



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA (ICAI)
MÁSTER EN INGENIERÍA INDUSTRIAL

HERRAMIENTA DE PLANIFICACIÓN DE SECUENCIAS DE TRATAMIENTO QUÍMICO SIN ESPERAS DE PIEZAS AERONAÚTICAS

Autor: Juan Nozaleda Pastor
Director: Pedro Sánchez Martín

Madrid
Julio 2016

AUTORIZACIÓN PARA LA DIGITALIZACIÓN, DEPÓSITO Y DIVULGACIÓN EN RED DE PROYECTOS FIN DE GRADO, FIN DE MÁSTER, TESIS O MEMORIAS DE BACHILLERATO

1º. Declaración de la autoría y acreditación de la misma.

El autor D. Juan Nozaleda Pastor DECLARA ser el titular de los derechos de propiedad intelectual de la obra: HERRAMIENTA DE PLANIFICACIÓN DE SECUENCIAS DE TRATAMIENTO QUÍMICO SIN ESPERAS DE PIEZAS AERONÁUTICAS, que ésta es una obra original, y que ostenta la condición de autor en el sentido que otorga la Ley de Propiedad Intelectual.

2º. Objeto y fines de la cesión.

Con el fin de dar la máxima difusión a la obra citada a través del Repositorio institucional de la Universidad, el autor **CEDE** a la Universidad Pontificia Comillas, de forma gratuita y no exclusiva, por el máximo plazo legal y con ámbito universal, los derechos de digitalización, de archivo, de reproducción, de distribución y de comunicación pública, incluido el derecho de puesta a disposición electrónica, tal y como se describen en la Ley de Propiedad Intelectual. El derecho de transformación se cede a los únicos efectos de lo dispuesto en la letra a) del apartado siguiente.

3º. Condiciones de la cesión y acceso

Sin perjuicio de la titularidad de la obra, que sigue correspondiendo a su autor, la cesión de derechos contemplada en esta licencia habilita para:

- a) Transformarla con el fin de adaptarla a cualquier tecnología que permita incorporarla a internet y hacerla accesible; incorporar metadatos para realizar el registro de la obra e incorporar “marcas de agua” o cualquier otro sistema de seguridad o de protección.
- b) Reproducirla en un soporte digital para su incorporación a una base de datos electrónica, incluyendo el derecho de reproducir y almacenar la obra en servidores, a los efectos de garantizar su seguridad, conservación y preservar el formato.
- c) Comunicarla, por defecto, a través de un archivo institucional abierto, accesible de modo libre y gratuito a través de internet.
- d) Cualquier otra forma de acceso (restringido, embargado, cerrado) deberá solicitarse expresamente y obedecer a causas justificadas.
- e) Asignar por defecto a estos trabajos una licencia Creative Commons.
- f) Asignar por defecto a estos trabajos un HANDLE (URL *persistente*).

4º. Derechos del autor.

El autor, en tanto que titular de una obra tiene derecho a:

- a) Que la Universidad identifique claramente su nombre como autor de la misma
- b) Comunicar y dar publicidad a la obra en la versión que ceda y en otras posteriores a través de cualquier medio.
- c) Solicitar la retirada de la obra del repositorio por causa justificada.
- d) Recibir notificación fehaciente de cualquier reclamación que puedan formular terceras personas en relación con la obra y, en particular, de reclamaciones relativas a los derechos de propiedad intelectual sobre ella.

5º. Deberes del autor.

El autor se compromete a:

- a) Garantizar que el compromiso que adquiere mediante el presente escrito no infringe ningún derecho de terceros, ya sean de propiedad industrial, intelectual o cualquier otro.
- b) Garantizar que el contenido de las obras no atenta contra los derechos al honor, a la intimidad y a la imagen de terceros.
- c) Asumir toda reclamación o responsabilidad, incluyendo las indemnizaciones por daños, que pudieran ejercitarse contra la Universidad por terceros que vieran infringidos sus derechos e intereses a causa de la cesión.
- d) Asumir la responsabilidad en el caso de que las instituciones fueran condenadas por infracción de derechos derivada de las obras objeto de la cesión.

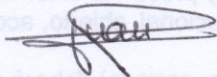
6º. Fines y funcionamiento del Repositorio Institucional.

