

Trabajo de Fin de Carrera

Ingeniería de Organización Industrial

Planificación de quirófanos en hospitales: análisis de diferentes políticas

Autor: Irene Domínguez Galán

Tutor: Víctor Fernández-Viagas Escudero

Dpto. Organización Industrial y Gestión de Empresas I
Escuela Técnica Superior de Ingeniería
Universidad de Sevilla

Sevilla, 2022



Trabajo de Fin de Carrera
Ingeniería de Organización Industrial

Planificación de quirófanos en hospitales: análisis de diferentes políticas

Autor:

Irene Domínguez Galán

Tutor:

Víctor Fernández-Viagas Escudero

Profesor titular

Dpto. de Organización Industrial y Gestión de Empresas I

Escuela Técnica Superior de Ingeniería

Universidad de Sevilla

Sevilla, 2022

Resumen

En el siguiente documento recojo un estudio sobre optimización de la planificación de las operaciones en los quirófanos de un hospital. Para ello, comparo la política actual seguida en el Hospital Virgen del Rocío (Sevilla, España) con otros modelos de planificación diseñados a partir de esta referencia. Nuestro objetivo de la investigación es analizar diferentes políticas de planificación y encontrar la que maximice la suma de los pesos clínicos de las operaciones realizadas en un día, priorizando su planificación en los días más tempranos. Para ello, traduciremos cada una de las políticas en un modelo matemático de un problema de optimización. Este a su vez lo pasaremos a código de programación usando el lenguaje Python. Resolveremos cada uno de los modelos diseñando un banco de datos iniciales, de manera que partimos de la misma situación en cada uno de los modelos, pero obteniendo en cada caso un resultado respetando sus limitaciones propias. Una vez recogidos los resultados de cada uno de estos modelos, realizaremos una comparación entre ellos y determinaremos qué modelos podrían mejorar el implementado actualmente, con qué modelos obtendríamos un rendimiento parecido y qué modelos proporcionan unos resultados peores a los actuales. Este estudio surge de la importancia que tiene un diseño adecuado del uso de quirófano dentro de los hospitales ya que suponen un cuello de botella dentro de estos (debido a sus altos costes, su alta demanda y los grandes beneficios que sacan los propios hospitales gracias a ellos).

Resumen	v
Índice de Tablas	viii
Índice de Ilustraciones	ix
1 Introducción	11
1.1. Producción	12
1.2. Planificación VS Producción	12
1.3. Importancia sanitaria y quirúrgica	13
1.4. Similitudes entre entorno sanitario e industrial	14
1.5. Problemas de optimización	15
1.5.1. Investigación operativa	15
1.5.2. Optimización	15
1.1.3. Modelado del problema	16
2. Modelo base estudiado	17
2.1. Descripción del problema	17
2.1.1. Planificación de cirujanos	17
2.2. Modelo matemático	18
2.3. Explicación del modelo	19
2.4. Comprobación del modelo	20
2.4.1. Definición de datos de entrada	20
2.4.2. Resultados obtenidos	22
3. Modelos alternativos	24
3.1. Modelo 1: asignación de cirujanos como variable	24
3.2. Modelo 2: Cirujanos operan en un único OR	25
3.3. Modelo 3: Cirujanos operan en $U[1,J]$ ORs	27
3.4. Modelo 4: Un único cirujano para cada OR	28
3.5. Modelo 5: Cirujanos solo operan 1 día	30
4. Entorno de trabajo	32
4.1. Equipo tecnológico: Huawei MateBook D15	32
4.2. Lenguaje de programación: Python	32
4.3. Entorno de Desarrollo Integrado: PyCharm	32
4.4. Solver: Gurobi	33
5. Generación de instancias	34
5.1. Definición de los conjuntos	34
5.2. Definición de los parámetros	34
5.3. Resumen las instancias	35
6. Resultados	37
6.1. Resultados obtenidos	37
6.2. Tratamiento de los resultados	44
6.3. Análisis de los resultados	51
7. Conclusiones	52
Referencias	54
Anexo	55

ÍNDICE DE TABLAS

Tabla 1: Historial de la lista de espera del Hospital Virgen del Rocío	11
Tabla 2: Índices y conjuntos del modelo	18
Tabla 3: Parámetros del modelo	18
Tabla 4: Variables del modelo	18
Tabla 5: Valores de $r[j, h]$ del ejemplo	20
Tabla 6: Valores de $a[s, h]$ del ejemplo	21
Tabla 7: Valores de $\gamma[i]$ del ejemplo	21
Tabla 8: Valores de $rd[i]$ del ejemplo	21
Tabla 9: Valores de $d[i]$ del ejemplo	21
Tabla 10: Valores de $t[i]$ del ejemplo	21
Tabla 11: Valores de $w[i]$ del ejemplo	21
Tabla 12: Valores de $\delta[i, j, h]$ del ejemplo	21
Tabla 13: Valores de $X[i, j, h]$ del ejemplo	22
Tabla 14: Valores de $Z[s, j, h]$ del ejemplo	22
Tabla 15: Tamaño de los conjuntos en las instancias	35
Tabla 16: Valores de los parámetros en las instancias	36
Tabla 17: Valores de otros parámetros en las instancias	36
Tabla 18: Datos de entrada para generar instancias	36
Tabla 19: Tabla resumen del número de óptimos	37
Tabla 20: Óptimos encontrados (iteración 1)	38
Tabla 21: Óptimos encontrados (iteración 2)	39
Tabla 22: Valor FO (iteración 1)	40
Tabla 23: Valor FO (iteración 2)	41
Tabla 24: Tiempos de ejecución (iteración 1)	42
Tabla 25: Tiempos de ejecución (iteración 2)	43
Tabla 26: Tiempos medio de ejecución	44
Tabla 27: Comparativa entre tiempos de ejecución	44
Tabla 28: Valores promedio para RPD[1]	45
Tabla 29: Valores promedio para RDP[2]	45
Tabla 30: Valores de RPD[1] (iteración 1)	47
Tabla 31: Valores de RPD[1] (iteración 2)	48
Tabla 32: Valores de RPD[2] (iteración 1)	49
Tabla 33: Valores de RPD[2] (iteración 2)	50

ÍNDICE DE ILUSTRACIONES

Ilustración 1: Personas en lista de espera quirúrgica	13
Ilustración 2: Relación entre valor y coste	13
Ilustración 3: Comparación entre un proceso industrial y uno quirúrgico	14
Ilustración 4: Diagrama de Gantt según $r[j, h]$	23
Ilustración 5: Diagrama de Gantt según $a[s, h]$	23

1 INTRODUCCIÓN

El que solo sabe de medicina, ni medicina sabe

- José de Letamendi y Manjarrés -

El sector sanitario siempre ha estado en el punto de mira a la hora de evaluar el rendimiento de un partido político en el poder. Sin embargo, podríamos decir que, desde que comenzó la pandemia, esto se ha visto realizado. La demanda de los servicios sanitarios ha sido disparada debido al gran número de casos de Covid-19 y a la alarma expandida entre la población. Ante el miedo de estar contagiado, ha sido habitual acudir a un centro sanitario o precisar de atención de un médico vía telemática al sentir cualquier síntoma. A pesar de ello, si hay un departamento dentro de los hospitales que se ha caracterizado desde antes del comienzo de esta situación pandémica por tener una alta demanda y un problema de escasez de recursos, ese es el área quirúrgica. Es habitual que el tiempo de espera para una intervención en un hospital español se extienda hasta durar varios meses. Es un problema instaurado en todos los hospitales de nuestro país y que los gobernantes intentan afrontar una vez alcanzan el poder.

Son muchos los estudios que buscan la optimización del departamento quirúrgico intentando minimizar los costes, reducir los recursos necesitados, aumentar la productividad de estos o mejorar el promedio de pacientes intervenidos. Todos ellos hacen una aproximación de un problema real y lo transforman a un problema matemático, el cual intentan resolver con diferentes herramientas científicas. Esta metodología es la misma usada cuando se afronta un problema en el entorno industrial: utilizamos los mismos términos (productividad, recursos, optimización, ...) e incluso podemos llegar a usar las mismas herramientas (planificación, programación, ...).

En este documento estudiaremos la planificación de operaciones en los quirófanos del Hospital Virgen del Rocío de Sevilla para intentar alcanzar una posible mejora del peso clínico promedio de los pacientes intervenidos en un día del horizonte temporal.

En la web del Servicio Andaluz de Salud (Listas de espera en la Sanidad pública, últimos datos oficiales, 2022) podemos encontrar el histórico de pacientes en la lista de pacientes para intervenciones quirúrgicas de dicho hospital y la demora media de días que sufren detalladas por grupo de procedimientos y periodo temporal considerado. A continuación, se adjunta una tabla resumen donde se han recogido los principales datos:

PERIODO TEMPORAL	TOTAL DE PACIENTES	DEMORA MEDIA (DÍAS)
Diciembre 2018	14725	183
Junio 2019	15878	169
Diciembre 2019	19286	182
Junio 2020	18719	259
Diciembre 2020	14568	258
Junio 2021	11464	171
Diciembre 2021	11880	137

Tabla 1: Historial de la lista de espera del Hospital Virgen del Rocío

Vemos que el último semestre ha sido el segundo más bajo en cuanto al número total de pacientes en la espera y el primero según la demora sufrida por estos, por lo que podríamos decir que se está consiguiendo mejorar los datos que teníamos en este hospital. Sin embargo, la demora media sigue siendo de 137 días, lo que equivale a más de 4 meses, por lo tanto, nos indica que todavía hay bastante margen de mejora en estos datos.

Para nuestro estudio nos basaremos principalmente en dos anteriores: el artículo “New heuristics for planning operations rooms” (Molina-Pariente, Hans, Framinan, & Gómez-Cía, 2015) y el Trabajo de Fin de Carrera “Intervención de planificación de intervenciones en quirófanos y camas postoperatorias” (Nieto Medina, 2021).

Para investigar esta posible mejora, seguiremos el siguiente procedimiento:

- 1- Estudiaremos la política seguida actualmente en el Hospital para la asignación de las intervenciones a los quirófanos, conociendo las principales restricciones que delimitan esta planificación.
- 2- Diseñaremos diferentes políticas que podrían seguir en el hospital alterando ciertos parámetros del modelo actual o añadiendo alguna nueva variable de decisión o restricción.
- 3- Resolveremos cada uno de los modelos diseñados y del modelo de referencia y recogeremos los resultados de cada uno.
- 4- Haremos una comparación de estos resultados, sacando conclusiones sobre ellos y las posibles mejoras que se producirían si adoptásemos alguna de estas nuevas políticas en el hospital.

A continuación, veremos algunos de los puntos tratados anteriormente con mayor detenimiento y desde un punto de vista más teórico.

1.1. Producción

Normalmente el término “producción” se asocia a un proceso de fabricación industrial. Sin embargo, si comprobamos la definición de dicho término, podemos ver que no es un término únicamente aplicable a este entorno. Podemos definir la producción como la creación de productos (bienes tangibles) o de servicios (intangibles) (Vilcarromero Ruiz, 2017). Es decir, hablaremos de producción siempre que estemos creando un producto o servicio mediante un proceso industrial u otro cualquiera. Como consecuencia de esto, podríamos ver la actividad de un quirófano como la producción de operaciones.

La producción es la actividad principal de muchas empresas, por lo que su optimización es esencial para el éxito empresarial. Para conseguir un rendimiento máximo, es imprescindible la gestión de esta producción. Esto se puede definir como el conjunto de herramientas administrativas cuyo objetivo es maximizar la productividad de la empresa. Algunas de esas herramientas son la planificación y programación de las actividades a desarrollar, términos que veremos en la siguiente sección.

1.2. Planificación VS Producción

Pese a que en nuestra vida diaria es usual utilizar los términos planificación y programación como sinónimos, hay una diferencia entre ellos importante que hay que tener en cuenta en la definición del alcance de este trabajo. A la hora de abordar un proyecto de cierta dimensión, la planificación y programación de sus actividades se convierten en esenciales si se quiere reducir la aparición de errores. Deducimos, por tanto, que ambas son herramientas previas a la ejecución de un proyecto.

La planificación tiene como objetivo asignar y distribuir los recursos disponibles en el proyecto para así facilitar el alcance del objetivo. La programación se realiza tras la planificación de un proyecto y antes de su ejecución. Determina dónde y cuándo se va a realizar cada una de esas actividades (Terrazas Pastor, 2011).

Aunque la planificación suele ser una herramienta que se usa para un horizonte a medio plazo, en este estudio la vamos a aplicar a un horizonte a corto plazo. Dentro de la programación realizada, omitiremos la secuenciación de las actividades, es decir, determinaremos en qué quirófano y qué día se realizarán las intervenciones, pero no se definirá el orden en el que se llevarán estas a cabo. Esta simplificación no supondría un problema a la hora de aplicar en un entorno real la programación obtenida ya que en un hospital la cantidad de cirujanos disponibles para operar es bastante alta, por lo que no suele haber coincidencias de horarios que limiten la planificación de sus intervenciones.

1.3. Importancia sanitaria y quirúrgica

Hace dos años, en los meses de confinamiento, vivíamos rodeados de noticias donde se dejaba ver la escasez de recursos frente a la alta demanda que había en los hospitales. Pese a que muchos pueden pensar que esto fue únicamente debido a la crítica situación pandémica que estábamos viviendo, si hoy en día buscamos noticias sobre las listas de espera de los quirófanos españoles veremos cientos de titulares donde se habla del crecimiento de esta y la necesidad de ampliar los recursos. Mientras que en la Tabla 1 veíamos el histórico de la lista de espera en el Hospital Virgen del Rocío, en el gráfico siguiente podemos ver la evolución de esta desde el año 2006 hasta la actualidad a nivel nacional (Listas de espera en la Sanidad pública, últimos datos oficiales, 2022)

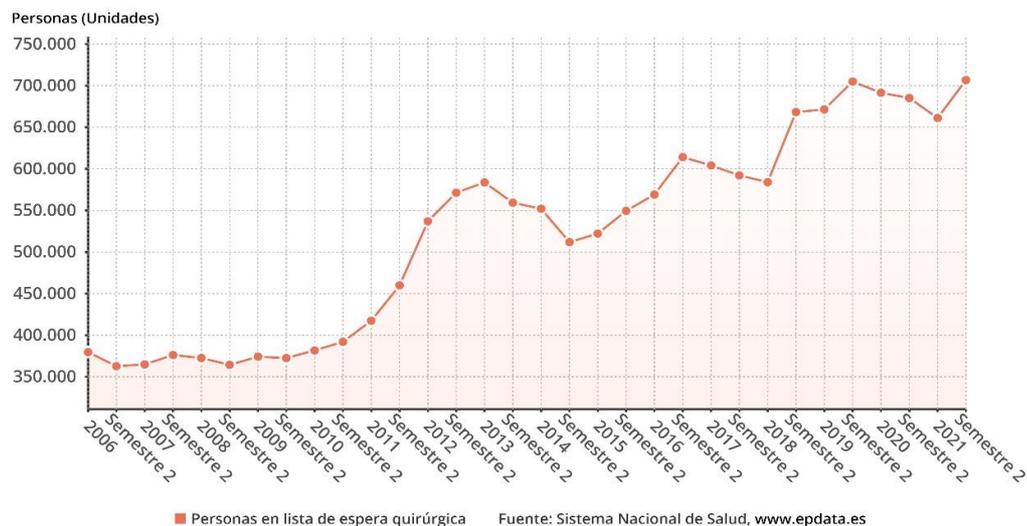


Ilustración 1: Personas en lista de espera quirúrgica

Frente a los 379.704 españoles que formaban la lista de espera a principios de 2006, a finales de 2021 eran 706.740 los que se encontraban esperando a una intervención quirúrgica. Vemos de nuevo, por tanto, que en el caso del departamento quirúrgico este problema de alta demanda de sus servicios es un problema más allá de la situación epidemiológica que vivimos.

Este área del hospital es una de las que más costes requiere (se estima que cada minuto con un quirófano operativo supone un gasto entre 15\$ y 50\$ (Rothstein & Raval, 2018), o lo que es lo mismo, entre 14,26€ y 47,54€), por lo que su uso y su planificación es una materia de estudio bastante recurrente.

Puede parecer frívolo que tratándose de la vida de personas enfermas, uno de los objetivos principales a la hora de buscar la maximización de la eficiencia de los quirófanos sea la reducción de costes que esto supone. Sin embargo, esta búsqueda de la eficiencia puede acabar en una mayor satisfacción tanto de pacientes como del personal sanitario (Rothstein & Raval, 2018):

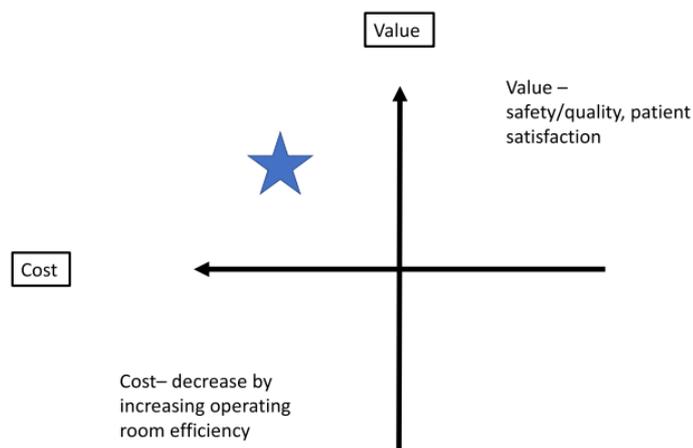


Ilustración 2: Relación entre valor y coste

En la búsqueda de la optimización de costes de los quirófanos, se obtiene una planificación de las intervenciones muy detallada en la que se ha tenido en cuenta tanto las necesidades de los cirujanos como la de los pacientes, por lo que ambos grupos suelen estar de acuerdo con las fechas obtenidas.

A la hora de realizar la planificación de los quirófanos, hay diversos factores a tener en cuenta (Rodríguez Perera & Peiró, 2012): programación de los recursos específicos necesarios (personal de enfermería, anestesia, banco de sangre, ...), utilización de los quirófanos, costes de estos, urgencia de las intervenciones, ... Según los intereses del hospital, se dará prioridad a un factor u otro a la hora de hacer dicha planificación.

No solo la cantidad de objetivos que se pueden definir en el problema determina su alta complejidad, sino también la variedad de factores inesperados que surgen en el día a día en el hospital: complicaciones en las intervenciones, demoras imprevistas durante estas, impuntualidad del personal o pacientes, ... Esto hace que se pueda tratar el problema de dos formas diferentes (Przasnyski, 1986): una más técnica, asumiendo muchas simplificaciones y restricciones teóricas, o una completamente no técnica, la cual puede aportar soluciones más cercanas a la realidad cuando tratamos un problema de menor dimensión. En nuestro caso, vamos a afrontar una resolución del problema desde una perspectiva más técnica, para la cual tendremos que hacer ciertas suposiciones y simplificaciones.

1.4. Similitudes entre entorno sanitario e industrial

Resulta común asociar todas estas herramientas de análisis como la planificación o programación a procesos industriales y, como hemos visto con anterioridad, estas herramientas no son exclusivas para el entorno industrial. Si lo analizamos desde un punto más abstracto el proceso de una producción industrial podemos ver rápidamente las similitudes con el proceso producido en un quirófano: en ambos casos partimos de unos elementos de entrada (como puede ser madera o hierro en un proceso industrial o un paciente con un órgano dañado) y a través de un proceso de transformación (llevado a cabo en una industria o en un quirófano) obtenemos el resultado final buscado (ya sea una herramienta o un paciente sanado).



Ilustración 3: Comparación entre un proceso industrial y uno quirúrgico

Si vemos el hospital como una empresa, podríamos preguntarnos qué beneficios económicos obtenemos gracias a su actividad. En el caso de los hospitales privados es más claro su beneficio, ya que los propios pacientes tienen que pagar para tener acceso a estos, pero, ¿y en el caso de los públicos? Los servicios son costeados por el Estado, por lo que los pacientes acceden gratuitamente a estos. Sin embargo, si analizamos las consecuencias de tratar a un paciente, podemos ver fácilmente cómo el Estado obtiene un beneficio económico gracias a esta actividad: se reducen los casos de invalidez (suponiendo esto una disminución de las ayudas destinadas a este sector de la población y un aumento de la población sin limitaciones para acceder al mundo laboral), una reducción del desempleo causado por enfermedades (con las mismas consecuencias que el resultado anterior) y un aumento de la producción industrial (gracias a los hechos anteriormente mencionados) (Llewelyn-Davies & Macaulay, 1966).

