doi.org/10.1007/s10916-014-0160-8>
- Mischek , F., & Musliu, N. Integer Programming and Heuristic Approaches for a Multi-Stage Nurse Rostering Problem. Udine, Italia, (26 / Agosto / 2016). URL <http://www.dbai.tuwien.ac.at/staff/musliu/NurseRosteringPatat2016.pdf>
- Santos D., FernandesP., Cardoso H. L., & Oliveira E. A weighted constraint optimization approach to the nurse scheduling problem. *Computational Science and Engineering (CSE), 2015 IEEE 18th International Conference*, 233-239, (20 / 11 / 2015). DOI: 10.1109/CSE.2015.46
- Ahmed Ali El Adoly, Mohamed Gheith & M. Nashat Fors. A new formulation and solution for the nurse scheduling problem: A case study in Egypt, (2017). DOI: <https://doi.org/10.1016/j.aej.2017.09.007>
- Martin Stølevik, Tomas Eric Nordlander, Atle Riise & Helle Frøyseth. A hybrid approach for solving real-world nurse rostering problems, (2011). DOI: https://doi.org/10.1007/978-3-642-23786-7_9

Edmund K. Burke, Timothy Curtois, Gerhard Post, Rong Qu & Bart Veltman. A hybrid heuristic ordering and variable neighborhood search for the nurse rostering problem, (2007). DOI: <https://doi.org/10.1016/j.ejor.2007.04.030>

Edmund Burke, Peter Cowling, Patrick De Causmaecker & Greet Vanden Berghe. A memetic approach to the nurse rostering problem, (2001). DOI: 10.1023/A:1011291030731

Patricio Alejandro, Pablo Escalonar & Raúl Stegmaiter. Nurse Rostering Problem: Un Caso Aplicado a un Hospital en Nueva Zelanda, (2016). URL: <http://hdl.handle.net/11673/23662>

Cheang, H. Li, A. Lim & B. Rodrigues. Nurse rostering problems a bibliographic survey, (2002). DOI: [https://doi.org/10.1016/S0377-2217\(03\)00021-3](https://doi.org/10.1016/S0377-2217(03)00021-3)

Edmund Burke, Patrick De Causmaecker & Greet Vanden Berghe. A hybrid approach for solving real-world nurse rostering problems, (1998). DOI: 10.1007/3-540-48873-1_25, Issn: 0302-9743

Ziran Zheng, Xiyu Liu & Xiaoju Gong. A simple randomized variable neighbourhood search for nurse rostering, (2015). DOI: <https://doi.org/10.1016/j.cie.2017.05.027>

Masoud Rabbani, Mehrdad Niyazi. Solving a nurse rostering problem considering nurses preferences by graph theory approach, (2017). ISSN: 1735-8272

M. Rajeswari, J. Amudhavel, Sujatha Pothula & P. Dhavachelvan. Directed Bee Colony Optimization Algorithm to Solve the Nurse Rostering Problem, (2017). DOI: <https://doi.org/10.1155/2017/6563498>

Antoine Legrain, Hocine Bouarab & Nadia Lahrichi. The nurse scheduling problem in real-life, (2015). DOI: <https://doi.org/10.1007/s10916-014-0160-8>

Ruibin Bai, Edmund K. Burke, Graham Kendall, Jingpeng Li & Barry McCollum. A Hybrid Evolutionary Approach to the Nurse Rostering Problem, (2010). DOI: 10.1109/TEVC.2009.2033583

- Sanja Petrovic & Greet Vanden Berghe. Comparison of Algorithms for Nurse Rostering Problems, (2010). URL: <https://www.patatconference.org/patat2008/proceedings/Petrovic-WD3c.pdf>
- Dragan Simic, Svetlana Simic, Dragana Milutinovic & Jovanka Djordjevic. A Challenges For Nurse Rostering Problem and Opportunities in Hospital Logistics, (2014). DOI: ISSN 1642-6037
- Steven den Hartog, J.A. Hoogeveen & H.L. Bodlaender. On the Complexity of Nurse Scheduling Problems, (2016). URL: <https://dspace.library.uu.nl/bitstream/handle/1874/330858/Scriptie%20over5.pdf?sequence=2>
- Amy Witkoski Stimpfel, Douglas M. Sloane & Linda H. Aiken. The Longer The Shifts For Hospital Nurses, The Higher The Levels Of Burnout And Patient Dissatisfaction, (2019). DOI: 10.1377/hlthaff.2011.1377.
- Diogo Santos, Pedro Fernandes, Henrique Lopes Cardoso & Eugénio Oliveirai. A weighted constraint optimization approach to the nurse scheduling problem, (2015). DOI: DOI: 10.1109/CSE.2015.46
- Florian Mischek & Nysret Musliu. Integer programming and heuristic approaches for a multi-stage nurse rostering problem., (2016). URL <http://www.dbai.tuwien.ac.at/staff/musliu/NurseRosteringPatat2016.pdf>
- Jazskiewicz, Andrzej. A metaheuristic approach to multiple objective nurse scheduling - Foundations of Computing and Decision Sciences, (1997). URL <https://scholar.google.com/citations?user=CmaPeOkAAAAJ&hl=th>
- Abernathy, W. J., N. Baloff, J. C. Hershey, and S. Wandel. A three-stage manpower planning and scheduling model—A service-sector example, (1973). DOI: 10.1287/opre.21.3.693
- D. Michael Warner & Juan Prawda. A Mathematical Programming Model for Scheduling Nursing Personnel in a Hospital, (1972). <http://dx.doi.org/10.1287/mnsc.19.4.411>
- Margarida Moz & Margarida Pato. Solving the Problem of Reroostering Nurse Schedules with Hard Constraints: New Multicommodity Flow Models, (2004). DOI: 10.1023/B:ANOR.0000019104.39239.ed

Scott Steve & Ron Simpson. Case-bases incorporating scheduling constraint dimensions: Experiences in Nurse rostering, (1998). DOI: <https://doi.org/10.1007/BFb0056350>

Li H., A. Lim, & B. Rodrigues. A hybrid AI approach for Nurse rostering problem, (2003). DOI: [10.1145/952670.952675](https://doi.org/10.1145/952670.952675)

Edmund K. Burke, P. De Causmaecker, G. Vanden Bergue & H. Van Landeghem. The state of the art of Nurse rostering, (2004). DOI: <https://doi.org/10.1023/B:JOSH.0000046076.75950.0b>