La obra se pondrá a disposición de los usuarios para que hagan de ella un uso justo y respetuoso con los derechos del autor, según lo permitido por la legislación aplicable, y con fines de estudio, investigación, o cualquier otro fin lícito. Con dicha finalidad, la Universidad asume los siguientes deberes y se reserva las siguientes facultades:

- La Universidad informará a los usuarios del archivo sobre los usos permitidos, y no garantiza ni asume responsabilidad alguna por otras formas en que los usuarios hagan un uso posterior de las obras no conforme con la legislación vigente. El uso posterior, más allá de la copia privada, requerirá que se cite la fuente y se reconozca la autoría, que no se obtenga beneficio comercial, y que no se realicen obras derivadas.
- La Universidad no revisará el contenido de las obras, que en todo caso permanecerá bajo la responsabilidad exclusiva del autor y no estará obligada a ejercitar acciones legales en nombre del autor en el supuesto de infracciones a derechos de propiedad intelectual derivados del depósito y archivo de las obras. El autor renuncia a cualquier reclamación frente a la Universidad por las formas no ajustadas a la legislación vigente en que los usuarios hagan uso de las obras.
- La Universidad adoptará las medidas necesarias para la preservación de la obra en un futuro.
- La Universidad se reserva la facultad de retirar la obra, previa notificación al autor, en supuestos suficientemente justificados, o en caso de reclamaciones de terceros.

Madrid, a 13 de Julio de 2016

ACEPTA

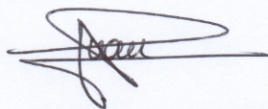


Fdo.....

Motivos para solicitar el acceso restringido, cerrado o embargado del trabajo en el Repositorio Institucional:

Declaro, bajo mi responsabilidad, que el Proyecto presentado con el título HERRAMIENTA DE PLANIFICACIÓN DE SECUENCIAS DE TRATAMIENTO QUÍMICO SIN ESPERAS DE PIEZAS AERONÁUTICAS

en la ETS de Ingeniería - ICAI de la Universidad Pontificia Comillas en el curso académico 2016/2017 es de mi autoría, original e inédito y no ha sido presentado con anterioridad a otros efectos. El Proyecto no es plagio de otro, ni total ni parcialmente y la información que ha sido tomada de otros documentos está debidamente referenciada.



Fdo.: Juan Nozaleda Pastor

Fecha: 18/07/2016

Autorizada la entrega del proyecto

EL DIRECTOR DEL PROYECTO



Fdo.: Pedro Sánchez Martín

Fecha: 18/07/2016

Vº Bº del Coordinador de Proyectos

Fdo.: Jaime de Rábago Marín

Fecha://



ESCUELA TÉCNICA SUPERIOR DE INGENIERÍA (ICAI)
MÁSTER EN INGENIERÍA INDUSTRIAL

HERRAMIENTA DE PLANIFICACIÓN DE SECUENCIAS DE TRATAMIENTO QUÍMICO SIN ESPERAS DE PIEZAS AERONAÚTICAS

Autor: Juan Nozaleda Pastor
Director: Pedro Sánchez Martín

Madrid
Julio 2016

HERRAMIENTA DE PLANIFICACIÓN DE SECUENCIAS DE TRATAMIENTO QUÍMICO SIN ESPERAS DE PIEZAS AERONÁUTICAS

Autor: Nozaleda Pastor, Juan

Director: Sánchez Martín, Pedro

Entidad Colaboradora: ICAI – Universidad Pontificia Comillas

RESUMEN DEL PROYECTO

INTRODUCCIÓN

Las características del sector aeronáutico obligan a que exista una constante reposición de piezas como consecuencia de la incesante exposición de estas a las condiciones extremas a las que están expuestas. Estas condiciones hacen que sea necesario aplicar determinados tratamientos a las piezas con el objetivo de aportar a las piezas ciertas características que no poseen por sí solas.

Cada pieza recibe un tratamiento distinto según su tipo, de forma que dichos tratamientos consisten en baños químicos en cubas dispuestas en línea, existiendo un único brazo transportador encargado de trasladar las piezas de una cuba a otra. Cabe destacar que cada cuba solo admite una pieza al mismo tiempo, sin embargo existen algunas cubas equivalentes capaces de aplicar el mismo tratamiento de forma que su uso puede habilitarse o no, según decida el usuario.

Como en cualquier fábrica, lo ideal es tratar el mayor número de piezas posible en el menor tiempo siempre y cuando se mantengan los niveles de calidad deseados y para ello es necesario secuenciar las tareas mediante una optimización. Pero la principal característica de este problema que lo diferencia de los problemas clásicos de tipo *job shop* es que no se permiten esperas. Es decir, cada pieza tiene que estar un mínimo de tiempo dentro de las cubas sin sobrepasar el límite que se fije. Ello hace que la optimización de la secuencia requiera mucha carga computacional y el programa de optimización tarde días en encontrar el óptimo, cuando lo que se necesita es disponer de la secuencia óptima al inicio de cada jornada laboral.