Además de las similitudes entre los procesos llevados a cabo en cada uno de los entornos, podemos encontrar más semejanzas en sus alcances:

- Primordial obtener gran calidad en el resultado final. En ambos procesos se procura obtener un resultado que cumpla con todos los objetivos fijados previamente. No solo para obtener un producto, una posible

venta o solventar el problema sanitario del paciente, sino además para evitar que se requiera de un uso posterior de los mismos recursos para solucionar problemas causados por esta falta de calidad.

- Necesidad de precisión a la hora de ejecutar el proceso. Para conseguir el objetivo que acabamos de comentar en el anterior punto, es necesario una alta precisión durante los diferentes pasos del proceso. Esto evitará posibles errores y retrasos en él.
- Posibilidad de tener que lidiar con factores aleatorios. A la hora de realizar ambas actividades hay que tener en cuenta la multitud de factores aleatorios que pueden afectar nuestra actividad: retrasos temporales, imprevistos en el rendimiento de los recursos, indisponibilidad de estos, ...
- Importancia del ahorro económico. Como ya hemos comentado anteriormente, los quirófanos es una de las áreas que más costes necesita de un hospital por lo que, al igual que en una industria, se busca la máxima reducción de la inversión.
- Obtención de grandes ventajas si se hace una planificación y programación de las actividades. En ambos procesos se puede llevar a cabo una planificación y programación previa a ellos para facilitar el alcance de todos los objetivos anteriormente nombrados.

1.5. Problemas de optimización

1.5.1. Investigación operativa

Podríamos definir la investigación operativa como la aplicación de métodos científicos con el objetivo de mejorar la efectividad de las operaciones, decisiones y gestiones, o lo que es lo mismo, de conseguir una satisfacción óptima de un objetivo específico deseado (Ramos, Sánchez, Ferrer, Barquín, & Linares, 2010). Gracias a esta, los responsables pueden basar sus decisiones y actuaciones en una investigación científica.

La investigación operativa tiene sus orígenes en la Segunda Guerra Mundial, cuando los dirigentes ante la escasez de productos y la urgencia por conseguir un óptimo uso de estos encargaron a científicos e ingenieros analizar los problemas militares existentes (Winston, 2005). Con esto podemos ver la variedad de campos de estudios en los que se puede aplicar la investigación operativa.

1.5.2. Optimización

Cuando estamos tratando de resolver un problema matemático, este puede tener muchas soluciones diferentes. Sin embargo, cuando nos encontramos ante un problema de optimización, no todas satisfacen el principal objetivo de este: maximizar o minimizar una función objetivo.

Podemos distinguir entre 3 elementos principales que componen un problema de optimización:

- Función objetivo: es la expresión matemática que queremos maximizar o minimizar en nuestro problema. Algunas de las más comunes son: maximizar el beneficio de un negocio, minimizar el tiempo o la distancia recorrida en un problema de transporte, ...
- Variables: representación de las decisiones a tomar. En los ejemplos anteriores algunas de las variables que podríamos tener en cuenta son el número de ventas realizadas en el negocio o una variable que recoja si el viaje entre dos localizaciones se va a realizar o no.
- Restricciones: son el conjunto de relaciones que las variables del problema están obligadas a satisfacer (Ramos, Sánchez, Ferrer, Barquín, & Linares, 2010). En el problema del negocio, una de las restricciones podría ser que el número de unidades vendidas no puede superar el stock existente en el almacén y, en el caso del problema de transporte, podríamos imponer que algunas combinaciones de localizaciones están prohibidas realizarlas.

Los métodos de optimización pueden clasificarse en métodos clásicos y métodos metaheurísticos. Entre los métodos clásicos podemos destacar la optimización lineal, lineal entera mixta, no lineal, ... mientras que en los métodos metaheurísticos están los algoritmos evolutivos, el método del recocido simulado y los sistemas multiagente. El primer grupo recibe el nombre de “clásicos” ya que son aquellos que suelen aparecer con más

frecuencia en los libros de optimización. Estos están diseñados para buscar y garantizar un óptimo local. El segundo grupo está más ligado con la inteligencia artificial e imitan fenómenos sencillos observados en la naturaleza, como es el caso de la genética. Estos métodos están diseñados para alcanzar un óptimo global aunque no está garantizada su obtención (Ramos, Sánchez, Ferrer, Barquín, & Linares, 2010).

Dentro de los modelos “clásicos” podemos encontrar diferentes clasificaciones según la naturaleza de sus restricciones (lineales o no lineales) o de sus variables (reales o continuas, enteras o discretas). Así distinguimos entre los problemas de programación lineal, de programación lineal entera mixta, de programación cuadráticas, ... (Ramos, Sánchez, Ferrer, Barquín, & Linares, 2010).

1.1.3. Modelado del problema

A la hora de resolver un problema de optimización, resulta indispensable la conversión de este en un modelo matemático. En él, veremos representadas la función objetivo, las variables y las restricciones previamente definidas.

La configuración de un problema de optimización puede secuenciarse en los siguientes pasos (Vilcarrromero Ruiz, 2017):

1. Identificación del problema

Los problemas se encuentran en la vida real definidos de manera difusa. En esta primera etapa, el instigador debe recoger toda la información relevante para el problema y traducir estos supuestos en frases precisas, convertibles más tarde en supuestos matemáticos. Este paso es fundamental ya que nos servirá para conocer el objetivo del problema y poder verificar que se ha respetado todas las limitaciones definidas.

2. Especificación matemática y formulación

Se realiza la escritura matemática del problema de optimización, definiendo sus ecuaciones, su función objetivo, sus restricciones y sus parámetros. En el caso de un modelo lineal, la elección de una formulación de un problema no afecta significativamente a la resolución del mismo. Sin embargo, si se trata de un MIP o NLP esta elección puede ser crucial.

3. Verificación, validación y refinamiento

Una vez obtenido el modelo matemático pasamos a su resolución (con o sin su conversión a lenguaje de programación). En estos primeros intentos de resolución comprobaremos si se ha producido algún error en la formulación matemática, comparando los resultados obtenidos con situaciones reales ya transcurridas o comprobando si estos son coherentes con los supuestos definidos previamente.

4. Interpretación y análisis de los resultados

Una vez conseguido el correcto funcionamiento del modelo, pasamos a hacer un análisis de sensibilidad tanto de los parámetros de entrada como de salida. Esta interpretación de los datos nos podría llevar incluso a proponer soluciones alternativas al modelo.

5. Implantación, documentación y mantenimiento

Es una etapa fundamental en el proceso de modelado. Para la correcta implantación del modelo y conseguir una difusión máxima de este, es imprescindible realizar una documentación precisa de todo el proceso seguido con indicaciones que faciliten el uso y mantenimiento del propio modelo. En esta documentación se deben incluir tanto la parte teórica como matemática e informática del modelo. El documento ante el cual nos encontramos ahora mismo sería la pieza fundamental de este paso del procedimiento, ya que es donde vamos a recoger todos los detalles del estudio realizado.

2. MODELO BASE ESTUDIADO

En este apartado vamos a conocer la política de planificación de las intervenciones en el Hospital Virgen del Rocío de Sevilla. Como hemos visto anteriormente, los problemas de optimización suelen ser aproximaciones del entorno real. En este apartado vamos a conocer todos los detalles de este modelo base estudiado: descripción del problema, modelo matemático y explicación del modelo. Todos los modelos que estudiaremos más adelante como alternativas partirán de este primero, alterando algunos de los parámetros incluidos o algunas restricciones.

2.1. Descripción del problema

En todo departamento quirúrgico de un hospital encontraremos diferentes recursos comunes. Para nuestro estudio, tendremos en cuenta los siguientes: los pacientes en la lista de espera, los cirujanos, los quirófanos y los días del horizonte temporal estudiado. Los pacientes en lista de espera para ser operados quedan recogidos en el conjunto I , usando el índice i para referirnos individualmente a cada uno de ellos. Estos serán asignados a los quirófanos (ORs) del hospital (conjunto J , índice j) y serán atendidos por un cirujano (conjunto S , índice s) en un día del plano horizontal (conjunto H , índice h).

La asignación del OR a cada paciente ha de tener en cuenta la capacidad temporal de este en cada periodo temporal (r_{jh}). Además, también tendrá que cumplirse la capacidad temporal de cada cirujano (a_{sh}). Cada paciente tendrá asignada una fecha a partir de la cual puede ser operado (rd_i) y una fecha última para llevarlo a cabo (d_i). También conocemos la duración estimada de las intervenciones (t_i) y el peso clínico asignado a estas (w_i). Actualmente, cada una de las operaciones tiene asignado de manera previa el cirujano que se va a encargar de ella. Esta relación queda recogida en nuestro problema en el parámetro γ_i , el cual será igual al índice correspondiente al cirujano encargado de intervenir al paciente i . No todos los pacientes pueden ser asignados a cualquier quirófano y en cualquier periodo temporal ya que algunos necesitan ser intervenidos en un quirófano especializado. Para conocer si es posible realizar esta planificación tenemos el parámetro binario δ_{ijh} , el cual tiene valor 1 si es posible realizar dicha combinación y valor 0 en caso contrario.

En el modelo contaremos con dos variables: X_{ijh} , que toma el valor 1 si el paciente i es operado en el OR j en el día h y 0 en el caso contrario, y Z_{sjh} , que será igual a 1 si el cirujano s es asignado a el OR j en el día h y 0 en el caso contrario.

El objetivo de nuestra planificación será maximizar la suma del peso clínico de las operaciones realizadas en el horizonte temporal estudiado, priorizando llevarlas a cabo lo más temprano posible. Para lograr esto último multiplicaremos la suma de los pesos clínicos de las intervenciones realizadas cada día por la inversa del índice de dicho día, es decir, aquellas operaciones realizadas en el día 1 estarán multiplicadas por 1, las del día 2 por $\frac{1}{2}$, las del día 3 por $\frac{1}{3}$, etc.

2.1.1. Planificación de cirujanos

Quedaría un factor por definir del problema: el número de quirófanos en el que un cirujano puede operar. Para la definición de este hay que tener en cuenta la estrategia que se sigue en el hospital de planificación (Fei, Meskens, & Chu, 2010):

- Estrategia de planificación abierta: los cirujanos pueden operar en cualquier quirófano y en cualquier tramo horario durante toda la semana laboral.
- Estrategia de planificación por bloques: se le asignarán unos bloques horarios a lo largo de la semana laboral en los cuales realizarán sus intervenciones pendientes.

- Estrategia de planificación por bloques modificados: esta se parece a la anterior, con la diferencia de que se podrán hacer modificaciones a dichos bloques según las necesidades. Estas modificaciones pueden ser de dos tipos: algunos horarios de apertura de los quirófanos serán reservados mientras que otros estarán libres, o se liberan bloques de horarios en los que los quirófanos no tienen programada ninguna intervención.

En nuestro caso no vamos a imponer ninguna limitación en este aspecto, es decir, los cirujanos van a poder operar en todos los quirófanos disponibles. Esto, junto al hecho de que pueden trabajar todos los días de la semana laboral, hace que tengamos una estrategia de planificación abierta.

2.2. Modelo matemático

A continuación, recogeremos en 3 tablas los índices y conjuntos, parámetros y variables que conformarán nuestro problema, nombrados en la sección anterior.

Índices y conjuntos

$h \in H$	Conjunto de días dentro del plano horizontal para los recursos postoperatorios
$i \in I$	Conjunto de pacientes (cirugías) en la lista de espera
$j \in J$	Conjunto de quirófanos (ORs)
$s \in S$	Conjunto de cirujanos

Tabla 2: Índices y conjuntos del modelo

Parámetros

r_{jh}	Capacidad regular de OR j en el periodo h (minutos)
a_{sh}	Capacidad regular del cirujano s en el periodo h (minutos)
γ_i	Cirujano a cargo del paciente i
rd_i	Primera fecha para operar al paciente i
d_i	Fecha última para operar al paciente i
δ_{ijh}	Parámetro binario que toma el valor 1 si el paciente i puede ser operado en OR j en el periodo h ; 0 c.c.
t_i	Duración esperada de la operación i (minutos)
w_i	Peso clínico de la operación i

Tabla 3: Parámetros del modelo

Variables

X_{ijh}	1 si el paciente i es operado en la OR j en el periodo h ; 0 c.c.
Z_{sjh}	1 si el cirujano s trabaja en la OR j en el periodo h ; 0 c.c.

Tabla 4: Variables del modelo

Una vez reconocidos todos los parámetros a tener en cuenta en nuestro problema, presentamos el modelo matemático donde veremos transformadas todas las limitaciones mencionadas anteriormente y donde estarán representados todos estos parámetros que acabamos de mencionar. Nuestro modelo matemático se compone de una función objetivo y 10 restricciones, estando basado en el artículo “New heuristics for planning operations rooms” (Molina-Pariente, Hans, Framinan, & Gómez-Cía, 2015).

$$\text{Maximizar } \sum_{h \in H} \frac{1}{h} \left(\sum_{i \in I} \sum_{j \in J} w_i \cdot X_{ijh} \right) \quad (1)$$

$$\sum_{j \in J} \sum_{h \in H} X_{ijh} \leq 1 \quad \forall i \in I \quad (2)$$

$$\sum_{j \in J} \sum_{h=1}^{rd_i-1} X_{ijh} = 0 \quad \forall i \in I \quad (3)$$

$$\sum_{j \in J} \sum_{\substack{h \in H \\ h > d_i}} X_{ijh} = 0 \quad \forall i \in I \mid d_i \leq |H| \quad (4)$$

$$\sum_{i \in I} t_i \cdot X_{ijh} \leq r_{jh} \quad \forall j \in J, \forall h \in H \quad (5)$$

$$\sum_{j \in J} \sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \leq a_{sh} \quad \forall s \in S, \forall h \in H \quad (6)$$

$$\sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \leq r_{jh} \cdot Z_{sjh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (7)$$

$$\sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \geq Z_{sjh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (8)$$

$$X_{ijh} = 0 \quad \forall i \in I, \forall j \in J, \forall h \in H \mid \delta_{ijh} = 0 \quad (9)$$

$$X_{ijh} \in \{0, 1\} \quad \forall i \in I, \forall j \in J, \forall h \in H \quad (10)$$

$$Z_{sjh} \in \{0, 1\} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (11)$$

2.3. Explicación del modelo

El objetivo de nuestro modelo será maximizar el peso clínico medio de las operaciones realizadas, priorizando su planificación en los días más tempranos posible (función objetivo (1)). Esta función objetivo estará sometida a 10 restricciones. En primer lugar, tenemos tres restricciones relacionadas con este horizonte temporal:

- Como máximo se le puede asignar un OR y un día al paciente i dentro del horizonte temporal (conjunto de restricciones (2)).

- La operación i no puede ser planificada en una fecha anterior a su fecha más temprana (rd_i) (restricciones (3)).
- Tampoco se podrá planificar en una fecha posterior a su fecha más tardía (restricciones (4)).

Las variables tendrán que cumplir dos conjuntos de restricciones de capacidad: tanto los quirófanos como los cirujanos tienen una capacidad máxima, por lo que el tiempo que un OR sea ocupado o que un cirujano pase operando no debe superar r_{jh} (ver restricciones (5)) ni a_{sh} (restricciones (6)) respectivamente.

Si un cirujano no trabaja en un quirófano y día determinados, no se asignará ninguna de sus operaciones pendientes a esta localización y, en caso contrario, si un cirujano trabaja en un quirófano y día, la suma de las duraciones de sus operaciones realizadas allí no podrá superar la capacidad del quirófano (conjunto de restricciones (7)). En caso de que un cirujano sea programado para trabajar en un quirófano y día determinados, la suma de la duración de sus operaciones asignadas ahí no podrá ser nula (restricciones (8)).

Si el paciente i no puede ser operado en el quirófano j en el periodo h ($\delta_{ijh} = 0$), X_{ijh} tomará el valor 0 (conjunto de restricciones (9)). Por último, definiremos las dos variables de nuestro modelo (X_{ijh} y Z_{sjh}) como binarias (restricciones (10) y (11), respectivamente).

2.4. Comprobación del modelo

Como veíamos en la sección “Modelado del problema”, uno de los pasos a la hora de configurar el modelo de un problema es la comprobación de este. En nuestro caso, vamos a llevar a cabo este paso aplicando el modelo matemático a un ejemplo sencillo. A continuación, veremos los datos de entrada que usaremos para ello, recogeremos los resultados obtenidos y realizaremos la comprobación de estos.

2.4.1. Definición de datos de entrada

En primer lugar, tenemos que definir el tamaño del problema, es decir, el tamaño de los conjuntos que forman nuestro modelo. Tendremos 6 pacientes en la lista de espera ($I = 6$), 2 quirófanos disponibles ($J = 2$), 2 días en el plano horizontal ($H = 2$) y 2 cirujanos trabajando ($S = 2$). Además del tamaño de cada uno de los conjuntos, debemos definir los parámetros que caracterizan a estos.

Los quirófanos estarán disponibles 150 min al día, mientras que los cirujanos podrán operar un total de 200 min. Los cirujanos serán designados a cada operación de forma aleatoria previamente a la resolución del modelo. La primera fecha posible para realizar la operación será designada de forma arbitraria entre el primer día del plano temporal y el último. De la misma forma, se definirá la fecha más tardía, teniendo en cuenta, además, que como mínimo ha de ser igual a la fecha más temprana. La posibilidad de hacer una operación en un determinado quirófano y un determinado día será definida de forma aleatoria, al igual que la duración de las operaciones (definiendo como mínimo 30 min y como máximo 90 min). Por último, el peso clínico de las operaciones también será decidido de forma arbitraria entre 1 y 5.

La resolución de este problema la llevaremos a cabo igual que en el estudio al completo. El entorno usado lo detallaremos en la sección 4 (“Entorno de trabajo”). Como veremos, traducimos el modelo matemático al lenguaje de programación Python. De esta forma generaremos los números aleatorios necesarios para la definición de algunos parámetros considerados en este problema.

Los datos obtenidos para cada uno de los parámetros son los siguientes:

r_{jh}			
$r_{00} = 150$	$r_{01} = 150$	$r_{10} = 150$	$r_{111} = 150$

Tabla 5: Valores de $r[j, h]$ del ejemplo

a_{sh}

$a_{00} = 200$	$a_{01} = 200$	$a_{10} = 200$	$a_{11} = 200$
----------------	----------------	----------------	----------------

Tabla 6: Valores de $a[s, h]$ del ejemplo

γ_i

$\gamma_0 = 0$	$\gamma_1 = 0$	$\gamma_2 = 1$	$\gamma_3 = 0$	$\gamma_4 = 1$	$\gamma_5 = 1$
----------------	----------------	----------------	----------------	----------------	----------------

Tabla 7: Valores de $\gamma[i]$ del ejemplo

rd_i

$rd_0 = 1$	$rd_1 = 1$	$rd_2 = 0$	$rd_3 = 0$	$rd_4 = 1$	$rd_5 = 0$
------------	------------	------------	------------	------------	------------

Tabla 8: Valores de $rd[i]$ del ejemplo

d_i

$d_0 = 1$	$d_1 = 1$	$d_2 = 0$	$d_3 = 1$	$d_4 = 1$	$d_5 = 0$
-----------	-----------	-----------	-----------	-----------	-----------

Tabla 9: Valores de $d[i]$ del ejemplo

t_i

$t_0 = 37$	$t_1 = 77$	$t_2 = 51$	$t_3 = 87$	$t_4 = 76$	$t_5 = 75$
------------	------------	------------	------------	------------	------------

Tabla 10: Valores de $t[i]$ del ejemplo

w_i

$w_0 = 5$	$w_1 = 4$	$w_2 = 5$	$w_3 = 2$	$w_4 = 3$	$w_5 = 3$
-----------	-----------	-----------	-----------	-----------	-----------

Tabla 11: Valores de $w[i]$ del ejemplo

δ_{ijh}

$\delta_{000} = 0$	$\delta_{001} = 1$	$\delta_{010} = 0$	$\delta_{011} = 0$
$\delta_{100} = 0$	$\delta_{101} = 0$	$\delta_{110} = 1$	$\delta_{111} = 0$
$\delta_{200} = 1$	$\delta_{201} = 0$	$\delta_{210} = 0$	$\delta_{211} = 1$
$\delta_{300} = 1$	$\delta_{301} = 0$	$\delta_{310} = 1$	$\delta_{311} = 0$
$\delta_{400} = 0$	$\delta_{401} = 1$	$\delta_{410} = 1$	$\delta_{411} = 0$
$\delta_{500} = 1$	$\delta_{501} = 0$	$\delta_{510} = 0$	$\delta_{511} = 1$

Tabla 12: Valores de $\delta[i, j, h]$ del ejemplo

Antes de pasar a la recogida de los resultados de la resolución de este problema y al análisis de estos, vamos a sacar ciertas conclusiones de los parámetros definidos:

- Comparando las fechas más tempranas y tardías de cada operación, vemos que las operaciones $i = 2$ e $i = 5$ han de ser llevadas a cabo en el día $h = 0$, las operaciones $i = 0$, $i = 1$ e $i = 4$ en el día $h = 1$ y la operación $i = 3$ podrá ser realizada en ambos días.
- Combinando las conclusiones anteriores con los valores obtenidos para el parámetro δ_{ijh} , vemos que la operación $i = 1$ no va a poder llevarse a cabo ya que mientras que los datos de sus fechas la obligan a ser programada en el día $h = 1$, no tiene ningún valor de δ_{ijh} igual a 1 para ese día.

2.4.2. Resultados obtenidos

Una vez asignado todos los parámetros necesarios para la resolución del problema, podemos resolver este. A continuación, detallaremos los resultados obtenidos para la función objetivo (FO) y las variables:

- $FO = 14$
- Variable X_{ijh} :

$X_{000} = 0$	$X_{001} = 1$	$X_{010} = 0$	$X_{011} = 0$
$X_{100} = 0$	$X_{101} = 0$	$X_{110} = 0$	$X_{111} = 0$
$X_{200} = 1$	$X_{201} = 0$	$X_{210} = 0$	$X_{211} = 0$
$X_{300} = 0$	$X_{301} = 0$	$X_{310} = 1$	$X_{311} = 0$
$X_{400} = 0$	$X_{401} = 1$	$X_{410} = 0$	$X_{411} = 0$
$X_{500} = 1$	$X_{501} = 0$	$X_{510} = 0$	$X_{511} = 0$

Tabla 13: Valores de $X[i, j, h]$ del ejemplo

- Variable Z_{sjh} :

$Z_{000} = 0$	$Z_{001} = 1$	$Z_{010} = 1$	$Z_{011} = 0$
$Z_{100} = 1$	$Z_{101} = 1$	$Z_{110} = 0$	$Z_{111} = 0$

Tabla 14: Valores de $Z[s, j, h]$ del ejemplo

Vemos que todas las operaciones han sido planificadas como máximo 1 vez en horizonte temporal, cumpliendo así (2). Volviendo a las conclusiones sacadas a partir de los parámetros d_i y rd_i , comprobamos que se han cumplido para todas las operaciones, satisfaciendo así las restricciones (3) y (4). Además, todas las intervenciones asociadas a cada cirujano están planificadas en quirófanos donde ellos operan y respetando la capacidad temporal de estos, satisfaciendo las restricciones (7) y (8). Por último, vemos que en todos los casos donde $\delta_{ijh} = 0$, $X_{ijh} = 0$, como dicta (9) (cumpliéndose así por tanto la suposición n° 1 realizada en el apartado anterior).

Para comprobar el cumplimiento de las restricciones (5) y (6) representaremos las operaciones en unos diagramas de Gantt. En primer lugar, haremos la representación usando como tiempo límite r_{jh} .

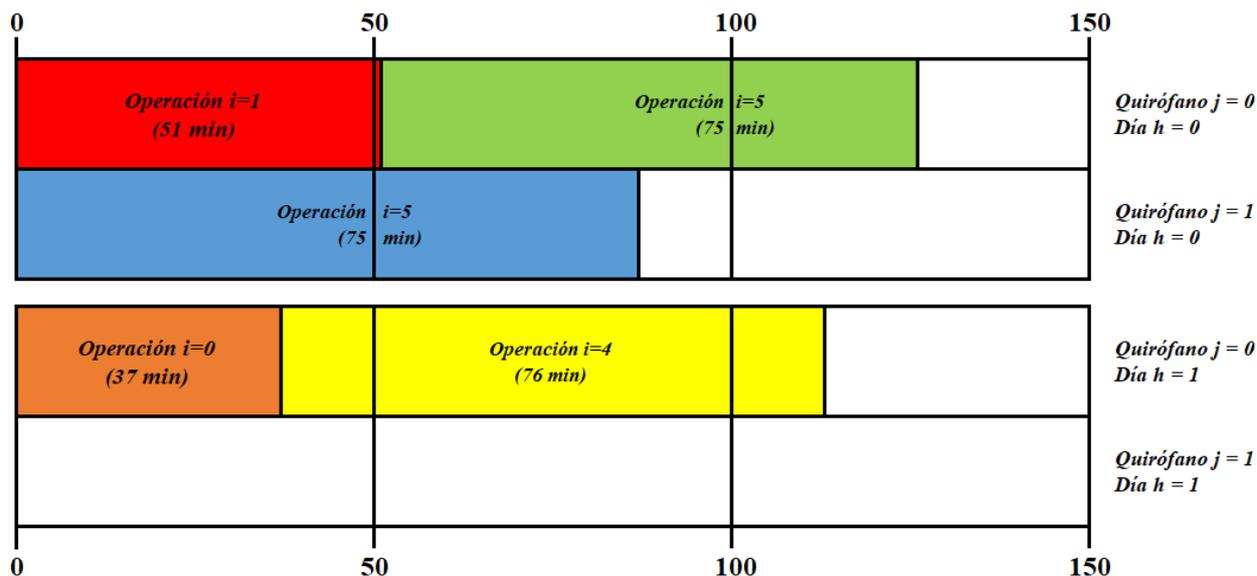


Ilustración 4: Diagrama de Gantt según $r[j, h]$

Vemos que en ningún caso se alcanza el valor máximo de uso de quirófanos, por lo que se satisface el conjunto de restricciones (5). Si usamos como tiempo límite a_{sh} podremos estudiar el cumplimiento de las restricciones (6). Solo representaremos en el diagrama para aquellos cirujanos que vayan a realizar alguna intervención.

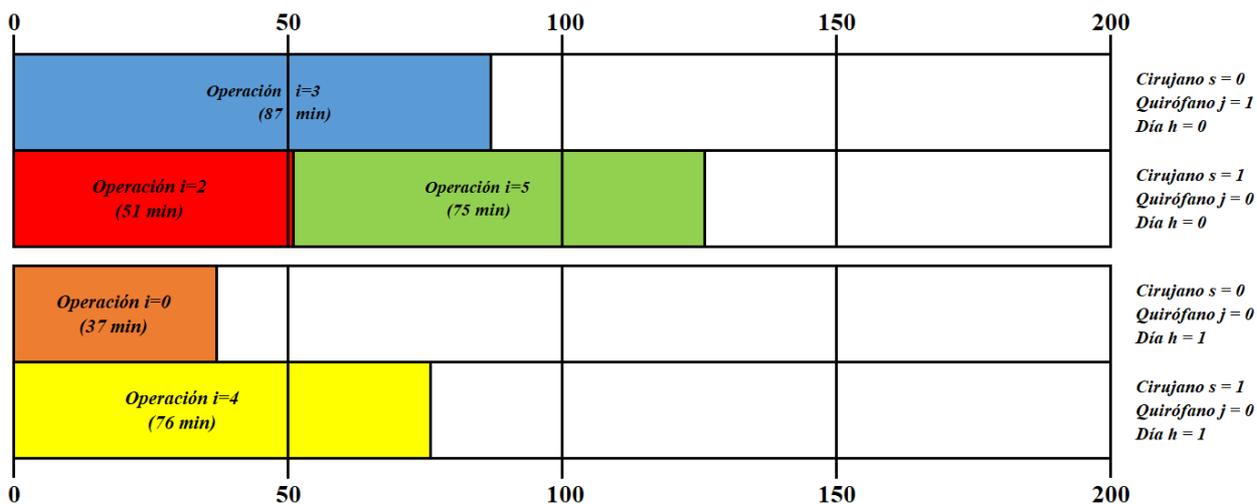


Ilustración 5: Diagrama de Gantt según $a[s, h]$

Comprobamos que en ninguno de los 4 casos se sobrepasa el límite temporal de los cirujanos y, por tanto, se satisface (6).

Por último, vemos que todas las variables han tomado valores binarios, cumpliendo las restricciones (10) y (11). Por tanto, hemos comprobado que las variables obtenidas cumplen todas las restricciones impuestas a la función objetivo.

3. MODELOS ALTERNATIVOS

Con el objetivo de conocer qué políticas ayudarían al hospital a maximizar sus resultados en el área quirúrgica, estudiaremos diferentes alternativas al Modelo Base para poder comparar los resultados obtenidos. En total estudiaremos 5 modelos, los cuales estarán formados por el Modelo Base más alguna variación en la definición de un parámetro, restricción o variable.

A continuación vamos a ver estos modelos, añadiendo al final de cada uno un breve resumen de las modificaciones realizadas respecto al Modelo base y el modelo matemático usado.

3.1. Modelo 1: asignación de cirujanos como variable

Una posible nueva política del hospital sería liberar la asignación de un cirujano a una intervención determinada. Actualmente, cada paciente tiene asignado a un cirujano que será el encargado de operarle. Sin embargo, podríamos dejar esto como una variable de manera que esta se adecuase para conseguir maximizar el número de operaciones realizadas.

Esto en el modelo matemático lo vemos traducido como la sustitución del parámetro γ_i por una nueva variable binaria (Y_{is}). Esta variable tomará el valor de 1 si la operación i es realizada por el cirujano s y 0 en el caso contrario. Debido a la introducción de esta nueva variable, tendremos que realizar ciertas modificaciones a las restricciones del Modelo Base e incorporar algunas nuevas.

Modificaciones respecto al Modelo base

- Creación de una nueva variable y su respectivo conjunto de restricciones: Y_{is} y ec. (25). Estas restricciones definen la condición binaria de la variable.
- Eliminación de un parámetro: γ_i .
- Modificación del conjunto de restricciones (6): ec. (17). El producto $t_i \cdot X_{ijh}$ solo se sumará si el cirujano s es el encargado de dicha operación, es decir, si $Y_{is} = 1$.
- Modificación de las restricciones (7): ec. (18). Esto se realiza con el mismo objetivo que la anterior.
- Modificación de las restricciones (8): ec. (19). Con el mismo objetivo.
- Nuevo conjunto de restricciones: ec. (20). Con ella nos aseguramos que si una operación es planificada en el quirófano j el día h , se le asignará un cirujano que trabaje en ese mismo quirófano y día.
- Nuevo conjunto de restricciones: ec. (21). En los casos que intervención y cirujano no sean planificados para, respectivamente, llevarse a cabo y trabajar en el mismo quirófano y día, la variable Y_{is} será nula.

Modelo matemático

$$\text{Maximizar } \sum_{h \in H} \frac{1}{h} \left(\sum_{i \in I} \sum_{j \in J} w_i \cdot X_{ijh} \right) \quad (12)$$

$$\sum_{j \in J} \sum_{h \in H} X_{ijh} \leq 1 \quad \forall i \in I \quad (13)$$

$$\sum_{j \in J} \sum_{h=1}^{rd_i-1} X_{ijh} = 0 \quad \forall i \in I \quad (14)$$

$$\sum_{j \in J} \sum_{\substack{h \in H \\ h > d_i}} X_{ijh} = 0 \quad \forall i \in I \mid d_i \leq |H| \quad (15)$$

$$\sum_{i \in I} t_i \cdot X_{ijh} \leq r_{jh} \quad \forall j \in J, \forall h \in H \quad (16)$$

$$\sum_{j \in J} \sum_{i \in I} t_i \cdot X_{ijh} \cdot Y_{is} \leq a_{sh} \quad \forall s \in S, \forall h \in H \quad (17)$$

$$\sum_{i \in I} t_i \cdot X_{ijh} \cdot Y_{is} \leq r_{jh} \cdot Z_{sjh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (18)$$

$$\sum_{i \in I} t_i \cdot X_{ijh} \cdot Y_{is} \geq Z_{sjh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (19)$$

$$\sum_{s \in S} Z_{sjh} \cdot Y_{is} = X_{ijh} \quad \forall i \in I, \forall j \in J, \forall h \in H \quad (20)$$

$$\sum_{j \in J} \sum_{h \in H} X_{ijh} \cdot Z_{sjh} \geq Y_{is} \quad \forall i \in I, \forall s \in S \quad (21)$$

$$X_{ijh} = 0 \quad \forall i \in I, \forall j \in J, \forall h \in H \mid \delta_{ijh} = 0 \quad (22)$$

$$X_{ijh} \in \{0, 1\} \quad \forall i \in I, \forall j \in J, \forall h \in H \quad (23)$$

$$Z_{sjh} \in \{0, 1\} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (24)$$

$$Y_{is} \in \{0, 1\} \quad \forall i \in I, \forall s \in S \quad (25)$$

3.2. Modelo 2: Cirujanos operan en un único OR

Mientras que la política seguida actualmente dicta que los cirujanos pueden operar en todos los quirófanos disponibles en el día (siguiendo una estrategia de planificación abierta), podríamos cambiar esto y hacer que los cirujanos únicamente puedan operar en un quirófano al día (convirtiendo la situación en una estrategia de planificación por bloques).

Gracias a este cambio, los profesionales estarían adaptados a su entorno de trabajo y evitarían un posible agotamiento por el cambio de quirófanos. Los quirófanos suelen estar situados cerca uno de otro, por lo que este cansancio no es causado principalmente por el desplazamiento entre ellos, sino por el protocolo que deben seguir siempre que salen o entran a un quirófano. No solo beneficiaría al cirujano, sino al correcto estado del quirófano. Gracias a la reducción de entradas y salidas de él, estaríamos disminuyendo las posibilidades de introducir algún agente infeccioso en su interior.

Para plasmar este cambio en nuestro modelo matemático, tendríamos que crear un nuevo conjunto de restricciones que definiese como máximo 1 el número de quirófano donde puede intervenir cada cirujano.

Modificaciones respecto al Modelo base

- Nuevas restricciones: ec. (34). Limitamos el número de quirófanos donde interviene cada cirujano cada día a 1.

Modelo matemático

$$\text{Maximizar } \sum_{h \in H} \frac{1}{h} \left(\sum_{i \in I} \sum_{j \in J} w_i \cdot X_{ijh} \right) \quad (26)$$

$$\sum_{j \in J} \sum_{h \in H} X_{ijh} \leq 1 \quad \forall i \in I \quad (27)$$

$$\sum_{j \in J} \sum_{h=1}^{rd_i-1} X_{ijh} = 0 \quad \forall i \in I \quad (28)$$

$$\sum_{j \in J} \sum_{\substack{h \in H \\ h > d_i}} X_{ijh} = 0 \quad \forall i \in I \mid d_i \leq |H| \quad (29)$$

$$\sum_{i \in I} t_i \cdot X_{ijh} \leq r_{jh} \quad \forall j \in J, \forall h \in H \quad (30)$$

$$\sum_{j \in J} \sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \leq a_{sh} \quad \forall s \in S, \forall h \in H \quad (31)$$

$$\sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \leq r_{jh} \cdot Z_{sjh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (32)$$

$$\sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \geq Z_{sjh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (33)$$

$$\sum_{j \in J} Z_{sjh} \leq 1 \quad \forall s \in S, \forall h \in H \quad (34)$$

$$X_{ijh} = 0 \quad \forall i \in I, \forall j \in J, \forall h \in H \mid \delta_{ijh} = 0 \quad (35)$$

$$X_{ijh} \in \{0, 1\} \quad \forall i \in I, \forall j \in J, \forall h \in H \quad (36)$$

$$Z_{sjh} \in \{0, 1\} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (37)$$

3.3. Modelo 3: Cirujanos operan en U[1,J] ORs

Las condiciones y necesidades de cada cirujano pueden variar y, como consecuencia, no todos se adaptan con la misma facilidad a su entorno de trabajo ni requieren operar en el mismo número de quirófanos. Definiendo este número según las necesidades de cada cirujano, estaríamos adaptando nuestra planificación a los requerimientos de nuestros trabajadores, lo que podría derivar en un aumento de su satisfacción y, como consecuencia, en una mejora de su rendimiento. En esta nueva política, estaríamos siguiendo una estrategia de planificación por bloques al igual que en el Modelo 2, ya que, pese a que algunos cirujanos sí van a poder operar en todas las localizaciones y tramos temporales posibles, algunos tendrán el horario más acotado a un solo bloque.

Para definir esta limitación, crearemos un nuevo parámetro donde recogeremos el número de quirófanos donde puede operar cada cirujano. Este parámetro lo denominaremos u_s . Además, será necesario añadir un nuevo conjunto de restricciones al Modelo Base igual al añadido en el Modelo 3, pero poniendo como límite superior u_s en vez de 1. Para nuestro estudio, haremos que este parámetro tome un valor aleatorio según una distribución uniforme $U[1, J]$. En un entorno real de trabajo este número se puede definir acorde a cada cirujano.

Modificaciones respecto al Modelo base

- Nuevo parámetro: $u_s = U[1, J] \quad \forall s \in S$
- Nuevas restricciones: ec. (46). Definimos como número máximo de quirófanos accesibles para cada cirujano u_s .

Modelo matemático

$$\text{Maximizar} \sum_{h \in H} \frac{1}{h} \left(\sum_{i \in I} \sum_{j \in J} w_i \cdot X_{ijh} \right) \quad (38)$$

$$\sum_{j \in J} \sum_{h \in H} X_{ijh} \leq 1 \quad \forall i \in I \quad (39)$$

$$\sum_{j \in J} \sum_{h=1}^{rd_i-1} X_{ijh} = 0 \quad \forall i \in I \quad (40)$$

$$\sum_{j \in J} \sum_{\substack{h \in H \\ h > d_i}} X_{ijh} = 0 \quad \forall i \in I \mid d_i \leq |H| \quad (41)$$

$$\sum_{i \in I} t_i \cdot X_{ijh} \leq r_{jh} \quad \forall j \in J, \forall h \in H \quad (42)$$

$$\sum_{j \in J} \sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \leq a_{sh} \quad \forall s \in S, \forall h \in H \quad (43)$$

$$\sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \leq r_{jh} \cdot Z_{sjh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (44)$$

$$\sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \geq Z_{sjh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (45)$$

$$\sum_{j \in J} Z_{sjh} \leq u_s \quad \forall s \in S, \forall h \in H \quad (46)$$

$$X_{ijh} = 0 \quad \forall i \in I, \forall j \in J, \forall h \in H | \delta_{ijh} = 0 \quad (47)$$

$$X_{ijh} \in \{0, 1\} \quad \forall i \in I, \forall j \in J, \forall h \in H \quad (48)$$

$$Z_{sjh} \in \{0, 1\} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (49)$$

3.4. Modelo 4: Un único cirujano para cada OR

En el Modelo Base estudiado podemos observar que los cirujanos no tienen una restricción que defina en qué quirófano pueden operar y en cuál no. Sin embargo, si asignásemos un único cirujano al día a cada quirófano podríamos observar ciertos beneficios para nuestro profesional. En el Modelo 4 no solo consideraremos exclusividad del cirujano hacia el quirófano (como ocurre en el Modelo 2), sino que esta exclusividad será recíproca. Cada cirujano operará únicamente en un quirófano al día y será el único en intervenir en dicha localización. De esta forma, volveríamos a estar ante un caso de planificación por bloques.

Al no cambiar de quirófano durante su día laboral, el cirujano podría organizar los recursos dentro de este de manera óptima para su trabajo. Además, evitaríamos de nuevo ciertas desventajas producidas por el cambio de quirófano (como ya hemos comentado en el Modelo 2).

Para trasladar este cambio al modelo matemático es necesario incorporar estos dos conjuntos de restricciones.

Modificaciones respecto al Modelo Base

- Nuevo conjunto de restricciones: ec. (58). Limitamos a 1 el número de quirófanos donde puede operar un cirujano en el mismo día.
- Nuevas restricciones: ec. (59). Igualamos a 1 el número de cirujanos que operan en un quirófano en el mismo día, evitando que este se quede vacío o que sea utilizado por más de 1 cirujano.