De modo que el objetivo de este proyecto es desarrollar una herramienta computacional que sea más ágil y aporte resultados suficientemente buenos (que no se alejen demasiado del óptimo). Para ello se ha realizado una búsqueda

de los métodos heurísticos que contribuyan a tal efecto mediante la asignación de prioridades de ciertas piezas. La forma de priorizar unas piezas frente a otras se ha realizado fijando las variables adecuadas de precedencia en el programa de optimización que se ha utilizado (GAMS).

METODOLOGÍA

Antes de comenzar con el desarrollo de los métodos heurísticos se ha realizado un análisis del comportamiento de la secuenciación, para así aplicar más adelante las mejoras propuestas en la creación de los heurísticos.

Este análisis se ha llevado a cabo mediante la optimización de 39 secuencias partiendo de los datos de 3 piezas proporcionadas para ello. Entre las herramientas utilizadas que han contribuido al análisis, cabe destacar el uso del lenguaje de programación Python tanto para crear los documentos de texto en los que se fijan las variables de precedencia, como para generar los diagramas de Gantt que ayudan a la visualización de las secuencias resultantes.

Las conclusiones extraídas de este primer análisis se han utilizado para proponer las mejoras en la resolución del problema, esto es, se han desarrollado 20 heurísticos basándose en los resultados del análisis. Para comprobar la eficacia de estos heurísticos y medir su robustez, se han creado 10 casos partiendo de los datos de 6 trabajos, de forma que se llevaron a cabo las optimizaciones de los 10 casos o escenarios sin ninguna asignación de prioridad, con el objetivo de poder medir cuánto empeoran los casos cuando se aplican los heurísticos.

Los criterios utilizados para la creación de los heurísticos son:

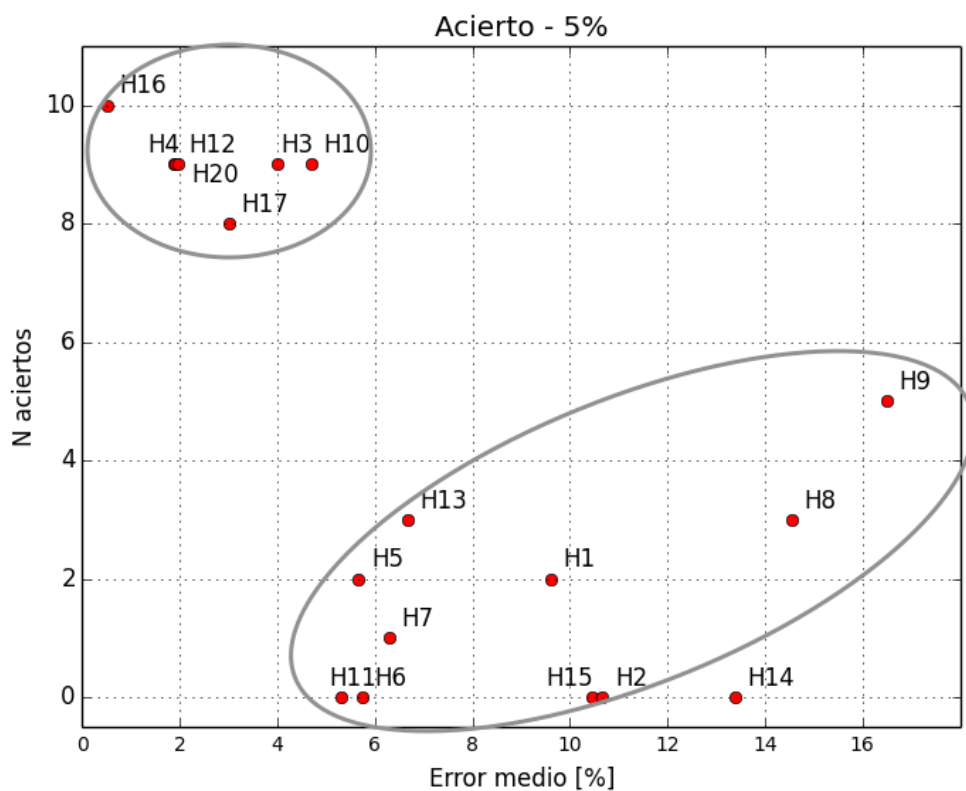
- Influencia de la duración media agregada sobre el total de las piezas. H1 y H2
- Influencia de la no-exclusividad de las piezas. H3, H4, H5 y H6
- Influencia de la exclusividad de las piezas. H7, H8, H9 y H10
- Influencia de la duración media agregada sobre la mitad de las piezas. H11 y H12
- Influencia de la amplitud agregada sobre el total de las piezas. H13 y H14
- Influencia de la aleatoriedad. H15
- Influencia de la repetitividad de las piezas. H16 y H17
- Influencia de la asignación de cubas equivalentes. H18 y H19
- Influencia de la priorización entre las piezas del mismo tipo. H20