Modelo matemático

$$\text{Maximizar} \sum_{h \in H} \frac{1}{h} \left(\sum_{i \in I} \sum_{j \in J} w_i \cdot X_{ijh} \right) \quad (50)$$

$$\sum_{j \in J} \sum_{h \in H} X_{ijh} \leq 1 \quad \forall i \in I \quad (51)$$

$$\sum_{j \in J} \sum_{h=1}^{rd_i-1} X_{ijh} = 0 \quad \forall i \in I \quad (52)$$

$$\sum_{j \in J} \sum_{\substack{h \in H \\ h > d_i}} X_{ijh} = 0 \quad \forall i \in I \mid d_i \leq |H| \quad (53)$$

$$\sum_{i \in I} t_i \cdot X_{ijh} \leq r_{jh} \quad \forall j \in J, \forall h \in H \quad (54)$$

$$\sum_{j \in J} \sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \leq a_{sh} \quad \forall s \in S, \forall h \in H \quad (55)$$

$$\sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \leq r_{jh} \cdot Z_{sjh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (56)$$

$$\sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \geq Z_{sjh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (57)$$

$$\sum_{j \in J} Z_{sjh} \leq 1 \quad \forall s \in S, \forall h \in H \quad (58)$$

$$\sum_{s \in S} Z_{sjh} = 1 \quad \forall j \in J, \forall h \in H \quad (59)$$

$$X_{ijh} = 0 \quad \forall i \in I, \forall j \in J, \forall h \in H \mid \delta_{ijh} = 0 \quad (60)$$

$$X_{ijh} \in \{0, 1\} \quad \forall i \in I, \forall j \in J, \forall h \in H \quad (61)$$

$$Z_{sjh} \in \{0, 1\} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (62)$$

3.5. Modelo 5: Cirujanos solo operan 1 día

Todos los modelos hasta ahora estudiados han considerado que los cirujanos no tenían limitación en el número de días que pueden trabajar a la semana. En este nuevo modelo los cirujanos podrán trabajar únicamente un día a la semana. Aunque podría parecer una política bastante restrictiva, si tenemos en cuenta la cantidad de carga laboral que tienen los cirujanos fuera de los quirófanos, podríamos ver que sería de utilidad. Los cirujanos concentrarían en este día todas sus intervenciones pendientes y, además, les ayudaría a la hora de concentración ya que durante un día a la semana toda su atención se centraría en el quirófano.

Para conocer si un cirujano trabaja un día o no crearemos una nueva variable binaria (W_{sh}), la cual tomará el valor 1 si el cirujano s opera el día h y 0 en el caso contrario. Tendremos que añadir otros 4 conjuntos de restricciones más para completar el nuevo modelo.

Modificaciones respecto al Modelo base

- Nueva variable y su restricción de signo: W_{sh} y ec. (77). Como hemos visto anteriormente, W_{sh} será igual a 1 si el cirujano s va a realizar una operación durante el día h y 0 en el caso contrario.
- Nuevas restricciones: ec. (71). Esta restricción hará que en los días que el cirujano no trabaje no tenga asignado ocupar algún quirófano.
- Nuevas restricciones: ec. (72). Con esta desigualdad nos aseguraremos que si el cirujano s trabaja el día h , tenga planificado operar como mínimo en un quirófano ese día.
- Nuevas restricciones: ec. (73). Definimos el límite superior de días operativos de cada cirujano a 1.

Modelo matemático

$$\text{Maximizar } \sum_{h \in H} \frac{1}{h} \left(\sum_{i \in I} \sum_{j \in J} w_i \cdot X_{ijh} \right) \quad (63)$$

$$\sum_{j \in J} \sum_{h \in H} X_{ijh} \leq 1 \quad \forall i \in I \quad (64)$$

$$\sum_{j \in J} \sum_{h=1}^{rd_i-1} X_{ijh} = 0 \quad \forall i \in I \quad (65)$$

$$\sum_{j \in J} \sum_{\substack{h \in H \\ h > d_i}} X_{ijh} = 0 \quad \forall i \in I \mid d_i \leq |H| \quad (66)$$

$$\sum_{i \in I} t_i \cdot X_{ijh} \leq r_{jh} \quad \forall j \in J, \forall h \in H \quad (67)$$

$$\sum_{j \in J} \sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \leq a_{sh} \quad \forall s \in S, \forall h \in H \quad (68)$$

$$\sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \leq r_{jh} \cdot Z_{sjh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (69)$$

$$\sum_{\substack{i \in I \\ \gamma_i = s}} t_i \cdot X_{ijh} \geq Z_{sjh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (70)$$

$$Z_{sjh} \leq W_{sh} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (71)$$

$$\sum_{j \in J} Z_{sjh} \geq W_{sh} \quad \forall s \in S, \forall h \in H \quad (72)$$

$$\sum_{h \in H} W_{sh} \leq 1 \quad \forall s \in S \quad (73)$$

$$X_{ijh} = 0 \quad \forall i \in I, \forall j \in J, \forall h \in H \mid \delta_{ijh} = 0 \quad (74)$$

$$X_{ijh} \in \{0, 1\} \quad \forall i \in I, \forall j \in J, \forall h \in H \quad (75)$$

$$Z_{sjh} \in \{0, 1\} \quad \forall s \in S, \forall j \in J, \forall h \in H \quad (76)$$

$$W_{sh} \in \{0, 1\} \quad \forall s \in S, \forall h \in H \quad (77)$$

4. ENTORNO DE TRABAJO

Antes de realizar la experimentación de los modelos planteados y llevar a cabo el análisis de estos, es necesario hacer algunas decisiones previas: diseño del entorno informático que vamos a usar para la optimización de los modelos. Para resolver los diferentes modelos matemáticos que hemos obtenido, será necesario su traducción usando un lenguaje de programación y un IDE (Entorno de Desarrollo Integrado). Una vez obtenido dicho código, este se ejecutará usando un solver instalado previamente en el equipo.

4.1. Equipo tecnológico: Huawei MateBook D15

Para desarrollar el código del modelo, su ejecución y, por tanto, completar la resolución de los modelos estudiados, necesitamos un equipo tecnológico. Este será un portátil Huawei MateBook D15, el cual está compuesto por un procesador Intel® Core™ i3-10110U CPU 2,10 GHz 2,59 GHz, una memoria Ram de 8,00 GB y un sistema operativo de 64 bits.

4.2. Lenguaje de programación: Python

Hoy en día los programas informáticos son considerados como indispensables a la hora de llevar a cabo la mayoría de actividades profesionales. Esto justifica el nacimiento y desarrollo de múltiples lenguajes de programación: C++, Python, Java, C#, ... Para la elaboración de nuestro código hemos elegido Python. Este se caracteriza por su sintaxis sencilla, la gran cantidad de librerías disponibles y su parecido al lenguaje natural (González Duque, 2011).

Algunas de las funciones usadas a lo largo del código son:

- *if*: esta sentencia va acompañada de una condición que se ha de cumplir para ejecutar las líneas situadas dentro de ella. En muchos casos, va seguida de la sentencia *else*. En esta no hace falta definir ninguna condición, ya que se ejecuta siempre que no se cumpla la condición del *if*.
- *for*: recorre los elementos de un objeto iterable y se ejecutan las líneas que siguen a la sentencia en cada una de estas iteraciones. En su definición es muy común el uso del operador de pertenencia *in*. Este comprueba si el dato que la precede está contenido en una secuencia asociada. Este operador tiene otro “antónimo” (*not in*) que comprueba el caso contrario.
- *while*: es una sentencia de tipo bucle. Va acompañada de una condición y un código de bloque. La condición es evaluada y, en caso de ser verdadera, se ejecuta dicho código de bloque repetidamente hasta que la condición sea falsa.
- *range(a)*: función que devuelve una secuencia de números, empezando esta por defecto en 0, con un paso de 1 y terminando antes del número indicado entre paréntesis (*a*). Esta función es de gran utilidad para acompañar la función *for* ya que esta requiere elementos iterables.

4.3. Entorno de Desarrollo Integrado: PyCharm

Para hacer más fácil la elaboración del código decidí usar un Entorno de Desarrollo Integrado (IDE). Estos proporcionan un entorno de programación que simplifica el desarrollo del software ya que facilita los procesos repetitivos a lo largo de él, aumentando así la productividad con la que podemos trabajar (Ponce Briones, 2016). Al tener ya decidido el lenguaje de programación (Python), fue sencilla la elección de un IDE adecuado, ya que centré mi elección en aquellos dedicados exclusivamente en el desarrollo de códigos en Python. Otro de los requisitos necesarios era la compatibilidad de este con las librerías que íbamos a necesitar a lo largo del código: *random* (para la generación de números aleatorios), *math* (para el cálculo del cociente entero superior) y *xlwt* (para la creación de un archivo Excel donde recogeremos todos los resultados del estudio). Finalmente, decidí usar PyCharm.

4.4. Solver: Gurobi

La última decisión necesaria en cuanto al entorno de trabajo era el solver que usaría para resolver el modelo codificado: Gurobi Optimizer 9.5.1. En este podemos introducir nuestras variables, restricciones y función objetivo que compone el problema y nos devolverá el resultado óptimo (en caso de que exista). Algunas de las comandas necesarias para la definición de estos parámetros son:

- *modelo.Model()*: es una de las líneas fundamentales para la solución de un modelo ya que es la encargada de su creación.
- *modelo.addVar(dimensión, vtype= tipo de variable, name = nombre de la variable)*: una vez creado el modelo, definimos las variables de este, indicando su dimensión, el tipo de variable y su nombre.
- *modelo.setObjective(expresión matemática de la función objetivo, GRB:MAXIMIZE /GRB.MINIMIZE)*: añadimos la expresión matemática que usaremos como función objetivo e indicamos si queremos maximizarla o minimizarla.
- *modelo.addConstrts(restricción, nombre de la restricción)*: al igual que para la función objetivo, escribimos la expresión matemática que usaremos como restricción del problema y el nombre con el cual nos referiremos a ella.
- *modelo.optimize()*: por último, para la resolución del modelo definido, tendremos que llamar a esta función, gracias a la cuál obtendremos los resultados de nuestro modelo.

5. GENERACIÓN DE INSTANCIAS

Con el objetivo de poder hacer una comparación objetiva de las diferentes alternativas que vamos a estudiar, usaremos los mismos datos de entrada para resolver todos los modelos. A cada uno de estos escenarios los denominaremos instancias. Para la elaboración de estos datos, usaremos los considerados en el artículo “New heuristics for planning operating rooms” (Molina-Pariente, Hans, Framinan, & Gómez-Cía, 2015), en el Trabajo de Fin de Grado “Integración de planificación de intervenciones en quirófanos y camas postoperatorias” (Nieto Medina, 2021) y algunas adaptaciones específicas para nuestro caso de estudio. A continuación, veremos una breve explicación de cada uno de los parámetros de entrada del problema y la forma en que serán definidos.

5.1. Definición de los conjuntos

- Operaciones (I): recoge el número total de intervenciones que están en la lista de espera en el horizonte temporal estudiado. Calcularemos la dimensión del conjunto de la siguiente forma: añadiremos de forma unitaria cada operación hasta que la suma de la duración aproximada de operación supere en un $\beta\%$ el tiempo total de disponibilidad de los quirófanos. Nosotros consideraremos dos valores para β : 100 y 125 (Nieto Medina, 2021).
- Quirófanos (J): estudiaremos instancias con tres números distintos de quirófanos disponibles: 2, 3 y 4. A la hora de la planificación de las intervenciones hay que tener en cuenta los quirófanos que están especializados, para su uso prioritario para operaciones que los requieran. El número de quirófanos especializados es del 30% (Nieto Medina, 2021).
- Días (H): estudiaremos instancias con cuatro horizontes temporales diferentes: de 2, 3, 4 y 5 días.
- Cirujanos (S): el tamaño de este conjunto lo calcularemos a partir de la suma de las capacidades temporales de los quirófanos (r_{jh}), el número de semanas estudiadas en el horizonte temporal (W), el tiempo máximo que un cirujano puede trabajar ($a_{sh_{m\acute{a}x}}$) y el número máximo de días que un cirujano puede a lo largo de una semana (mds). Este último valor va a depender de la alternativa en la que nos situemos. Sin embargo, para poder realizar un análisis objetivo de los resultados de todos los modelos debemos usar un único valor de S . Usaremos por tanto el mínimo valor posible mds valor posible (1), obteniendo así el máximo valor de S . Además, se añade un factor de control a la fórmula (α), el cual tomará los valores de 1.5 y 2 (Molina-Pariente, Hans, Framinan, & Gómez-Cía, 2015).

$$S = \alpha * \frac{\sum_{j \in J} \sum_{h \in H} r_{jh}}{W * a_{sh_{m\acute{a}x}} * mds} \quad (78)$$

5.2. Definición de los parámetros

- Capacidad temporal de los quirófanos (r_{jh}): consideramos que todos los quirófanos tendrán una capacidad de 8 h todos los días (Nieto Medina, 2021).
- Capacidad temporal de los cirujanos (a_{sh}): asignamos una capacidad igual a la de los quirófanos, es decir, de 8 h.
- Cirujano encargado (γ_i): será igual al índice asignado al cirujano que intervendrá al paciente i . Será calculado con una distribución uniforme de números enteros $U[0, 1]$ (Nieto Medina, 2021).

- Primer día de posible intervención (rd_i): cada operación tendrá asignado un día del plano horizontal, el cual será el primero en el que se podrá planificar la intervención. Será calculado por una distribución uniforme $U[1, J]$.
- Último día de posible intervención (d_i): en este caso, será el último día del plano horizontal donde se podrá planificar una operación. Esta fecha será calculada como la diferencia entre el máximo tiempo que un paciente puede esperar antes de ser intervenido ($MTBT$, *Maximum Time Before Treatment*) y los días que lleva el paciente en la lista de espera (dwl).

$$d_i = MTBT_i - dwl_i \quad (79)$$

$MTBT_i$ tomará aleatoriamente uno de los siguientes tres valores: 45, 180 y 360, los cuales son los valores de dicho parámetro según el Servicio Nacional de Salud, dependiendo de la especialidad de la intervención (Molina-Pariente, Hans, Framinan, & Gómez-Cía, 2015). Los días que lleva el paciente en la lista de espera serán calculados con una distribución uniforme de $U[0, MTBT_i - 1]$.

- Factibilidad de la operación (δ_{ijh}): es un parámetro binario. Tomará un valor igual a 1 cuando la operación i se puede realizar en el quirófano j y el día h , y 0 en el caso contrario. Este parámetro depende de si la operación estudiada requiere de un quirófano especializado o no. En nuestro estudio, consideraremos que únicamente el 10% de las operaciones que se encuentran en la lista de espera requieren de este tipo de quirófanos (Molina-Pariente, Hans, Framinan, & Gómez-Cía, 2015) y, por tanto, estas operaciones solo tendrán valores $\delta_{ijh} = 1$ en aquellos quirófanos que sean especializados. Aquellas operaciones que no necesiten estos recursos no tendrán valores nulos para este parámetro.
- Duración de la operación (t_i): se le asigna a cada intervención una duración aproximada. Será calculado siguiendo una distribución $U[90, 120]$ min.
- Peso clínico (w_i): será calculado como una combinación lineal de múltiples factores (Molina-Pariente, Hans, Framinan, & Gómez-Cía, 2015):

$$w_i = a * \frac{mp_i}{mp_{m\acute{a}x}} + a * \frac{dwl_i}{MTBT_i} \quad (80)$$

donde mp_i representa la prioridad médica de la intervención, la cual puede tomar valores enteros entre 1 y 5. Asumiremos que el peso de ambos factores es el mismo ($a = 0,5$).

5.3. Resumen las instancias

Una vez conocido lo que representa cada uno de los parámetros y los valores que van a tomar, mostraré un cuadro resumen de ellos.

Conjuntos

I	J	H	S
Añadimos de 1 en 1 hasta superar un $\beta\%$ (100, 125) el $\sum_{j \in J} \sum_{h \in H} r_{jh}$	2, 3, 4	2, 3, 4, 5	<i>Ec. (78)</i>

Tabla 15: Tamaño de los conjuntos en las instancias

Parámetros

r_{jh}	a_{sh}	γ_i	rd_i	d_i	δ_{ijh}	t_i	w_i
480	480	U[0, 1]	U[1, J]	Ec. (79)	Según requiera quirófano especializado	U[90, 120]	Ec. (80)

Tabla 16: Valores de los parámetros en las instancias

Otros parámetros

β	α	W	mds	$MTBT_i$	dwl_i
100%, 125%	1.5, 2	[H/7]	1	{45, 180, 360}	U[0, $MTBT_i - 1$]

Tabla 17: Valores de otros parámetros en las instancias

Todos los parámetros dependen de qué valores toman J , α , β y H . Por ello, si conocemos todas las combinaciones diferentes entre estos valores sabremos las diferentes estancias que vamos a estudiar.

J	α	β	H
2, 3, 4	1.5, 2	100, 125	2, 3, 4, 5

Tabla 18: Datos de entrada para generar instancias

Nos saldría un total de 48 instancias diferentes. Cada una de ellas la ejecutaremos 2 veces, obteniendo así un total de 96 resoluciones de cada modelo.

6. RESULTADOS

Una vez diseñados los modelos a estudiar y la batería de datos que vamos a usar para resolverlos, pasamos a la ejecución del código, la recogida de resultados y su análisis. Al ser cada una de las instancias de un tamaño diferente y variar el número de restricciones entre los modelos, resulta imprescindible definir un tiempo límite de ejecución de cada modelo para poder hacer un análisis objetivo. Este tiempo es de 600 seg (10 min). De esta manera, si se alcanza ese tiempo y aún no se ha encontrado una solución óptima al problema, el programa devuelve el mejor valor de la función objetivo hasta ahora conseguido.

6.1. Resultados obtenidos

En esta sección veremos los resultados obtenidos tras resolver las 48 instancias posibles 2 veces cada una. Para su análisis posterior, hemos recogido tres tipos de datos por cada instancia: si se ha alcanzado la solución óptima, el valor de esta o, en su defecto, el mejor valor hasta ahora hallado y el tiempo de ejecución.

En primer lugar, nos centraremos en el número de óptimos encontrados por cada modelo. En la Tabla 20 y la Tabla 21 podemos ver los resultados obtenidos para este parámetro. Aquellas instancias cuya casilla tenga un 1 significa que hemos obtenido una solución óptima para el problema. Por el contrario, si encontramos un 0, significa que se ha alcanzado el tiempo límite de ejecución. Como resumen de estos datos, he elaborado la siguiente tabla, donde podemos ver el número total de óptimos encontrados por modelo e iteración

MODELO	B	1	2	3	4	5
ÓPTIMO (I1)	45	30	44	43	48	47
LÍMITE ALCANZADO (I1)	3	18	4	5	0	1
ÓPTIMO (I2)	46	32	46	46	48	47
LÍMITE ALCANZADO (I2)	2	16	2	2	0	1
ÓPTIMO	91	62	90	89	96	94
LÍMITE ALCANZADO	5	34	6	7	0	2

Tabla 19: Tabla resumen del número de óptimos

Podemos observar que hemos conseguido datos similares en cada instancia para cada modelo, por lo que podemos decir que se han comportado de forma similar ante los cambios producidos en los datos de entrada.

Vemos que todos los modelos tienen un alto ratio de óptimos hallados, aunque solamente el Modelo 4 ha conseguido la solución óptima en cada una de las instancias estudiadas. El Modelo 2 es el único que presenta un bajo ratio de óptimos comparado con el resto, habiendo encontrado únicamente 62 soluciones óptimas de las 96 estudiadas. Estas variaciones entre los datos, además de deberse a la diferencia de tamaños entre los modelos (tienen un número diferente de restricciones, algunos cuentan con restricciones más estrictas que otros, ...), también podrían ser causadas debido a que la resolución de todos los modelos no tiene el mismo rendimiento ante el mismo método de resolución.

A continuación, adjunto, además de las tablas con los óptimos alcanzados, otras 4 tablas donde podemos ver todos los valores de la función objetivo (Tabla 22 y Tabla 23) y los tiempos de ejecución (Tabla 24 y Tabla 25). En todas las tablas hemos usado el mismo criterio: aquellas instancias donde esté el valor en rojo son aquellas en las que se ha alcanzado el tiempo límite de ejecución y, por tanto, no son valores óptimos.

J	α	β	H	MB - I1	M1 - I1	M2 - I1	M3 - I1	M4 - I1	M5 - I1	
2	1,5	1	2	1	1	1	1	1	1	
			3	1	1	1	1	1	1	
			4	1	1	1	1	1	1	
			5	1	0	1	1	1	1	
		1,25	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	0	1	1	1	1	1
			5	1	0	1	1	1	1	1
	2	1	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	1	1	1	1	1	1
			5	1	1	1	1	1	1	1
		1,25	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	1	1	1	1	1	1
			5	1	0	1	1	1	1	1
3	1,5	1	2	1	1	1	1	1	1	
			3	1	1	1	1	1	1	
			4	1	0	1	1	1	1	1
			5	1	0	0	0	1	1	1
		1,25	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	0	1	1	1	1	1
			5	1	0	1	1	1	1	1
	2	1	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	1	1	1	1	1	1
			5	1	0	0	1	1	1	1
		1,25	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	0	1	1	1	1	1
			5	1	0	1	1	1	1	1
4	1,5	1	2	1	1	1	1	1	1	
			3	1	1	1	1	1	1	
			4	1	0	1	1	1	1	1
			5	0	0	0	0	1	1	1
		1,25	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	0	1	1	1	1	1
			5	0	0	0	0	1	0	1
	2	1	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	1	1	1	1	1	1
			5	0	0	1	0	1	1	1
		1,25	2	1	1	1	1	1	1	1
			3	1	0	1	1	1	1	1
			4	1	1	1	1	1	1	1
			5	1	0	1	0	1	1	1

Tabla 20: Óptimos encontrados (iteración 1)

J	α	β	H	MB - I1	M1 - I1	M2 - I1	M3 - I1	M4 - I1	M5 - I1	
2	1,5	1	2	1	1	1	1	1	1	
			3	1	1	1	1	1	1	
			4	1	1	1	1	1	1	
			5	1	0	1	1	1	1	
		1,25	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	1	1	1	1	1	1
			5	1	0	1	1	1	1	1
	2	1	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	1	1	1	1	1	1
			5	1	1	1	1	1	1	1
		1,25	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	1	1	1	1	1	1
			5	1	0	1	1	1	1	1
3	1,5	1	2	1	1	1	1	1	1	
			3	1	1	1	1	1	1	
			4	1	1	1	1	1	1	
			5	1	0	1	1	1	1	
		1,25	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	0	1	1	1	1	1
			5	0	0	0	0	1	1	
	2	1	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	1	1	1	1	1	1
			5	1	0	1	1	1	1	1
		1,25	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	0	1	1	1	1	1
			5	1	0	1	1	1	1	1
4	1,5	1	2	1	1	1	1	1	1	
			3	1	1	1	1	1	1	
			4	1	0	1	1	1	1	
			5	1	0	1	1	1	1	
		1,25	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	0	1	1	1	1	1
			5	1	0	1	1	1	1	1
	2	1	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	1	1	1	1	1	1
			5	1	0	1	1	1	1	1
		1,25	2	1	1	1	1	1	1	1
			3	1	1	1	1	1	1	1
			4	1	0	1	1	1	1	1
			5	0	0	0	0	1	0	

Tabla 21: Óptimos encontrados (iteración 2)

J	α	β	H	MB - I1	M1 - I1	M2 - I1	M3 - I1	M4 - I1	M5 - I1
2	1,5	1	2	2,0736	2,0736	2,0736	2,0736	1,3861	2,0736
			3	5,2917	5,2917	5,2917	5,2917	3,9882	5,2368
			4	6,4347	6,4347	6,4278	6,4282	5,0773	6,0289
			5	7,1746	7,1745	7,1584	7,1746	5,1648	6,6532
		1,25	2	2,3694	2,3694	2,2028	2,3694	1,4250	2,3694
			3	5,0264	5,0264	5,0130	5,0264	3,4991	4,7998
			4	6,5844	6,5844	6,5513	6,5513	5,2787	6,4507
			5	8,2398	8,2398	8,2324	8,2398	6,2073	8,0222
	2	1	2	2,1222	2,1222	2,1222	2,1222	1,2743	2,1222
			3	4,4981	4,4981	4,4981	4,4981	3,6264	4,3972
			4	6,2196	6,2196	6,2196	6,2196	3,6321	5,2565
			5	7,5822	7,5822	7,5503	7,5604	4,9993	7,1983
		1,25	2	3,0993	3,0993	3,0118	3,0993	2,1250	3,0993
			3	5,1627	5,1627	5,1627	5,1627	3,0660	5,0523
			4	6,2640	6,2640	6,2640	6,2640	4,0536	6,1163
			5	8,4465	8,4465	8,4395	8,4465	5,9485	8,1433
3	1,5	1	2	3,6778	3,6778	3,6778	3,6778	2,0785	3,6778
			3	6,3428	6,3428	6,2396	6,3354	4,3630	6,0917
			4	9,3986	9,3986	9,3868	9,3986	5,3042	7,7846
			5	10,5297	10,5026	10,4772	10,5269	6,8459	9,5188
		1,25	2	3,7535	3,7535	3,7535	3,7535	2,5049	3,7535
			3	5,7509	5,7509	5,6218	5,6847	3,7211	5,2468
			4	9,5057	9,5057	9,3142	9,4376	6,3035	8,3308
			5	11,6469	11,5063	11,5815	11,6330	8,2875	10,7719
	2	1	2	2,1903	2,1903	2,1903	2,1903	1,7465	2,1903
			3	6,5595	6,5595	6,4252	6,4965	4,3289	6,3949
			4	8,5019	8,5019	8,4523	8,4830	5,1251	7,4605
			5	10,6278	10,5653	10,5391	10,6278	6,1240	9,7932
		1,25	2	3,0069	3,0069	3,0069	3,0069	1,8292	3,0069
			3	6,9921	6,9921	6,9310	6,9525	4,1181	6,7144
			4	9,8426	9,8426	9,7245	9,7500	5,8503	8,9354
			5	11,9702	11,9031	11,9338	11,9706	8,2597	11,4684
4	1,5	1	2	2,1306	2,1306	2,1306	2,1306	1,7576	2,1306
			3	5,9079	5,9931	5,9079	5,9079	3,9741	5,4051
			4	12,0823	12,0823	11,8941	12,0660	7,2513	9,7860
			5	14,5743	14,4851	14,4464	14,5525	9,6541	12,6588
		1,25	2	2,3347	2,3347	2,3347	2,3347	1,6792	2,3347
			3	6,4072	6,4072	6,4072	6,4072	4,5611	5,9771
			4	10,9343	10,9218	10,6816	10,8288	6,8816	9,2907
			5	15,7793	15,6620	15,5543	15,7179	11,5803	14,5095
	2	1	2	2,7722	2,7722	2,7722	2,7722	1,7674	2,7722
			3	6,4370	6,4370	6,4370	6,4370	3,6894	5,7877
			4	11,1852	11,1852	11,0815	11,1852	5,6993	9,3561
			5	13,4387	13,3929	13,2921	13,4387	7,7887	11,2537
		1,25	2	4,3035	4,3035	4,3035	4,3035	2,5674	4,3035
			3	8,4840	8,4840	8,3891	8,4400	5,1551	7,6324
			4	11,5537	11,5537	11,4985	11,5537	6,5044	10,2569
			5	14,5414	14,2607	14,3538	14,4884	9,0706	13,7458

Tabla 22: Valor FO (iteración 1)

J	α	β	H	MB - I2	M1 - I2	M2 - I2	M3 - I2	M4 - I2	M5 - I2
2	1,5	1	2	1,5153	1,5153	1,5153	1,5153	1,3118	1,5153
			3	4,1648	4,1648	4,1546	4,1648	2,3917	3,7924
			4	6,3997	6,3997	6,3950	6,3997	4,5167	5,8632
			5	7,3978	7,3978	7,3654	7,3654	5,3244	6,9917
		1,25	2	2,8167	2,8167	2,8167	2,8167	1,6917	2,8167
			3	4,6150	4,6150	4,6025	4,6025	3,5986	4,5727
			4	6,6881	6,6881	6,6569	6,6844	5,1654	6,2367
			5	7,8557	7,8534	7,7810	7,8501	6,2626	7,4688
	2	1	2	2,7299	2,7299	2,7299	2,7299	1,4389	2,7299
			3	5,1005	5,1005	5,0549	5,0549	3,5787	5,0322
			4	6,3212	6,3212	6,3042	6,3212	4,4312	5,7464
			5	7,5912	7,5912	7,5819	7,5834	5,0265	7,1690
		1,25	2	2,1688	2,1688	2,1688	2,1688	1,4215	2,1688
			3	5,0421	5,0421	5,0271	5,0271	3,0840	4,9303
			4	6,3466	6,3466	6,3466	6,3466	4,0331	5,8234
			5	8,8715	8,8715	8,8196	8,8701	6,5361	8,5427
3	1,5	1	2	1,5806	1,5806	1,5806	1,5806	1,2056	1,5806
			3	4,2602	4,2602	4,2602	4,2602	2,6005	3,7991
			4	8,9934	8,9934	8,8420	8,9934	5,9613	7,8815
			5	11,0414	10,8924	11,0300	11,0414	7,0390	10,1123
		1,25	2	2,3375	2,3375	2,3375	2,3375	1,6889	2,3375
			3	6,1424	6,1424	6,1206	6,1287	4,2796	5,8956
			4	9,4440	9,4440	9,4000	9,4440	6,2465	7,9981
			5	11,8497	11,7630	11,8036	11,8399	8,6689	10,8060
	2	1	2	1,7542	1,7542	1,7542	1,7542	1,5778	1,7542
			3	5,0273	5,0273	5,0106	5,0106	2,4211	4,7241
			4	7,7532	7,7539	7,6287	7,7539	4,4939	7,2914
			5	11,2570	11,1769	11,2404	11,2541	7,1638	9,9977
		1,25	2	3,4368	3,4368	3,1743	3,4368	1,4167	3,4368
			3	6,5315	6,5315	6,4745	6,4796	4,5977	6,2208
			4	9,5700	9,4656	9,4770	9,5610	5,8242	8,9791
			5	11,3924	11,3344	11,3792	11,3813	7,5815	10,5736
4	1,5	1	2	2,0618	2,0618	2,0618	2,0618	1,5181	2,0618
			3	5,4704	5,4704	5,4704	5,4704	4,2553	5,0454
			4	11,5630	11,5630	11,4101	11,4993	7,5410	9,6900
			5	13,0546	12,8540	12,8493	12,8911	8,3784	10,7638
		1,25	2	3,4889	3,4889	3,4889	3,4889	2,5569	3,4889
			3	7,7523	7,7523	7,6662	7,7523	4,5794	7,3799
			4	11,9940	11,9799	11,8968	11,9900	7,4610	10,5878
			5	14,9672	14,9502	14,9137	14,9672	9,1394	13,4209
	2	1	2	2,3222	2,3222	2,3222	2,3222	1,7194	2,3222
			3	7,3002	7,3002	7,3002	7,3002	4,1856	6,7370
			4	9,8755	9,8755	9,8292	9,8569	5,5068	8,8084
			5	13,9015	13,7806	13,8424	13,8754	7,9168	12,1761
		1,25	2	3,6111	3,6111	3,3778	3,6111	2,3257	3,6111
			3	5,5718	5,5718	5,5718	5,5718	3,3678	4,7970
			4	11,0135	11,0135	10,8611	11,0135	6,3343	9,5752
			5	15,4660	15,4197	15,4070	15,4660	10,4775	14,7094

Tabla 23: Valor FO (iteración 2)

J	α	β	H	MB - I1	M1 - I1	M2 - I1	M3 - I1	M4 - I1	M5 - I1
2	1,5	1	2	0,0090	0,0107	0,0000	0,0000	0,0217	0,0000
			3	0,0480	1,3561	0,0226	0,0225	0,0272	0,0600
			4	1,4846	70,6628	1,3531	1,3853	0,0476	2,0924
			5	232,8744	600,0584	348,9959	582,4469	0,0890	17,4344
		1,25	2	0,0156	0,0655	0,0324	0,0000	0,0000	0,0053
			3	0,7754	11,5940	0,5023	0,9863	0,0350	0,8491
			4	500,8428	600,0031	56,8075	113,9980	0,1900	4,7442
			5	24,1524	600,0449	19,7363	39,0957	0,1581	14,8042
	2	1	2	0,0000	0,0480	0,0000	0,0000	0,0000	0,0000
			3	0,0000	0,4060	0,0156	0,0107	0,0000	0,0334
			4	0,2767	12,5834	0,1252	0,1964	0,0480	4,5895
			5	2,3872	257,2716	3,5909	1,8649	0,0627	7,2303
		1,25	2	0,0156	0,1305	0,0642	0,0000	0,0000	0,0497
			3	0,3646	14,0531	0,3024	0,5336	0,0076	0,4701
			4	0,4816	70,6293	0,7185	0,8766	0,0897	0,5442
			5	5,1089	600,0244	7,3233	6,9012	0,1882	5,1438
3	1,5	1	2	0,0000	0,0492	0,0168	0,0000	0,0000	0,0000
			3	0,1200	7,4039	0,0777	0,1448	0,0156	0,0881
			4	14,0255	600,0271	4,7168	1,5076	0,1432	1,3419
			5	216,5876	600,0180	600,0034	600,0036	0,3147	9,0047
		1,25	2	0,0000	0,0579	0,0000	0,0011	0,0000	0,0156
			3	0,0747	5,9292	1,3565	1,2244	0,0156	0,1716
			4	4,7680	600,0352	8,1280	6,2101	0,1382	8,7044
			5	67,9437	600,0220	91,4901	66,5625	0,9320	42,8677
	2	1	2	0,0033	0,0990	0,0052	0,0031	0,0061	0,0047
			3	0,0857	5,2419	0,6491	0,2578	0,0244	0,1488
			4	0,7795	319,4323	4,1723	1,4700	0,0876	6,0499
			5	447,4788	600,0104	600,0068	201,1017	0,1742	22,0444
		1,25	2	0,0000	0,1269	0,0055	0,0095	0,0000	0,0053
			3	1,5935	63,1075	0,9221	1,0805	0,0069	0,1943
			4	4,4206	600,0594	6,6886	5,7415	0,1693	47,5006
			5	60,6535	600,0082	151,5272	67,7553	0,8687	182,0337
4	1,5	1	2	0,0000	0,0550	0,0000	0,0000	0,0000	0,0000
			3	0,0000	0,4656	0,0000	0,0156	0,0050	0,0382
			4	1,4256	600,1198	5,0208	3,3363	0,1933	2,8154
			5	600,0125	600,0143	600,0084	600,0218	0,4738	28,3414
		1,25	2	0,0000	0,1294	0,0000	0,0000	0,0000	0,0000
			3	0,0000	0,4466	0,0000	0,0159	0,0265	0,0186
			4	5,3703	600,0654	34,6711	5,9131	0,1309	5,7418
			5	600,0273	600,1964	600,0051	600,0264	0,6652	600,0014
	2	1	2	0,0000	0,0875	0,0000	0,0168	0,0000	0,0000
			3	0,0784	79,7591	0,1186	0,1549	0,0180	0,3131
			4	1,4342	225,6597	0,2289	1,0789	0,0672	5,5043
			5	600,0117	600,3060	74,5639	600,0089	0,4639	9,1330
		1,25	2	0,0000	0,1357	0,0000	0,0000	0,0093	0,0168
			3	11,4196	600,0203	3,6872	3,2059	0,0494	20,5660
			4	0,1656	144,1465	0,2582	0,0977	0,0905	2,3058
			5	99,0182	600,0299	246,6826	600,0112	0,4018	296,6765

Tabla 24: Tiempos de ejecución (iteración 1)

J	α	β	H	MB - I1	M1 - I1	M2 - I1	M3 - I1	M4 - I1	M5 - I1		
2	1,5	1	2	0,0026	0,0101	0,0000	0,0000	0,0032	0,0087		
			3	0,1582	3,6465	0,1842	0,1540	0,0169	0,1165		
			4	0,4702	21,0614	0,3153	0,3053	0,1773	2,7564		
			5	12,9921	600,0540	7,5082	18,9895	0,1019	4,9448		
			2	0,0156	0,0528	0,0000	0,0000	0,0055	0,0000		
		3	0,0981	3,6626	0,1098	0,1454	0,0325	0,1294			
		4	3,6268	82,2103	6,4841	2,1926	0,0446	3,2890			
		5	52,5529	600,0755	15,2668	49,2911	0,2800	12,0841			
		1,25	2	0,0000	0,0601	0,0000	0,0000	0,0000	0,0000		
			3	0,1603	4,7207	0,1300	0,1639	0,0312	0,1480		
	4		1,0659	56,1528	1,1802	0,9147	0,0645	0,4809			
	5		9,1772	388,5138	4,8438	5,8554	0,0966	6,7192			
	2		0,0000	0,0611	0,0000	0,0000	0,0156	0,0000			
	2	1	3	0,0969	2,1593	0,0732	0,1136	0,0120	0,0596		
			4	1,0881	35,1166	0,7189	0,8418	0,0502	2,0898		
			5	23,5527	600,0326	40,2938	17,2222	0,1206	27,2670		
			1,25	2	0,0000	0,0317	0,0000	0,0000	0,0000	0,0000	
				3	0,0257	2,2307	0,0156	0,0325	0,0157	0,1004	
		4		1,2397	143,6099	2,3035	1,4479	0,0657	2,6097		
		5		50,5772	600,0083	17,8431	33,5178	0,2924	36,7597		
2		0,0021		0,0620	0,0020	0,0020	0,0156	0,0037			
3		1,5	3	4,9530	346,3992	4,2346	34,9074	0,0845	0,3764		
			4	11,6189	600,0536	8,5785	14,2736	0,0883	4,8135		
	5		600,0103	600,0131	600,0189	600,0203	0,8507	360,5579			
	1		2	0,0000	0,0489	0,0038	0,0000	0,0092	0,0065		
			3	0,0251	3,5403	0,0412	0,0490	0,0231	0,1377		
		4	0,5802	159,3179	1,0323	0,5769	0,0869	0,3502			
		5	8,5769	600,0173	15,1323	20,1365	0,1903	32,1631			
		2	0,0090	0,0719	0,0075	0,0056	0,0047	0,0070			
	4	1,5	1	3	0,1426	18,8617	0,1793	0,1978	0,0152	1,0317	
				4	19,8118	600,0521	26,0507	20,4573	0,1837	8,0097	
5				107,9105	600,0886	142,9370	77,3898	0,6050	298,3754		
1,25				2	0,0000	0,0604	0,0000	0,0000	0,0000	0,0000	
				3	0,0169	0,4601	0,0159	0,0161	0,0249	0,0252	
		4	33,7158	600,0578	35,6390	52,8553	0,2192	2,3296			
		5	84,1284	600,0155	594,1912	499,3406	0,6338	22,7698			
		2	0,0210	0,0516	0,0000	0,0158	0,0000	0,0000			
4		1,5	1,25	3	0,0304	2,7166	0,3628	0,0312	0,0410	0,0849	
				4	78,5581	600,0794	13,2550	31,7952	0,1189	27,4610	
	5			281,8961	600,0304	227,5601	323,1724	0,8410	26,2640		
	2			1	2	0,0000	0,0731	0,0000	0,0000	0,0000	0,0000
					3	0,0156	0,7679	0,0168	0,0202	0,0399	0,0767
		4	0,2104		257,8036	0,3923	0,6317	0,1038	0,2670		
		5	52,9321		600,0264	47,0246	76,8620	0,2221	5,9110		
		2	0,0000		0,0923	0,0000	0,0000	0,0000	0,0000		
	2	1,25	3	0,2118	33,1158	0,1422	0,4959	0,0169	1,1940		
			4	3,3393	600,0262	5,4538	2,7752	0,1342	2,6962		
5			600,0130	600,0286	600,0255	600,0090	0,8229	600,0155			

Tabla 25: Tiempos de ejecución (iteración 2)

Como vemos en la Tabla 20 y Tabla 21, hemos obtenido una solución, sea óptima o no, en todas las instancias estudiadas, es decir, todos los modelos son factibles para los datos de entrada utilizados.

En los valores de la FO podemos observar una tendencia ascendente según aumentamos el número de días estudiados en el plano horizontal. Esto resulta coherente ya que cuantos más días tengamos a nuestra disposición, más intervenciones se pueden realizar. Esta tendencia la podemos observar también en los tiempos de ejecución ya que, cuantos más días estemos considerando, más grande es nuestro problema y más se tarda en encontrar la solución.

A modo de resumen de las tablas de los tiempos de ejecución, elaboramos dos tablas resumen: una donde recogemos el tiempo medio para cada modelo y otra donde comparamos los tiempos de ejecución de cada una de las alternativas con el Modelo Base.

MODELO	BASE	1	2	3	4	5
Tiempo medio de ejecución	57,8330	242,2876	61,3976	68,7762	0,1381	29,6270

Tabla 26: Tiempos medio de ejecución

Si ordenamos los modelos según los tiempos promedios obtenidos y hacemos lo mismo según el número de veces que ha alcanzado cada uno de ellos el tiempo límite, vemos que obtenemos exactamente el mismo orden. Esto es consecuencia de que cuantas más veces se ha alcanzado el tiempo límite, mayor es la suma de los tiempos de ejecución para cada modelo y, por tanto, mayor su valor promedio.