RESULTADOS

Debido a la naturaleza del proyecto, no es posible representar los resultados en una única gráfica ya que la medida de empeoramiento es subjetiva. Por ello, a continuación se muestran dos gráficas que resumen los resultados. La primera se centra en el error medido en tanto por ciento respecto al óptimo, mientras que la segunda usa el criterio de desviación en minutos respecto al óptimo.

En las gráficas, en el eje vertical se representa el número de aciertos de cada heurístico (máximo 10), es decir, número de casos en los que se ha obtenido una desviación inferior a la que se marca en el título de cada gráfica; y en el eje horizontal se representa la media que cada heurístico ha obtenido en los 10 casos.

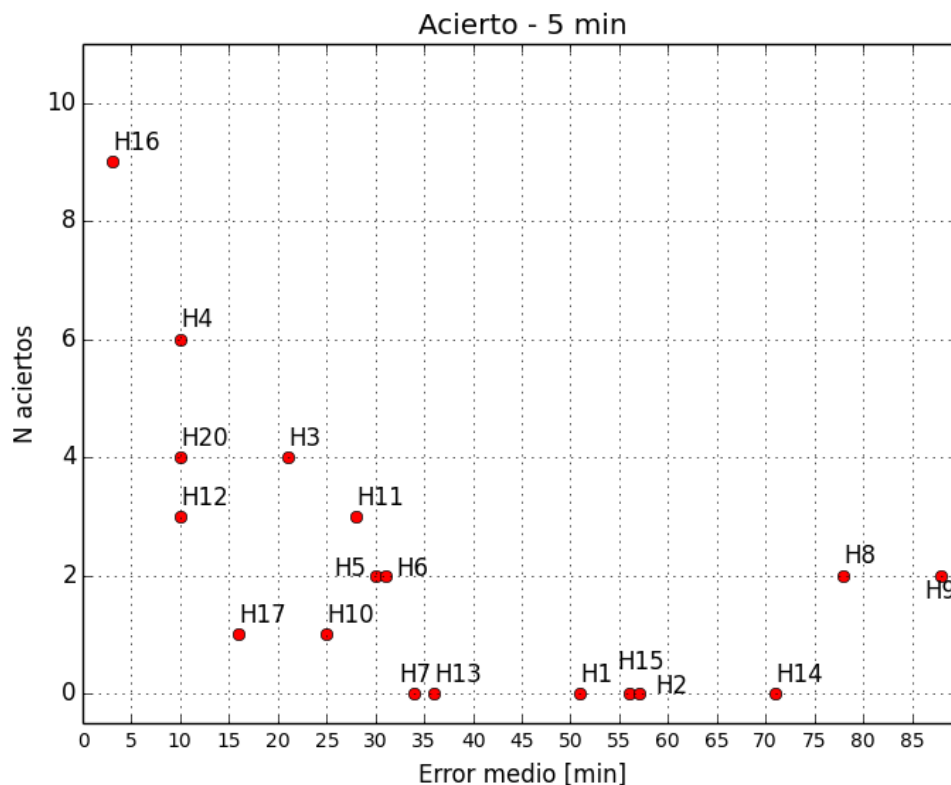
Observando ambas gráficas se ve claramente la diferencia a la hora de marcar el criterio de aceptación de heurísticos, pues mientras que en la gráfica 1 se pueden observar dos núcleos de resultados: uno aceptable y otro muy malo (menos aciertos y mayor error), en la gráfica 2 los resultados siguen una tendencia que aísla más los mejores heurísticos.



Gráfica 1: Representación gráfica de los resultados con error límite 5%

Analizando los heurísticos que pertenecen al *cluster* superior izquierdo, es decir, el que contiene los heurísticos con los mejores resultados, se podría hacer una primera distinción entre los heurísticos con buenos resultados y malos. De modo que H3, H4, H10, H12, H16, H17 y H20 serían los mejores heurísticos.

La decisión de representar los resultados bajo el criterio de tiempo se debe a que el tiempo es el recurso más importante de este problema y permite una mejor comprensión. Por ejemplo, con el heurístico H10 se obtienen buenos resultados si se tiene en cuenta que obtiene 9 aciertos con un error inferior a tan solo el 5%; sin embargo, si se observa la gráfica 2, se puede ver que solamente obtiene 4 aciertos con una desviación aceptable de 5 minutos.



Gráfica 2: Representación gráfica de los resultados con tiempo límite 5 min

CONCLUSIONES

De los heurísticos analizados, el que mejores resultados aporta es H16, que consiste en una ordenación de las tareas en orden creciente bajo el criterio de la duración agregada de las tareas con una asignación de prioridades basada en la repetitividad de las piezas en cada secuencia. Es decir, a las familias de piezas que solo aparecen 1 o 2 veces en la secuencia, no se les asigna ninguna prioridad; mientras que entre las que aparecen 3 o más veces, se asigna una

prioridad entre los distintos tipos de piezas basada en el tiempo medio agregado en orden creciente.

Es preciso encontrar un compromiso entre el número de piezas a las que asignar prioridades porque cuando se asignan prioridades a demasiadas piezas los resultados empeoran; y de igual forma, cuando no se asignan prioridades a suficientes piezas, existen casos en los que no se alcanzan soluciones. Es importante recalcar que dependiendo de los requerimientos de cada fábrica, pueden considerarse como aceptables determinados heurísticos.

Se puede reducir la carga computacional en gran medida mediante la asignación de prioridades que a priori puede parecer que carecen de utilidad, como se ve en el caso de H20, pues los resultados están entre los mejores.

Los la robustez de los heurísticos analizados no se puede extrapolar al caso en el que se habilitan las cubas equivalentes.

PLANNING TOOL OF SEQUENCES FOR CHEMICAL TREATMENT OF AERONAUTICAL COMPONENTS UNDER NO-WAIT CONSTRAINTS

INTRODUCTION

The characteristics of the aviation industry require the existence of a constant replacement of components as a result of the ceaseless exposure of them, and the extreme conditions to which they are exposed. These conditions make it necessary to apply certain treatments to the components with the aim of making the components acquire characteristics they do not have by themselves.

Each component is treated differently according to their type, so that these treatments consist of chemical baths in barrels arranged in line, having a single carrier arm in charge of moving the pieces from one barrel to another. Note that each barrel supports only one piece at the same time, however there are some equivalent barrels able to apply the same treatment so that its use can be enabled or not, as decided by the user.

As in any factory, the ideal technique is to try as many pieces as possible in the shortest time possible as long as the desired quality standards are maintained, so this requires sequencing tasks through optimization. But the main feature of this issue that differentiates it from the classic problems of job shop type is that waiting is not allowed. That is, each piece must stay a minimum of time inside the barrels without exceeding the fixed limit. This means the sequence optimization requires a lot of computational load so it takes days for the optimization software to find the optimum, while it is necessary to have the optimal sequence at the beginning of each workday.

So the aim of this project is to develop a computational tool that is more agile and able to provide sufficiently good results (not stray too far from the optimal value). For this, a search of heuristic methods has been done that contribute to this effect by the prioritization of certain components. The way to prioritize some components over the rest has been performed by setting the right values of the precedence variable in the optimization program that has been used (GAMS).

METHODOLOGY

Before starting with the development of heuristic methods, an analysis of sequencing behaviour has been conducted in order to apply the proposed improvements later when creating heuristic methods.

This analysis was carried out by optimizing 39 sequences from the data of 3 components. Among the tools used that have contributed to the analysis, the use of Python programming language is included, both for creating text documents in which the variables of precedence are written, and for generating Gantt charts that help the visualization of the resulting sequences.

The conclusions drawn from this preliminary analysis have been used to propose improvements in the resolution of the problem, that is, 20 heuristics have been developed based on the results of the analysis. To check the effectiveness of these heuristics and measure their robustness, 10 cases have been created from the data of 6 components so that optimizations of 10 cases or scenarios without any priority assignment were carried out, with the aim of measuring how far the result is from the optimum value.