Observamos que hay una gran diferencia entre el tiempo promedio del Modelo 1 y el siguiente en el ranking (Modelo 3). El Modelo 1 supera casi en un 400% el tiempo del Modelo 3, lo cual queda justificado con que sea el modelo que más veces ha alcanzado el tiempo límite de ejecución. En el extremo contrario, encontramos al Modelo 4, donde el tiempo promedio no supera ni 0,5 segundos. Los Modelos Base, 2 y 3 muestran un tiempo medio parecido (con 10,9432 s de diferencia entre los tiempos máximos y mínimos).

Al ser el objetivo principal de este estudio comparar el rendimiento de cada una de las alternativas diseñadas con el Modelo Base, es interesante conocer una comparativa de los tiempos de ejecución entre cada uno de ellos con el modelo de referencia en aquellas instancias donde ambos modelos han alcanzado el óptimo.

MODELO	1	2	3	4	5
Mayor tiempo que MB	62	41	42	12	47
Menor tiempo que MB	0	48	47	79	44

Tabla 27: Comparativa entre tiempos de ejecución

Vemos que el Modelo 1 es el único que presenta un tiempo mayor que el Modelo Base en cada una de las instancias óptimas estudiadas, mientras que el Modelo 4 es donde encontramos el menor número de instancias que requieren mayor tiempo que el Modelo Base. Los otros 3 modelos presentan unos números equilibrados con el Modelo Base.

Podemos observar que todos los valores temporales de color rojo de las Tabla 24 y Tabla 25 superan los 600 s de tiempo de parada. A pesar de establecer este valor como tiempo límite, la optimización no se detiene instantáneamente ya que requiere de más tiempo hasta completar los cálculos necesarios para su finalización (TimeLimit, s.f.).

6.2. Tratamiento de los resultados

Para un análisis más profundo de los valores obtenidos para la función objetivo vamos a usar 2 parámetros: RPD_1 y RPD_2 . El primero lo usaremos para comparar todos los valores obtenidos, sean óptimos o no, con el mejor resultado, es decir, con el máximo valor hallado en esa instancia e iteración. El segundo lo usaremos para estudiar el comportamiento de cada uno de los modelos alternativos respecto al Modelo Base, comprobando así si mejoran el rendimiento de este. En este caso, solo compararemos aquellos valores que sean óptimos.

A continuación, podemos ver las fórmulas de cada uno de los parámetros:

$$RPD_1 = \frac{FO_{Máx} - FO_{Modelo}}{FO_{Máx}} \quad (81)$$

$$RPD_2 = \frac{FO_{MB} - FO_{Modelo}}{FO_{MB}} \quad (82)$$

Estos parámetros los calculamos para cada una de las instancias, iteraciones y modelos estudiados. Al estar calculado RPD_2 siempre respecto al Modelo Base, omitiremos el cálculo de este parámetro para dicho modelo.

En primer lugar, estudiaremos los datos obtenidos para RPD_1 (Tabla 30 y Tabla 31). Vemos a primera vista que hay una gran diferencia entre los valores obtenidos para el Modelo 4 con el resto de modelos. En todas las instancias (para ambas iteraciones) se ha obtenido en este modelo el valor máximo de RPD_1 . Analizando la fórmula de este parámetro, podemos afirmar que esto se debe a que hemos encontrado en dicho modelo el valor mínimo en cada instancia.

Para comparar el comportamiento de cada modelo, calculamos el valor promedio para cada iteración y el total.

MODELO	B	1	2	3	4	5
ARPD _{1,11}	0,03%	0,13%	0,77%	0,20%	34,22%	6,15%
ARPD _{1,12}	0,00%	0,15%	0,70%	0,13%	33,70%	6,20%
ARPD ₁	0,01%	0,14%	0,74%	0,16%	33,96%	6,18%

Tabla 28: Valores promedio para RPD[1]

Mientras que, en los Modelos Base, 1, 2 y 3 hemos encontrado unos valores cercanos al 0,00% (lo que significa que sus resultados son los máximos hallados o están muy cercanos a estos), el Modelo 4 presenta un valor de ARPD₁ igual al 33,96%, lo que demuestra que son los valores con mayor distancia al mejor óptimo.

Conociendo los valores de este parámetro, podemos deducir cómo se va a comportar el parámetro RPD_2 en cada uno de los modelos:

- Los valores para los Modelos 1, 2 y 3 deben ser cercanos a 0,00%, ya que los tres modelos se han comportado de manera similar al Modelo Base.
- En el caso del Modelo 4, debemos encontrar un valor positivo y considerablemente alejado del 0,00%, debido a que hemos hallado valores menores que en el Modelo Base.
- El valor obtenido en el Modelo 5 debe ser positivo y ligeramente alejado del 0,00%, ya que vemos que en el Modelo Base hemos obtenido valores mayores que en el Modelo 5.

En la Tabla 32 y Tabla 33 recogemos el valor de RPD_2 para cada instancia. En los casos en los que el Modelo Base o el modelo con el que se está comparando no ha alcanzado una solución óptima, encontraremos “-” en la celda, ya que solo estamos comparando los valores óptimos.

A continuación, vemos una tabla resumen con los valores promedios de RPD_2 en cada iteración y la media de estas.

MODELO	1	2	3	4	5
ARPD _{2,11}	-0,05%	0,73%	0,17%	34,20%	6,08%
ARPD _{2,12}	0,00%	0,71%	0,13%	33,70%	6,23%
ARPD ₂	-0,02%	0,72%	0,15%	33,95%	6,16%

Tabla 29: Valores promedio para RDP[2]

Observamos que las hipótesis anteriores se han cumplido en todos los casos. En la iteración 2 obtenemos un

valor igual a a 0,00% para el Modelo 1, lo que significa que en dicha iteración hemos obtenido valores idénticos a los del Modelo Base. Además, este es el único que presenta un $ARPD_2$ negativo, lo que nos indica que es el único modelo que mejora los resultados del Modelo Base.

J	α	β	H	MB - RPD1	M1 - RPD1	M2 - RPD1	M3 - RPD1	M4 - RPD1	M5 - RPD1
2	1,5	1	2	0,00%	0,00%	0,00%	0,00%	33,15%	0,00%
			3	0,00%	0,00%	0,00%	0,00%	24,63%	1,04%
			4	0,00%	0,00%	0,11%	0,10%	21,10%	6,31%
			5	0,00%	0,00%	0,23%	0,00%	28,01%	7,27%
		1,25	2	0,00%	0,00%	7,03%	0,00%	39,86%	0,00%
			3	0,00%	0,00%	0,27%	0,00%	30,39%	4,51%
			4	0,00%	0,00%	0,50%	0,50%	19,83%	2,03%
			5	0,00%	0,00%	0,09%	0,00%	24,67%	2,64%
	2	1	2	0,00%	0,00%	0,00%	0,00%	39,95%	0,00%
			3	0,00%	0,00%	0,00%	0,00%	19,38%	2,24%
			4	0,00%	0,00%	0,00%	0,00%	41,60%	15,48%
			5	0,00%	0,00%	0,42%	0,29%	34,07%	5,06%
		1,25	2	0,00%	0,00%	2,82%	0,00%	31,44%	0,00%
			3	0,00%	0,00%	0,00%	0,00%	40,61%	2,14%
			4	0,00%	0,00%	0,00%	0,00%	35,29%	2,36%
			5	0,00%	0,00%	0,08%	0,00%	29,57%	3,59%
3	1,5	1	2	0,00%	0,00%	0,00%	0,00%	43,49%	0,00%
			3	0,00%	0,00%	1,63%	0,12%	31,21%	3,96%
			4	0,00%	0,00%	0,13%	0,00%	43,56%	17,17%
			5	0,00%	0,26%	0,50%	0,03%	34,98%	9,60%
		1,25	2	0,00%	0,00%	0,00%	0,00%	33,27%	0,00%
			3	0,00%	0,00%	2,25%	1,15%	35,30%	8,77%
			4	0,00%	0,00%	2,01%	0,72%	33,69%	12,36%
			5	0,00%	1,21%	0,56%	0,12%	28,84%	7,51%
	2	1	2	0,00%	0,00%	0,00%	0,00%	20,26%	0,00%
			3	0,00%	0,00%	2,05%	0,96%	34,01%	2,51%
			4	0,00%	0,00%	0,58%	0,22%	39,72%	12,25%
			5	0,00%	0,59%	0,84%	0,00%	42,38%	7,85%
		1,25	2	0,00%	0,00%	0,00%	0,00%	39,17%	0,00%
			3	0,00%	0,00%	0,87%	0,57%	41,10%	3,97%
			4	0,00%	0,00%	1,20%	0,94%	40,56%	9,22%
			5	0,00%	0,56%	0,31%	0,00%	31,00%	4,20%
4	1,5	1	2	0,00%	0,00%	0,00%	0,00%	17,50%	0,00%
			3	1,42%	0,00%	1,42%	1,42%	33,69%	9,81%
			4	0,00%	0,00%	1,56%	0,14%	39,98%	19,01%
			5	0,00%	0,61%	0,88%	0,15%	33,76%	13,14%
		1,25	2	0,00%	0,00%	0,00%	0,00%	28,08%	0,00%
			3	0,00%	0,00%	0,00%	0,00%	28,81%	6,71%
			4	0,00%	0,11%	2,31%	0,96%	37,06%	15,03%
			5	0,00%	0,74%	1,43%	0,39%	26,61%	8,05%
	2	1	2	0,00%	0,00%	0,00%	0,00%	36,25%	0,00%
			3	0,00%	0,00%	0,00%	0,00%	42,69%	10,09%
			4	0,00%	0,00%	0,93%	0,00%	49,05%	16,35%
			5	0,00%	0,34%	1,09%	0,00%	42,04%	16,26%
		1,25	2	0,00%	0,00%	0,00%	0,00%	40,34%	0,00%
			3	0,00%	0,00%	1,12%	0,52%	39,24%	10,04%
			4	0,00%	0,00%	0,48%	0,00%	43,70%	11,22%
			5	0,00%	1,93%	1,29%	0,36%	37,62%	5,47%

Tabla 30: Valores de RPD[1] (iteración 1)

J	α	β	H	MB - RPD1	M1 - RPD1	M2 - RPD1	M3 - RPD1	M4 - RPD1	M5 - RPD1
2	1,5	1	2	0,00%	0,00%	0,00%	0,00%	13,43%	0,00%
			3	0,00%	0,00%	0,24%	0,00%	42,57%	8,94%
			4	0,00%	0,00%	0,07%	0,00%	29,42%	8,38%
			5	0,00%	0,00%	0,44%	0,44%	28,03%	5,49%
		1,25	2	0,00%	0,00%	0,00%	0,00%	39,94%	0,00%
			3	0,00%	0,00%	0,27%	0,27%	22,02%	0,92%
			4	0,00%	0,00%	0,47%	0,06%	22,77%	6,75%
			5	0,00%	0,03%	0,95%	0,07%	20,28%	4,92%
	2	1	2	0,00%	0,00%	0,00%	0,00%	47,29%	0,00%
			3	0,00%	0,00%	0,89%	0,89%	29,84%	1,34%
			4	0,00%	0,00%	0,27%	0,00%	29,90%	9,09%
			5	0,00%	0,00%	0,12%	0,10%	33,79%	5,56%
		1,25	2	0,00%	0,00%	0,00%	0,00%	34,45%	0,00%
			3	0,00%	0,00%	0,30%	0,30%	38,83%	2,22%
			4	0,00%	0,00%	0,00%	0,00%	36,45%	8,24%
			5	0,00%	0,00%	0,58%	0,02%	26,33%	3,71%
3	1,5	1	2	0,00%	0,00%	0,00%	0,00%	23,73%	0,00%
			3	0,00%	0,00%	0,00%	0,00%	38,96%	10,82%
			4	0,00%	0,00%	1,68%	0,00%	33,71%	12,36%
			5	0,00%	1,35%	0,10%	0,00%	36,25%	8,42%
		1,25	2	0,00%	0,00%	0,00%	0,00%	27,75%	0,00%
			3	0,00%	0,00%	0,35%	0,22%	30,33%	4,02%
			4	0,00%	0,00%	0,47%	0,00%	33,86%	15,31%
			5	0,00%	0,73%	0,39%	0,08%	26,84%	8,81%
	2	1	2	0,00%	0,00%	0,00%	0,00%	10,06%	0,00%
			3	0,00%	0,00%	0,33%	0,33%	51,84%	6,03%
			4	0,01%	0,00%	1,62%	0,00%	42,04%	5,96%
			5	0,00%	0,71%	0,15%	0,03%	36,36%	11,19%
		1,25	2	0,00%	0,00%	7,64%	0,00%	58,78%	0,00%
			3	0,00%	0,00%	0,87%	0,79%	29,61%	4,76%
			4	0,00%	1,09%	0,97%	0,09%	39,14%	6,18%
			5	0,00%	0,51%	0,12%	0,10%	33,45%	7,19%
4	1,5	1	2	0,00%	0,00%	0,00%	0,00%	26,37%	0,00%
			3	0,00%	0,00%	0,00%	0,00%	22,21%	7,77%
			4	0,00%	0,00%	1,32%	0,55%	34,78%	16,20%
			5	0,00%	1,54%	1,57%	1,25%	35,82%	17,55%
		1,25	2	0,00%	0,00%	0,00%	0,00%	26,71%	0,00%
			3	0,00%	0,00%	1,11%	0,00%	40,93%	4,80%
			4	0,00%	0,12%	0,81%	0,03%	37,79%	11,72%
			5	0,00%	0,11%	0,36%	0,00%	38,94%	10,33%
	2	1	2	0,00%	0,00%	0,00%	0,00%	25,96%	0,00%
			3	0,00%	0,00%	0,00%	0,00%	42,66%	7,71%
			4	0,00%	0,00%	0,47%	0,19%	44,24%	10,80%
			5	0,00%	0,87%	0,43%	0,19%	43,05%	12,41%
		1,25	2	0,00%	0,00%	6,46%	0,00%	35,60%	0,00%
			3	0,00%	0,00%	0,00%	0,00%	39,56%	13,91%
			4	0,00%	0,00%	1,38%	0,00%	42,49%	13,06%
			5	0,00%	0,30%	0,38%	0,00%	32,25%	4,89%

Tabla 31: Valores de RPD[1] (iteración 2)

J	α	β	H	M1 - RPD2	M2 - RPD2	M3 - RPD2	M4 - RPD2	M5 - RPD2
2	1,5	1	2	0,00%	0,00%	0,00%	33,15%	0,00%
			3	0,00%	0,00%	0,00%	24,63%	1,04%
			4	0,00%	0,11%	0,10%	21,10%	6,31%
			5	-	0,23%	0,00%	28,01%	7,27%
		1,25	2	0,00%	7,03%	0,00%	39,86%	0,00%
			3	0,00%	0,27%	0,00%	30,39%	4,51%
			4	-	0,50%	0,50%	19,83%	2,03%
			5	-	0,09%	0,00%	24,67%	2,64%
	2	1	2	0,00%	0,00%	0,00%	39,95%	0,00%
			3	0,00%	0,00%	0,00%	19,38%	2,24%
			4	0,00%	0,00%	0,00%	41,60%	15,48%
			5	0,00%	0,42%	0,29%	34,07%	5,06%
		1,25	2	0,00%	2,82%	0,00%	31,44%	0,00%
			3	0,00%	0,00%	0,00%	40,61%	2,14%
			4	0,00%	0,00%	0,00%	35,29%	2,36%
			5	-	0,08%	0,00%	29,57%	3,59%
3	1,5	1	2	0,00%	0,00%	0,00%	43,49%	0,00%
			3	0,00%	1,63%	0,12%	31,21%	3,96%
			4	-	0,13%	0,00%	43,56%	17,17%
			5	-	-	-	34,98%	9,60%
		1,25	2	0,00%	0,00%	0,00%	33,27%	0,00%
			3	0,00%	2,25%	1,15%	35,30%	8,77%
			4	-	2,01%	0,72%	33,69%	12,36%
			5	-	0,56%	0,12%	28,84%	7,51%
	2	1	2	0,00%	0,00%	0,00%	20,26%	0,00%
			3	0,00%	2,05%	0,96%	34,01%	2,51%
			4	0,00%	0,58%	0,22%	39,72%	12,25%
			5	-	-	0,00%	42,38%	7,85%
		1,25	2	0,00%	0,00%	0,00%	39,17%	0,00%
			3	0,00%	0,87%	0,57%	41,10%	3,97%
			4	-	1,20%	0,94%	40,56%	9,22%
			5	-	0,30%	0,00%	31,00%	4,19%
4	1,5	1	2	0,00%	0,00%	0,00%	17,50%	0,00%
			3	-1,44%	0,00%	0,00%	32,73%	8,51%
			4	-	1,56%	0,14%	39,98%	19,01%
			5	-	-	-	33,76%	13,14%
		1,25	2	0,00%	0,00%	0,00%	28,08%	0,00%
			3	0,00%	0,00%	0,00%	28,81%	6,71%
			4	-	2,31%	0,96%	37,06%	15,03%
			5	-	-	-	26,61%	-
	2	1	2	0,00%	0,00%	0,00%	36,25%	0,00%
			3	0,00%	0,00%	0,00%	42,69%	10,09%
			4	0,00%	0,93%	0,00%	49,05%	16,35%
			5	-	1,09%	-	42,04%	16,26%
		1,25	2	0,00%	0,00%	0,00%	40,34%	0,00%
			3	-	1,12%	0,52%	39,24%	10,04%
			4	0,00%	0,48%	0,00%	43,70%	11,22%
			5	-	1,29%	-	37,62%	5,47%

Tabla 32: Valores de RPD[2] (iteración 1)

J	α	β	H	M1 - RPD2	M2 - RPD2	M3 - RPD2	M4 - RPD2	M5 - RPD2
2	1,5	1	2	0,00%	0,00%	0,00%	13,43%	0,00%
			3	0,00%	0,24%	0,00%	42,57%	8,94%
			4	0,00%	0,07%	0,00%	29,42%	8,38%
			5	-	0,44%	0,44%	28,03%	5,49%
		1,25	2	0,00%	0,00%	0,00%	39,94%	0,00%
			3	0,00%	0,27%	0,27%	22,02%	0,92%
			4	0,00%	0,47%	0,06%	22,77%	6,75%
			5	-	0,95%	0,07%	20,28%	4,92%
	2	1	2	0,00%	0,00%	0,00%	47,29%	0,00%
			3	0,00%	0,89%	0,89%	29,84%	1,34%
			4	0,00%	0,27%	0,00%	29,90%	9,09%
			5	0,00%	0,12%	0,10%	33,79%	5,56%
		1,25	2	0,00%	0,00%	0,00%	34,45%	0,00%
			3	0,00%	0,30%	0,30%	38,83%	2,22%
			4	0,00%	0,00%	0,00%	36,45%	8,24%
			5	-	0,58%	0,02%	26,33%	3,71%
3	1,5	1	2	0,00%	0,00%	0,00%	23,73%	0,00%
			3	0,00%	0,00%	0,00%	38,96%	10,82%
			4	0,00%	1,68%	0,00%	33,71%	12,36%
			5	-	0,10%	0,00%	36,25%	8,42%
		1,25	2	0,00%	0,00%	0,00%	27,75%	0,00%
			3	0,00%	0,35%	0,22%	30,33%	4,02%
			4	-	0,47%	0,00%	33,86%	15,31%
			5	-	-	-	26,84%	8,81%
	2	1	2	0,00%	0,00%	0,00%	10,06%	0,00%
			3	0,00%	0,33%	0,33%	51,84%	6,03%
			4	-0,01%	1,61%	-0,01%	42,04%	5,96%
			5	-	0,15%	0,03%	36,36%	11,19%
		1,25	2	0,00%	7,64%	0,00%	58,78%	0,00%
			3	0,00%	0,87%	0,79%	29,61%	4,76%
			4	-	0,97%	0,09%	39,14%	6,18%
			5	-	0,12%	0,10%	33,45%	7,19%
4	1,5	1	2	0,00%	0,00%	0,00%	26,37%	0,00%
			3	0,00%	0,00%	0,00%	22,21%	7,77%
			4	-	1,32%	0,55%	34,78%	16,20%
			5	-	1,57%	1,25%	35,82%	17,55%
		1,25	2	0,00%	0,00%	0,00%	26,71%	0,00%
			3	0,00%	1,11%	0,00%	40,93%	4,80%
			4	-	0,81%	0,03%	37,79%	11,72%
			5	-	0,36%	0,00%	38,94%	10,33%
	2	1	2	0,00%	0,00%	0,00%	25,96%	0,00%
			3	0,00%	0,00%	0,00%	42,66%	7,71%
			4	0,00%	0,47%	0,19%	44,24%	10,80%
			5	-	0,43%	0,19%	43,05%	12,41%
		1,25	2	0,00%	6,46%	0,00%	35,60%	0,00%
			3	0,00%	0,00%	0,00%	39,56%	13,91%
			4	-	1,38%	0,00%	42,49%	13,06%
			5	-	-	-	32,25%	-

Tabla 33: Valores de RPD[2] (iteración 2)

6.3. Análisis de los resultados

Una vez obtenidos todos los resultados y calculados los parámetros RPD_1 y RPD_2 , podemos sacar ciertas conclusiones de nuestro estudio. Al ser nuestro principal objetivo comparar el comportamiento de los modelos alternativos frente al Modelo Base, realizaremos el análisis de cada uno de los modelos tomando como referencia los resultados obtenidos en este.