The criteria used to create the heuristics are:

- Influence of the aggregated average duration over all the components. H1 y H2
- Influence of the non-exclusive components. H3, H4, H5 y H6
- Influence of the exclusive components. H7, H8, H9 y H10
- Influence of the aggregated average duration over half of the components. H11 y H12
- Influence of the aggregated width time over all the components. H13 y H14
- Influence of randomness. H15
- Influence of the repetitiveness of the components. H16 y H17
- Influence of the designation of equivalent barrels. H18 y H19
- Influence of the prioritisation among the components of the same type. H20

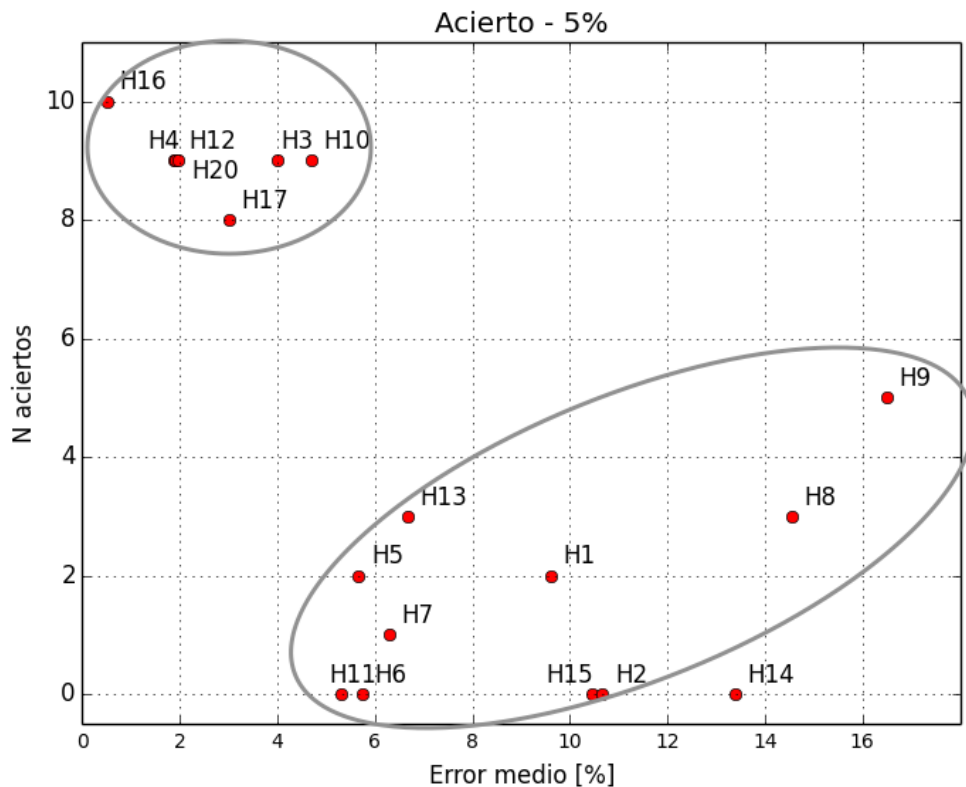
RESULTS

Due to the nature of this project, it is not possible to represent the results in a single graph as the worsening measurement is subjective. Therefore, two graphs that summarize the results are shown. The first focuses on the error measured as percent relative to the optimum, while the second uses the criterion of deviation from the optimal measured in minutes.

In the graphs, the vertical axis represents the number of hits of each heuristic (maximum 10), i.e., the number of cases in which it is obtained a deviation lower

than marked on the title of each graph; and on the horizontal axis it is represented the average that each heuristic obtained along the 10 cases is represented.

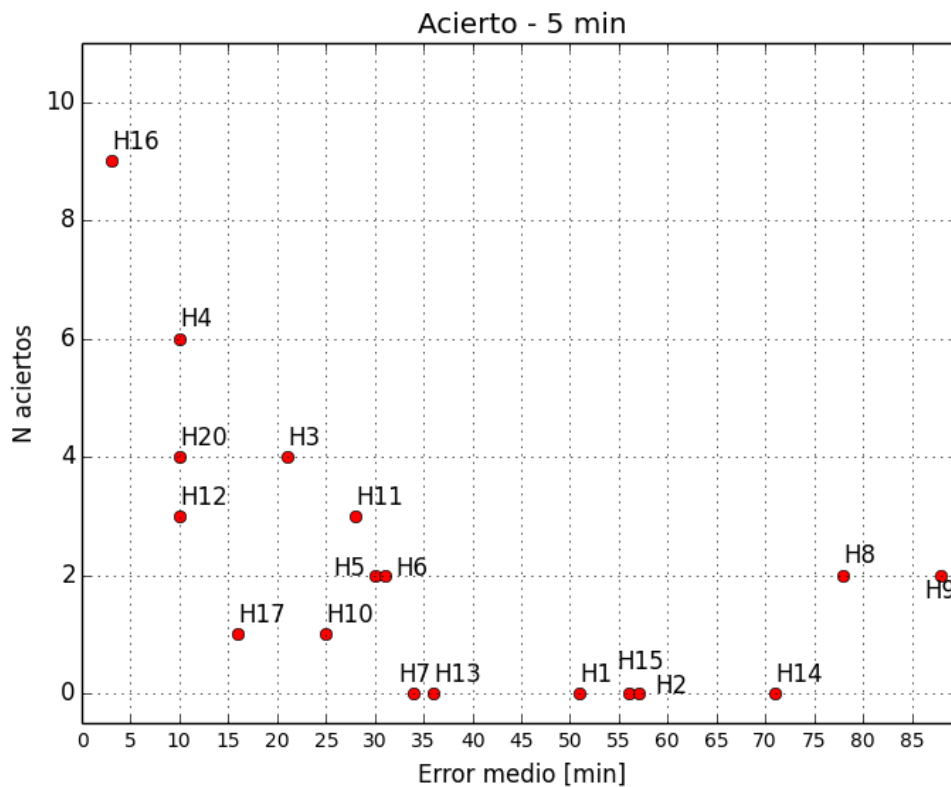
Watching both graphics, the difference when choosing the criteria of acceptance of the heuristics can be seen, because while in the graph 1 shows two cores results: one acceptable and one very bad (fewer hits and more error) in graph 2 results follow a trend that isolates the best heuristics.



Graph 1: Graphic representation of the results with limit error of 5%

Analysing the heuristics that belong to the upper left cluster, i.e. containing the heuristic with the best results, a first distinction between heuristic with good and bad results could be made. So H3, H4, H10, H12, H16, H17 and H20 would be the best heuristic.

The decision to represent the results under the test of time is because time is the most important resource of this problem and allows a better understanding. For example, in the heuristic H10 good results are obtained if one considers that gets nine hits with just an error of 5%; however, as the graph 2 shows, it can be seen that it only gets 4 hits with an acceptable deviation of 5 minutes.



Graph 2: Graphic representation of the results with limit time of 5 min

CONCLUSIONS

From the heuristics analysed, the one that brings best results is H16, which consists of an array of tasks in ascending order under the criteria of the aggregated average duration of tasks with a prioritization based on the repetitiveness of the pieces in each sequence. That is, families of components that appear only 1 or 2 times in the sequence, are not assigned any priority; while among those that appear three or more times a priority is assigned within the different types of components based on the aggregated average duration in increasing order.

It is necessary to find a compromise in the number of components to prioritize because when priorities are assigned to too many components, results worsen; and similarly, when priorities are not assigned to enough components, there are cases where no solutions are reached. It is also important to mention that, depending on the requirements of each plant, some heuristics can be regarded as acceptable while others are not.

The computational load can be highly reduced by the assignation of priorities that a priori may seem lacking utility, as seen in H20, where its results are among the best.

The robustness of heuristics analysed can not be extrapolated to the case in which the equivalent barrels are enabled.

ÍNDICE

ÍNDICE	1
1 INTRODUCCIÓN	3
1.1 Ámbito del proyecto	3
1.2 Motivación.....	3
1.3 Objetivo.....	5
1.4 Taxonomía general del problema	5
1.5 Estado del arte.....	8
1.6 Estructura del documento	11
2 MODELADO DEL PROBLEMA.....	13
2.1 Introducción	13
2.2 Formulación genérica	14
2.3 Formulación matemática	17
2.3.1 Notación.....	17
2.3.2 Función objetivo.....	18
2.3.3 Restricciones genéricas.....	19
3 ANÁLISIS DEL COMPORTAMIENTO DE LA SECUENCIACIÓN	23
3.1 INTRODUCCIÓN.....	23
3.2 GAMS	24
3.3 DIAGRAMAS DE GANTT	26
3.4 VARIABLES.....	28
3.5 HEURÍSTICOS – ESTADO DEL ARTE.....	30
3.6 TIEMPOS DE PROCESADO.....	31
3.7 PUNTOS DE PARTIDA	32
3.8 SECUENCIACIONES	34
3.9 CONCLUSIONES	46
4 MEJORAS EN LA RESOLUCIÓN DEL PROBLEMA.....	47
4.1 INTRODUCCIÓN.....	47
4.2 CASOS ORIGINALES	47
4.3 HEURISTICOS	52
4.3.1 H1 y H2. Duración agregada sobre el total de las piezas.....	54
4.3.2 H3, H4, H5 y H6 No exclusividad de las tareas	55
4.3.3 H7, H8, H9 y H10. Exclusividad de las tareas	58
4.3.4 H11 y H12. Duración agregada sobre la mitad de las piezas	61
4.3.5 H13 y H14. Amplitud agregada sobre el total de las piezas	62
4.3.6 H15. Aleatoriedad	64
4.3.7 H16 y H17. Repetitividad de las piezas	64