El Modelo Base es el tercer modelo mejor posicionado en la comparación de número de óptimos obtenidos (91 de las 96 instancias estudiadas han resultado en una solución óptima) y de tiempo promedio de ejecución. Es el modelo que presenta un valor menor de $ARPD_1$ (0,01%), lo que nos ayuda a concluir que es el Modelo en el que encontramos los valores más satisfactorios para nuestra función objetivo al comparar tanto valores óptimos como no óptimos.

En el Modelo 1 encontramos el menor número de óptimos encontrado (62 de 96 instancias) y un valor de tiempo promedio de 242,2876 s. Sin embargo, cuando estudiamos los valores de $ARPD_1$ y $ARPD_2$, encontramos muy buenos resultados: $ARPD_1 = 0,14\%$ (segundo valor más bajo) y $ARPD_2 = -0,02\%$ (valor más bajo). Pese a que el valor de $ARPD_1$ nos lleva a afirmar que encontramos unos valores ligeramente menores en este modelo que en el Modelo Base (ya que este presenta un valor menor para el parámetro), el valor de $ARPD_2$ nos indica lo contrario (debido a su signo negativo). Esto es consecuencia de que en el segundo parámetro solo estamos comparando aquellas instancias donde ambos modelos han sido capaces de llegar a la solución óptima del problema, lo que nos lleva a confirmar que al comparar todos los resultados encontramos mejores valores en el Modelo Base pero que si solo tenemos en cuenta aquellas soluciones óptimas, es el Modelo 1 el que presenta mejores resultados. Comparando los tiempos de ejecución con los del Modelo Base vemos que este último ha necesitado menor tiempo en todas las instancias para alcanzar el resultado final.

En los Modelos 2 y 3 encontramos resultados parecidos: únicamente se diferencian por alcanzar 1 óptimo más en el caso del Modelo 2, por 7,3786 s entre sus tiempos medios de ejecución y por algunos decimales en los valores de $ARPD_1$ (0,74% el Modelo 1 y 0,16% el Modelo 3) y $ARPD_2$ (0,72% el Modelo 1 y 0,15% el Modelo 3). Debido a estos datos, podemos concluir que encontramos rendimientos parecidos en ambos modelos entre ellos y el Modelo Base. Sin embargo, ninguno mejora los resultados alcanzados por este último. Ambos modelos presentan una comparación con los tiempos del Modelo Base equilibrada: en el caso del Modelo 2 ha necesitado un tiempo mayor que el Modelo Base en el 46% de las instancias y el Modelo 3 en el 47%.

El Modelo 4 es el que presenta unos resultados más distantes respecto al Modelo Base. Es donde encontramos los valores mayores para $ARPD_1$ y $ARPD_2$, lo que nos indica que no solo es el modelo donde hemos alcanzado unas soluciones menores, sino que además es el que mayor diferencia presenta con el Modelo Base. Es el único caso donde se ha logrado alcanzar la solución óptima en cada una de las instancias estudiadas. Además, es el modelo con menor tiempo promedio de ejecución (0,1381 s) y el que ha requerido menor tiempo que el Modelo Base en el mayor número de ocasiones: únicamente le ha superado al Modelo Base en 12 de las 91 instancias estudiadas.

Por último, encontramos el Modelo 5. En este hemos obtenido el segundo mejor ratio de óptimos alcanzados y tiempo medio, pero el segundo con peores resultados en los parámetros estudiados. Pese a no encontrar una diferencia tan grande como la del Modelo 4, sí que está distanciado a los valores tan bajos encontrados en el resto de modelos. Es el único, junto al Modelo 1, que ha requerido un tiempo de ejecución mayor en más instancias óptimas que el Modelo Base, aunque en esta ocasión la diferencia es únicamente del 2% de estas instancias.

7. CONCLUSIONES

Una vez diseñados los modelos, recogidos sus resultados y realizado un análisis de estos, podemos pasar a sacar las conclusiones del estudio para dar por finalizado este. Antes de comenzar con ellas, me gustaría hacer un breve resumen del alcance del estudio. Hemos partido del modelo actual de planificación de las operaciones en los quirófanos del Hospital Virgen del Rocío de Sevilla y hemos diseñado 5 políticas alternativas a este. Mientras que en la planificación actual el cirujano encargado de cada intervención está asignado previamente, en el Modelo 1 se ha estudiado el comportamiento del modelo si esta relación fuese tratada como una variable más del problema. En el modelo de referencia los cirujanos pueden operar en todos los quirófanos disponibles en el hospital, mientras que en el Modelo 2 solo lo pueden hacer en un quirófano durante todo el día y en el Modelo 3 en un número comprendido entre 1 y el número total de ellos. En el Modelo 4 hemos estudiado una política donde no solo el cirujano puede operar en un único quirófano, sino que además será él el único que puede hacer intervenciones ahí durante ese día. Por último, hemos impuesto en un nuevo modelo que los cirujanos solo puedan trabajar un día en su semana laboral. Todas estas variantes estudiadas nos llevan a pensar en la infinitud de políticas diferentes que se pueden seguir en cada hospital y en la cantidad de factores que, por insignificantes que parezcan, pueden influir en esta planificación.

Diseñado el código y ejecutado el programa, recogimos los datos obtenidos y comparamos sus resultados. Vemos que el único modelo alternativo que mejora los resultados obtenidos por el Modelo Base es el Modelo 1 (en el cual la asignación de los cirujanos a cada intervención es tratada como una variable más de decisión). A pesar de no llegar al óptimo en un alto número de instancias, aquellas instancias donde hemos llegado al valor óptimo presentan un valor ligeramente mejor que el Modelo Base. Sería interesante estudiar este modelo con otro método de resolución (como podrían ser los algoritmos heurísticos) para comprobar si así alcanzamos mayor número de óptimos y si estos son mejores que los encontrados por el Modelo Base.

Resultan lógicos los rendimientos de cada uno de los modelos. Al dejar más libertad para realizar las operaciones en el Modelo 1, es coherente que podamos realizar un mayor número de intervenciones durante el día. En el lado contrario, tenemos al Modelo 4, donde encontramos los peores resultados. Al basarse en una política donde los quirófanos son ocupados únicamente por un cirujano al día, los bajos números alcanzados obedecen al incremento de limitaciones impuesto en esta política.

El uso de un método de resolución alternativo también sería interesante para verificar si así obtenemos resultados diferentes entre aquellos modelos que nos han brindado unos resultados similares (Modelos 2 y 3) y conocer así si realmente no son determinantes las diferencias en la definición de aquellos factores.

Otra posible ampliación del estudio sería el aumento del número de parámetros incluidos en el estudio. Como hemos comentado anteriormente, los diseños de estas políticas en la vida real tienen en consideración muchos más factores de los que hemos considerado en nuestro estudio. Es por ello por lo que, si se quiere hacer un estudio más profundo y más cercano a la realidad, habría que ampliar considerablemente el tamaño del problema. Algunos de estos factores son la disponibilidad de los recursos pre y post-operatorios, los tiempos necesarios de descanso y de adecuación del quirófano entre dos operaciones y las limitaciones económicas existentes en un hospital.

Para nuestro estudio hemos priorizado como objetivo el alcance del máximo peso clínico de las operaciones realizadas, priorizando su intervención en el día más temprano posible. Sin embargo, no en todos los hospitales se hace así ya que el factor económico es fundamental en la mayoría de ellos. Siguiendo esta línea, podríamos realizar una nueva ampliación del estudio enfocándola de dos formas distintas: basar nuestra función objetivo

únicamente en la optimización de los recursos económicos o incluir en la función actual un factor en relación a ellos. Con este último método se estaría priorizando tanto la atención a nuestros pacientes como el ahorro económico.

Como conclusión final, podríamos afirmar que la política actual de planificación de los quirófanos en el Hospital Virgen del Rocío de Sevilla presenta unos buenos resultados frente a las alternativas estudiadas en este documento. Únicamente serían ligeramente mejorados con la implantación del Modelo 1, dejando la asignación de los cirujanos a las intervenciones como una variable de decisión más a definir en la planificación. Sin embargo, al no ser tan grande la diferencia entre los resultados de ambas políticas, podría ser más conveniente continuar con el modelo actual ya que las posibles desventajas derivadas por el cambio de política (adaptación de los recursos usados para la planificación, conformidad de los profesionales afectados por los cambios, ...) podrían ser más significativos que la ligera mejora que alcanzaríamos con la nueva política.

REFERENCIAS

- Fei, H., Meskens, N., & Chu, C. (marzo de 2010). A planning and scheduling problem for an operating theatre using an open scheduling strategy. *Computers & Industrial Engineering*, 58, 221-230.
- González Duque, R. (2011). Python para todos.
- Listas de espera en la Sanidad pública, últimos datos oficiales. (3 de mayo de 2022). Obtenido de epdata: <https://www.epdata.es/datos/listas-espera-sanidad-publica/24/espana/106>
- Llewelyn-Davies, R., & Macaulay, H. (1966). Planificación y administración de hospitales. *Publicación científica*(191).
- Molina-Pariente, J. M., Hans, E. W., Framinan, J. M., & Gómez-Cía, T. (2015). New heuristics for planning operating rooms. *Computers & Industrial Engineering*, 429-443.
- Nieto Medina, Á. (2021). *Trabajo de Fin de Grado: Integración de planificación de intervenciones en quirófanos y camas postoperatorias*. Escuela Técnica de Ingeniería (Universidad de Sevilla), Sevilla.
- Ponce Briones, D. K. (2016). *Proyecto de Titulación: Análisis comparativo de los Entorno de Desarrollo Integrados (IDE): Eclipse, Netbeans y JDeveloper para el desarrollo de aplicaciones Java Enterprise Edition*. Facultad de Ciencias Matemáticas y Físicas (Universidad de Guayaquil), Guayaquil.
- Przasnyski, Z. (julio de 1986). Operating Room Scheduling: A Literature Review. *AORN Journal*, 44(1), 67-68, 70-72, 74-76, 78-79, 82.
- Ramírez Rojas, M. d. (2020). *Trabajo Fin de Grado: Integración de planificación de consultas y quirófanos en el sector sanitario*. Escuela Técnica de Ingenieros (Universidad de Sevilla), Sevilla.
- Ramos, A., Sánchez, P., Ferrer, J. M., Barquín, J., & Linares, P. (2010). *Modelos matemáticos de optimización* (Vol. 1). Madrid: Publicación Técnica.
- Rodríguez Perera, F., & Peiró, M. (Agosto de 2012). La planificación estratégica en las organizaciones sanitarias. *Revista Española de Cardiología*, 65(8), 749-754.
- Rothstein, D., & Raval, M. (2018). Operating room efficiency. *Seminars in Pediatric Surgery* 27, (págs. 79-85).
- Terrazas Pastor, R. (2011). Planificación y programación de operaciones. *Revista Perspectivas*, 28, 7-32.
- Tiempos de respuesta asistencial. Listas de espera*. (s.f.). Obtenido de Servicio Andaluz de Salud. Consejería de Salud y Familias: <https://www.sspa.juntadeandalucia.es/servicioandaluzdesalud/ciudadania/derechos-y-garantias/tiempos-de-respuesta-asistencial-listas-de-espera>
- TimeLimit*. (s.f.). Obtenido de Gurobi Optimization: <https://www.gurobi.com/documentation/9.5/refman/timelimit.html>
- Vilcarromero Ruiz, R. (2017). *Gestión de la Producción*.
- Winston, W. (2005). *Investigacion de Operaciones: Aplicaciones y Algoritmos* (4ª ed.). Thomson.

```
import random
from math import *
from gurobipy import *
from xlwt import Workbook

iteraciones = 2
J = [2, 3, 4]
alpha = [1.5, 2]
beta = [1, 1.25]
H = [2, 3, 4, 5]
aw = 0.5
mds = 1
rjh=480
resultados = Workbook()

#Funcion GeneracionInstancia
def GenerarInstancia(j, al, b, h):
    #Definicion r[j][h]
    r = [[rjh for hh in range(h)] for jj in range(j)]

    #Definicion W
    W = ceil(h / 7)

    #Definicion S
    suma_r = 0
    for jj in range(j):
        for hh in range(h):
            suma_r += r[jj][hh]

    S = ceil((al*suma_r)/(W*rjh*mds))

    #Definicion I, t[i]. gamma[i], rd[i]
    I = 0
    suma_t=0
    t = []
```

```

gamma = []
rd = []
while suma_t <= suma_r*b:
    I += 1
    t.append(random.uniform(90,120))
    suma_t += t[I-1]
    gamma.append(random.randint(0, S-1))
    rd.append(random.randint(1,j))

#Definicion a[s][h]
a = [[rjh for hh in range(h)] for s in range(S)]

#Definicion d[i]
MTBT = []
dwl = []
d = []
for i in range(I):
    MTBT.append(random.choice([45, 180, 360]))
    dwl.append(random.randint(1, MTBT[i] - 1))
    d.append(MTBT[i] - dwl[i])

#Definicion w[i]
w = []
mp = []
for i in range(I):
    mp.append(random.randint(1, 5))
    w.append((aw*mp[i]/5+(1-aw)*dwl[i]/MTBT[i]))

#Definicion delta[i][j][h]
J_especializados = round(0.3*j)
if (J_especializados == 0):
    J_especializados = 1 #nos aseguramos de que hay mínimo 1 quirófano especializado

porcentaje = []
delta = [[[1 for hh in range(h)] for jj in range(j)] for i in range(I)]
for i in range(I):
    porcentaje.append(random.uniform(0,1))

```

```

if (porcentaje[i] < 0.1): #si la operacion es especializada
    for jj in range(j):
        if (jj >= J_especializados):
            for hh in range(h):
                delta[i][jj][hh] = 0 #asignamos valor 0 a los quirofanos no especializados

#Definición tiempo de parada
t_limite = 600

return(I, S, r, a, gamma, rd, d, delta, t, w, mp, MTBT, dwl, t_limite)

def ModeloBase(I, S, r, a, gamma, rd, d, delta, t, w, t_parada):
    # Inicializamos modelo
    modelo = Model("Modelo base")

    # Creación de variables
    X = modelo.addMVar((I, j, h), vtype=GRB.BINARY, name='X') # Restricción 10
    Z = modelo.addMVar((S, j, h), vtype=GRB.BINARY, name='Z') # Restricción 11

    modelo.setObjective(
        quicksum(quicksum(w[i] * X[i][jj][hh] for i in range(I) for jj in range(j)) / (hh + 1) for hh in range(h)),
        GRB.MAXIMIZE)

    # Definición de restricciones

    # Restricción 1
    modelo.addConstrs((quicksum(X[i][jj][hh] for jj in range(j) for hh in range(h)) <= 1 for i in range(I)), "Rest1")

    # Restricción 2
    modelo.addConstrs(
        (quicksum(X[i][jj][hh] for jj in range(j) for hh in range(h) if hh <= (rd[i] - 1)) == 0 for i in range(I)),
        "Rest2")

    # Restricción 3
    modelo.addConstrs(
        (quicksum(X[i][jj][hh] for jj in range(j) for hh in range(h) if hh > d[i]) == 0 for i in range(I) if
        d[i] <= abs(h - 1)), "Rest3")

```

```

# Restricción 4
modelo.addConstrs(
    (quicksum(t[i] * X[i][jj][hh] for i in range(I)) <= r[jj][hh] for jj in range(j) for hh in range(h)), "Rest4")

# Restricción 5
modelo.addConstrs(
    (quicksum((t[i] * X[i][jj][hh]) for i in range(I) if gamma[i] == s for jj in range(j)) <= a[s][hh] for s in
    range(S) for hh in range(h)), "Rest5")

# Restricción 6
modelo.addConstrs(
    (quicksum((t[i] * X[i][jj][hh]) for i in range(I) if gamma[i] == s) <= r[jj][hh] * Z[s][jj][hh] for s in
    range(S) for jj in range(j) for hh in range(h)), "Rest6")

# Restricción 7
modelo.addConstrs(
    (quicksum((t[i] * X[i][jj][hh]) for i in range(I) if gamma[i] == s) >= Z[s][jj][hh] for s in range(S) for jj in
    range(j) for hh in range(h)), "Rest7")

# Restricción 8
modelo.addConstrs(
    (X[i][jj][hh] == 0 for i in range(I) for jj in range(j) for hh in range(h) if delta[i][jj][hh] == 0), "Rest8")

# Resolución
print("\n")
print("Modelo BASE resolviéndose: J:", j, ", alpha", al, ", beta:", b, ", H:", h, ", iteración n°:", iter)
modelo.Params.timeLimit = t_parada
modelo.optimize()
print("Modelo BASE resuelto: J:", j, ", alpha", al, ", beta:", b, ", H:", h, ", iteración n°:", iter)

if (modelo.Status == 9):
    optimo = 0
    print("Tiempo excedido\n")
else:
    optimo = 1
    print("Óptimo encontrado\n")

obj = modelo.getObjective()

```

```

return(optimo, obj.getValue(), modelo.RunTime)

def ModeloAlt1(I, S, r, a, rd, d, delta, t, w, t_parada):
    # Inicializamos modelo
    modelo1 = Model("Modelo 1")

    # Creación de variables
    X1 = modelo1.addMVar((I, j, h), vtype=GRB.BINARY, name='X') # Restricción 10
    Z1 = modelo1.addMVar((S, j, h), vtype=GRB.BINARY, name='Z') # Restricción 11
    Y1 = modelo1.addMVar((I, S), vtype=GRB.BINARY, name='Y')

    modelo1.setObjective(
        quicksum(quicksum(w[i] * X1[i][jj][hh] for i in range(I) for jj in range(j)) / (hh + 1) for hh in range(h)),
        GRB.MAXIMIZE)

    # Definición de restricciones

    # Restricción 1
    modelo1.addConstrs((quicksum(X1[i][jj][hh] for jj in range(j) for hh in range(h)) <= 1 for i in range(I)),
        "Rest11")

    # Restricción 2
    modelo1.addConstrs(
        (quicksum(X1[i][jj][hh] for jj in range(j) for hh in range(h) if hh <= (rd[i] - 1)) == 0 for i in range(I)),
        "Rest21")

    # Restricción 3
    modelo1.addConstrs(
        (quicksum(X1[i][jj][hh] for jj in range(j) for hh in range(h) if hh > d[i]) == 0 for i in range(I) if
        d[i] <= abs(h - 1)), "Rest31")

    # Restricción 4
    modelo1.addConstrs(
        (quicksum(t[i] * X1[i][jj][hh] for i in range(I)) <= r[jj][hh] for jj in range(j) for hh in range(h)), "Rest41")

    # Restricción 5
    modelo1.addConstrs(
        (quicksum(t[i] * X1[i][jj][hh] * Y1[i][s] for i in range(I) for jj in range(j)) <= a[s][hh] for s in range(S)
        for hh in range(h)), "Rest51")

```

```

# Restricción 6
modelo1.addConstrs(
    (quicksum(t[i] * X1[i][jj][hh] * Y1[i][s] for i in range(I)) <= r[jj][hh] * Z1[s][jj][hh] for s in range(S) for
    jj in range(j) for hh in range(h)), "Rest61")

# Restricción 7
modelo1.addConstrs(
    (quicksum(t[i] * X1[i][jj][hh] * Y1[i][s] for i in range(I)) >= Z1[s][jj][hh] for s in range(S) for jj in
    range(j) for hh in range(h)), "Rest71")

# Restricción 8
modelo1.addConstrs(
    (X1[i][jj][hh] == 0 for i in range(I) for jj in range(j) for hh in range(h) if delta[i][jj][hh] == 0), "Rest81")

# Restricciones Nuevas
modelo1.addConstrs(
    (quicksum(Y1[i][s] * Z1[s][jj][hh] for s in range(S)) == X1[i][jj][hh] for i in range(I) for jj in range(j) for
    hh in range(h)), "Rest*1")
modelo1.addConstrs(
    (quicksum(X1[i][jj][hh] * Z1[s][jj][hh] for jj in range(j) for hh in range(h)) >= Y1[i][s] for s in range(S) for
    i in range(I)), "Rest**1")

# Resolución modelo
print("\n")
print("Modelo 1 resolviéndose: J:", j, ", alpha", al, ", beta:", b, ", H:", h, ", iteración nº:", iter)
modelo1.Params.timeLimit = t_parada
modelo1.optimize()
print("Modelo 1 resuelto: J:", j, ", alpha", al, ", beta:", b, ", H:", h, ", iteración nº:", iter)

if (modelo1.Status == 9):
    optimo = 0
    print("Tiempo excedido\n")
else:
    optimo = 1
    print("Óptimo encontrado\n")

obj = modelo1.getObjective()

```

```

return (optimo, obj.getValue(), modelo1.RunTime)

def ModeloAlt2(I, S, r, a, gamma, rd, d, delta, t, w, t_parada):
    # Inicializamos modelo
    modelo2 = Model("Modelo 2")

    # Creación de variables
    X2 = modelo2.addMVar((I, j, h), vtype=GRB.BINARY, name='X') # Restricción 10
    Z2 = modelo2.addMVar((S, j, h), vtype=GRB.BINARY, name='Z') # Restricción 11

    modelo2.setObjective(
        quicksum(quicksum(w[i] * X2[i][jj][hh] for i in range(I) for jj in range(j)) / (hh + 1) for hh in range(h)),
        GRB.MAXIMIZE)

    # Definición de restricciones

    # Restricción 1
    modelo2.addConstrs((quicksum(X2[i][jj][hh] for jj in range(j) for hh in range(h)) <= 1 for i in range(I)),
        "Rest12")

    # Restricción 2
    modelo2.addConstrs(
        (quicksum(X2[i][jj][hh] for jj in range(j) for hh in range(h) if hh <= (rd[i] - 1)) == 0 for i in range(I)),
        "Rest22")

    # Restricción 3
    modelo2.addConstrs(
        (quicksum(X2[i][jj][hh] for jj in range(j) for hh in range(h) if hh > d[i]) == 0 for i in range(I) if
        d[i] <= abs(h - 1)), "Rest32")

    # Restricción 4
    modelo2.addConstrs(
        (quicksum(t[i] * X2[i][jj][hh] for i in range(I)) <= r[jj][hh] for jj in range(j) for hh in range(h)), "Rest42")

    # Restricción 5
    modelo2.addConstrs(
        (quicksum((t[i] * X2[i][jj][hh]) for i in range(I) if gamma[i] == s for jj in range(j)) <= a[s][hh] for s in
        range(S) for hh in range(h)), "Rest52")

```

```

# Restricción 6
modelo2.addConstrs(
    (quicksum((t[i] * X2[i][jj][hh]) for i in range(I) if gamma[i] == s) <= r[jj][hh] * Z2[s][jj][hh] for s in
    range(S) for jj in range(j) for hh in range(h)), "Rest62")

# Restricción 7
modelo2.addConstrs(
    (quicksum((t[i] * X2[i][jj][hh]) for i in range(I) if gamma[i] == s) >= Z2[s][jj][hh] for s in range(S) for jj
    in
    range(j) for hh in range(h)), "Rest72")

# Restricción 8
modelo2.addConstrs(
    (X2[i][jj][hh] == 0 for i in range(I) for jj in range(j) for hh in range(h) if delta[i][jj][hh] == 0), "Rest82")

# Restricción nueva
modelo2.addConstrs((quicksum(Z2[s][jj][hh] for jj in range(j)) <= 1 for s in range(S) for hh in range(h)))
# Resolución
print("\n")
print("Modelo Alternativo 2 resolviéndose: J:", j, ", alpha", al, ", beta:", b, ", H:", h, ", iteración n°:", iter)
modelo2.Params.timeLimit = t_parada
modelo2.optimize()
print("Modelo Alternativo 2 resuelto: J:", j, ", alpha", al, ", beta:", b, ", H:", h, ", iteración n°:", iter)

if (modelo2.Status == 9):
    optimo = 0
    print("Tiempo excedido\n")
else:
    optimo = 1
    print("Óptimo encontrado\n")

obj = modelo2.getObjective()
return (optimo, obj.getValue(), modelo2.RunTime)

def ModeloAlt3(I, S, r, a, gamma, rd, d, delta, t, w, t_parada):
    # Definimos nuevo parámetro u[s]
    u3 = [round(random.uniform(1, j)) for s in range(S)]
    print("u[s]")

```

```

for s in range(S):
    print([s], u3[s])

# Inicializamos modelo
modelo3 = Model("Modelo base")

# Creación de variables
X3 = modelo3.addMVar((I, j, h), vtype=GRB.BINARY, name='X') # Restricción 10
Z3 = modelo3.addMVar((S, j, h), vtype=GRB.BINARY, name='Z') # Restricción 11

modelo3.setObjective(
    quicksum(quicksum(w[i] * X3[i][jj][hh] for i in range(I) for jj in range(j)) / (hh + 1) for hh in range(h)),
    GRB.MAXIMIZE)

# Definición de restricciones

# Restricción 1
modelo3.addConstrs((quicksum(X3[i][jj][hh] for jj in range(j) for hh in range(h)) <= 1 for i in range(I)),
    "Rest13")

# Restricción 2
modelo3.addConstrs(
    (quicksum(X3[i][jj][hh] for jj in range(j) for hh in range(h) if hh <= (rd[i] - 1)) == 0 for i in range(I)),
    "Rest23")

# Restricción 3
modelo3.addConstrs(
    (quicksum(X3[i][jj][hh] for jj in range(j) for hh in range(h) if hh > d[i]) == 0 for i in range(I) if
    d[i] <= abs(h - 1)), "Rest33")

# Restricción 4
modelo3.addConstrs(
    (quicksum(t[i] * X3[i][jj][hh] for i in range(I)) <= r[jj][hh] for jj in range(j) for hh in range(h)), "Rest43")

# Restricción 5
modelo3.addConstrs(
    (quicksum((t[i] * X3[i][jj][hh]) for i in range(I) if gamma[i] == s for jj in range(j)) <= a[s][hh] for s in
    range(S) for hh in range(h)), "Rest53")

```

```

# Restricción 6
modelo3.addConstrs(
    (quicksum((t[i] * X3[i][jj][hh]) for i in range(I) if gamma[i] == s) <= r[jj][hh] * Z3[s][jj][hh] for s in
    range(S) for jj in range(j) for hh in range(h)), "Rest63")

# Restricción 7
modelo3.addConstrs(
    (quicksum((t[i] * X3[i][jj][hh]) for i in range(I) if gamma[i] == s) >= Z3[s][jj][hh] for s in range(S) for jj
    in range(j) for hh in range(h)), "Rest73")

# Restricción 8
modelo3.addConstrs(
    (X3[i][jj][hh] == 0 for i in range(I) for jj in range(j) for hh in range(h) if delta[i][jj][hh] == 0), "Rest83")

# Restricción nueva
modelo3.addConstrs(quicksum(Z3[s][jj][hh] for jj in range(j)) <= u3[s] for s in range(S) for hh in range(h))
# Resolución
print("\n")
print("Modelo Alternativo 3 resolviéndose: J:", j, ", alpha", al, ", beta:", b, ", H:", h, ", iteración nº:", iter)
modelo3.Params.timeLimit = t_parada
modelo3.optimize()
print("Modelo Alternativo 3 resuelto: J:", j, ", alpha", al, ", beta:", b, ", H:", h, ", iteración nº:", iter)

if (modelo3.Status == 9):
    optimo = 0
    print("Tiempo excedido\n")
else:
    optimo = 1
    print("Óptimo encontrado\n")

obj = modelo3.getObjective()
return (optimo, obj.getValue(), modelo3.RunTime)

def ModeloAlt4(I, S, r, a, gamma, rd, d, delta, t, w, t_parada):
    # Inicializamos modelo
    modelo4 = Model("Modelo 4")

    # Creación de variables

```

```

X4 = modelo4.addMVar((I, j, h), vtype=GRB.BINARY, name='X') # Restricción 10
Z4 = modelo4.addMVar((S, j, h), vtype=GRB.BINARY, name='Z') # Restricción 11

modelo4.setObjective(
    quicksum(quicksum(w[i] * X4[i][jj][hh] for i in range(I) for jj in range(j)) / (hh + 1) for hh in range(h)),
    GRB.MAXIMIZE)

# Definición de restricciones

# Restricción 1
modelo4.addConstrs((quicksum(X4[i][jj][hh] for jj in range(j) for hh in range(h)) <= 1 for i in range(I)),
    "Rest14")

# Restricción 2
modelo4.addConstrs(
    (quicksum(X4[i][jj][hh] for jj in range(j) for hh in range(h) if hh <= (rd[i] - 1)) == 0 for i in range(I)),
    "Rest24")

# Restricción 3
modelo4.addConstrs(
    (quicksum(X4[i][jj][hh] for jj in range(j) for hh in range(h) if hh > d[i]) == 0 for i in range(I) if
    d[i] <= abs(h - 1)), "Rest34")

# Restricción 4
modelo4.addConstrs(
    (quicksum(t[i] * X4[i][jj][hh] for i in range(I)) <= r[jj][hh] for jj in range(j) for hh in range(h)), "Rest44")

# Restricción 5
modelo4.addConstrs(
    (quicksum((t[i] * X4[i][jj][hh]) for i in range(I) if gamma[i] == s for jj in range(j)) <= a[s][hh] for s in
    range(S) for hh in range(h)), "Rest54")

# Restricción 6
modelo4.addConstrs(
    (quicksum((t[i] * X4[i][jj][hh]) for i in range(I) if gamma[i] == s) <= r[jj][hh] * Z4[s][jj][hh] for s in
    range(S) for jj in range(j) for hh in range(h)), "Rest64")

# Restricción 7
modelo4.addConstrs(

```

```
(quicksum((t[i] * X4[i][jj][hh]) for i in range(I) if gamma[i] == s) >= Z4[s][jj][hh] for s in range(S) for jj
in range(j) for hh in range(h)), "Rest74")
```

```
# Restricción 8
```

```
modelo4.addConstrs(
    (X4[i][jj][hh] == 0 for i in range(I) for jj in range(j) for hh in range(h) if delta[i][jj][hh] == 0), "Rest84")
```

```
# Restricción Nueva1
```

```
#
```

```
modelo4.addConstrs((quicksum(Z4[s][jj][hh] for jj in range(j)) <= 1 for s in range(S) for hh in range(h)),
    "Rest*4")
```

```
# Restricción Nueva1
```

```
modelo4.addConstrs((quicksum(Z4[s][jj][hh] for s in range(S)) <= 1 for jj in range(j) for hh in range(h)),
    "Rest*4")
```

```
# Resolución
```

```
print("\n")
```

```
print("Modelo 4 resolviéndose: J:", j, ", alpha", al, ", beta:", b, ", H:", h, ", iteración nº:", iter)
```

```
modelo4.Params.timeLimit = t_parada
```

```
modelo4.optimize()
```

```
print("Modelo 4 resuelto: J:", j, ", alpha", al, ", beta:", b, ", H:", h, ", iteración nº:", iter)
```

```
if (modelo4.Status == 9):
```

```
    optimo = 0
```

```
    print("Tiempo excedido\n")
```

```
else:
```

```
    optimo = 1
```

```
    print("Óptimo encontrado\n")
```

```
obj = modelo4.getObjective()
```

```
return (optimo, obj.getValue(), modelo4.RunTime)
```

```
def ModeloAlt5(I, S, r, a, gamma, rd, d, delta, t, w, t_parada):
```

```
    # Inicializamos modelo
```

```
    modelo5 = Model("Modelo 5")
```

```
    # Creación de variables
```

```
    X5 = modelo5.addMVar((I, j, h), vtype=GRB.BINARY, name='X') # Restricción 10
```

```

Z5 = modelo5.addMVar((S, j, h), vtype=GRB.BINARY, name='Z') # Restricción 11
W5 = modelo5.addMVar((S, h), vtype=GRB.BINARY, name='W')

modelo5.setObjective(
    quicksum(quicksum(w[i] * X5[i][jj][hh] for i in range(I) for jj in range(j)) / (hh + 1) for hh in range(h)),
    GRB.MAXIMIZE)

# Definición de restricciones

# Restricción 1
modelo5.addConstrs((quicksum(X5[i][jj][hh] for jj in range(j) for hh in range(h)) <= 1 for i in range(I)),
    "Rest15")

# Restricción 2
modelo5.addConstrs(
    (quicksum(X5[i][jj][hh] for jj in range(j) for hh in range(h) if hh <= (rd[i] - 1)) == 0 for i in range(I)),
    "Rest25")

# Restricción 3
modelo5.addConstrs(
    (quicksum(X5[i][jj][hh] for jj in range(j) for hh in range(h) if hh > d[i]) == 0 for i in range(I) if
    d[i] <= abs(h - 1)), "Rest35")

# Restricción 4
modelo5.addConstrs(
    (quicksum(t[i] * X5[i][jj][hh] for i in range(I)) <= r[jj][hh] for jj in range(j) for hh in range(h)), "Rest45")

# Restricción 5
modelo5.addConstrs(
    (quicksum((t[i] * X5[i][jj][hh]) for i in range(I) if gamma[i] == s for jj in range(j)) <= a[s][hh] for s in
    range(S) for hh in range(h)), "Rest55")

# Restricción 6
modelo5.addConstrs(
    (quicksum((t[i] * X5[i][jj][hh]) for i in range(I) if gamma[i] == s) <= r[jj][hh] * Z5[s][jj][hh] for s in
    range(S) for jj in range(j) for hh in range(h)), "Rest65")

# Restricción 7
modelo5.addConstrs(

```

```
(quicksum((t[i] * X5[i][jj][hh]) for i in range(I) if gamma[i] == s) >= Z5[s][jj][hh] for s in range(S) for jj
in range(j) for hh in range(h)), "Rest75")
```

```
# Restricción 8
```

```
modelo5.addConstrs(
    (X5[i][jj][hh] == 0 for i in range(I) for jj in range(j) for hh in range(h) if delta[i][jj][hh] == 0), "Rest85")
```

```
# Restricción Nueva1
```

```
modelo5.addConstrs((Z5[s][jj][hh] <= W5[s][hh] for s in range(S) for jj in range(j) for hh in range(h)),
    "Rest*5")
```

```
# Restricción Nuevo2
```

```
modelo5.addConstrs((quicksum(Z5[s][jj][hh] for jj in range(j)) >= W5[s][hh] for s in range(S) for hh in
    range(h)),
    "Rest**5")
```

```
# Restricción Nueva3
```

```
modelo5.addConstrs((quicksum(W5[s][hh] for hh in range(h)) <= 1 for s in range(S)), "Rest***5")
```

```
# Resolución
```

```
print("\n")
```

```
print("Modelo Alternativo 5 resolviéndose: J:", j, ", alpha", al, ", beta:", b, ", H:", h, ", iteración n°:", iter)
```

```
modelo5.Params.timeLimit = t_parada
```

```
modelo5.optimize()
```

```
print("Modelo Alternativo 5 resuelto: J:", j, ", alpha", al, ", beta:", b, ", H:", h, ", iteración n°:", iter)
```

```
if (modelo5.Status == 9):
```

```
    optimo = 0
```

```
    print("Tiempo excedido\n")
```

```
else:
```

```
    optimo = 1
```

```
    print("Óptimo encontrado\n")
```

```
obj = modelo5.getObjective()
```

```
return (optimo, obj.getValue(), modelo5.RunTime)
```

```
#Creación hoja excel
```

```
hoja = resultados.add_sheet("Resultados")
```

```
hoja.write(0, 0, 'J')
hoja.write(1, 0, 'alpha')
hoja.write(2, 0, 'beta')
hoja.write(3, 0, 'H')
hoja.write(4, 0, 'MB - I1')
hoja.write(7, 0, 'M1 - I1')
hoja.write(10, 0, 'M2 - I1')
hoja.write(13, 0, 'M3 - I1')
hoja.write(16, 0, 'M4 - I1')
hoja.write(19, 0, 'M5 - I1')
hoja.write(22, 0, 'MB - I2')
hoja.write(25, 0, 'M1 - I2')
hoja.write(28, 0, 'M2 - I2')
hoja.write(31, 0, 'M3 - I2')
hoja.write(34, 0, 'M4 - I2')
hoja.write(37, 0, 'M5 - I2')
cj = 1
cal = 1
cb = 1
ch = 1
```

```
for jj in J:
```

```
    hoja.write(0, cj, jj)
    cj += 16
```

```
for all in alpha:
```

```
    hoja.write(1, cal, all)
    cal += 8
```

```
for bb in beta:
```

```
    hoja.write(2, cb, bb)
    cb += 4
```

```
for hh in H:
```

```
    hoja.write(3, ch, hh)
    ch += 1
```

```
column = 0
for j in J:
    for al in alpha:
        for b in beta:
            for h in H:
                column += 1
                fila = 4
                for iter in range(iteraciones):
                    parametros = GenerarInstancia(j, al, b, h)
                    I = parametros[0]
                    S = parametros[1]
                    r = parametros[2]
                    a = parametros[3]
                    gamma = parametros[4]
                    rd = parametros[5]
                    d = parametros[6]
                    delta = parametros[7]
                    t = parametros[8]
                    w = parametros[9]
                    mp = parametros[10]
                    MTBT = parametros[11]
                    dwl = parametros[12]
                    t_parada = parametros[13]

                    #Resultados pasados a hoja excel

                    #          MODELO BASE
                    modbase = ModeloBase(I, S, r, a, gamma, rd, d, delta, t, w, t_parada)
                    # Excel
                    hoja.write(fila, column, str(modbase[0]))
                    fila += 1
                    hoja.write(fila, column, str(modbase[1]))
                    fila += 1
                    hoja.write(fila, column, str(modbase[2]))
                    fila += 1

                    #          MODELO ALTERNATIVA 1
```

```

modeloalt1 = ModeloAlt1(I, S, r, a, rd, d, delta, t, w, t_parada)
# Excel
hoja.write(fila, colum, str(modeloalt1[0]))
fila += 1
hoja.write(fila, colum, str(modeloalt1[1]))
fila += 1
hoja.write(fila, colum, str(modeloalt1[2]))
fila += 1

#           MODELO ALTERNATIVA 2
modeloalt2 = ModeloAlt2(I, S, r, a, gamma, rd, d, delta, t, w, t_parada)
# Excel
hoja.write(fila, colum, str(modeloalt2[0]))
fila += 1
hoja.write(fila, colum, str(modeloalt2[1]))
fila += 1
hoja.write(fila, colum, str(modeloalt2[2]))
fila += 1

#           MODELO ALTERNATIVA 3
modeloalt3 = ModeloAlt3(I, S, r, a, gamma, rd, d, delta, t, w, t_parada)
# Excel
hoja.write(fila, colum, str(modeloalt3[0]))
fila += 1
hoja.write(fila, colum, str(modeloalt3[1]))
fila += 1
hoja.write(fila, colum, str(modeloalt3[2]))
fila += 1

#           MODELO ALTERNATIVA 4
modeloalt4 = ModeloAlt4(I, S, r, a, gamma, rd, d, delta, t, w, t_parada)
# Excel
hoja.write(fila, colum, str(modeloalt4[0]))
fila += 1
hoja.write(fila, colum, str(modeloalt4[1]))
fila += 1
hoja.write(fila, colum, str(modeloalt4[2]))
fila += 1

#           MODELO ALTERNATIVA 5
modeloalt5 = ModeloAlt5(I, S, r, a, gamma, rd, d, delta, t, w, t_parada)

```

```
# Excel
hoja.write(fila, colum, str(modeloalt5[0]))
fila += 1
hoja.write(fila, colum, str(modeloalt5[1]))
fila += 1
hoja.write(fila, colum, str(modeloalt5[2]))
fila += 1
```

```
#Impresión hoja excel
resultados.save('resultados.xls')
```