4.3.8	H18 y H19. Cubas equivalentes	66
4.3.9	H20. Priorización entre las piezas del mismo tipo	67
5	RESULTADOS Y CONCLUSIONES.....	69
5.1	Orden creciente vs decreciente con criterio de tiempo agregado	69
5.1.1	H1 vs H2. Total de las piezas	69
5.1.2	H3 vs H5. No exclusividad	70
5.1.3	H4 vs H6. No exclusividad menos restrictiva	71
5.1.4	H7 vs H9. Exclusividad	71
5.1.5	H8 vs H10. Exclusividad más restrictiva	72
5.1.6	H11 vs H12. Duración agregada sobre la mitad de piezas.....	73
5.1.7	H13 vs H14. Amplitud agregada	73
5.1.8	H16 vs H17. Repetitividad de las piezas	73
5.1.9	Conclusiones	74
5.2	Exclusividad vs no exclusividad.....	75
5.2.1	H3 vs H7. Orden creciente.....	75
5.2.2	H4 vs H8. Orden creciente con distintas restricciones	75
5.2.3	H5 vs H9. Orden decreciente.....	76
5.2.4	H6 vs H10. Orden decreciente con distintas prioridades.....	76
5.2.5	Análisis.....	76
5.2.6	Conclusiones	78
5.3	Duración agregada vs amplitud	78
5.3.1	H1 vs H13. Orden creciente.....	78
5.3.2	H2 vs H14. Orden decreciente.....	78
5.3.3	Conclusiones	79
5.4	Cubas equivalentes	79
5.4.1	Conclusiones	80
5.5	Priorización entre piezas del mismo tipo	80
5.6	Resumen resultados.....	80
6	FUTUROS DESARROLLOS.....	87
7	BIBLIOGRAFÍA	89

1 INTRODUCCIÓN

1.1 Ámbito del proyecto

Las características del sector aeronáutico obligan a que exista una constante reposición de piezas, esto es consecuencia de la incesante exposición de éstas a las condiciones extremas y de diferente naturaleza que sufren a lo largo de su vida útil. Estas condiciones extremas obligan a que dichas piezas sean ligeras a la vez que cumplan ciertos requisitos de resistencia, ya sea mecánica, térmica o incluso eléctrica; siendo necesario que se les apliquen determinados tratamientos de recubrimiento, como el anodizado o el desengrase alcalino entre otros, de forma que les confieran características indispensables que no poseen por sí solas (tales como la dureza superficial o la resistencia a la corrosión). [1]

En definitiva, el ámbito del proyecto es extensible a la industria química o a cualquier otro tipo de industria en la que las tareas estén acotadas en el tiempo, es decir, que haya una correlatividad de operaciones con tiempos limitados tanto de espera como de transporte.

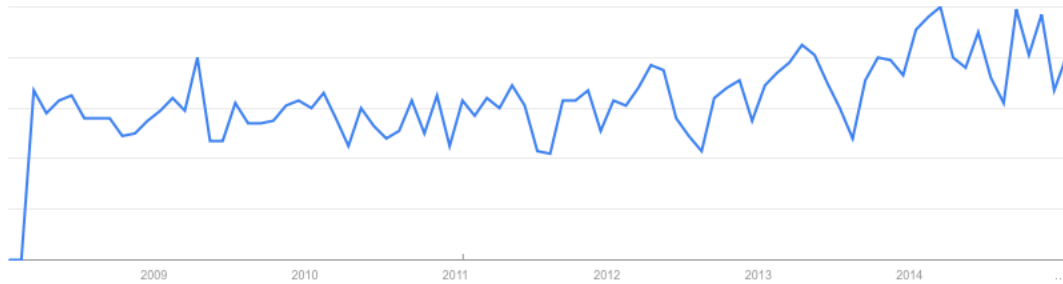
Una empresa que se dedique al tratamiento de estas piezas recibirá cada día un conjunto distinto de piezas y necesitará conocer aquella secuencia óptima que permita procesarlas en el menor tiempo posible, ya que al fin y al cabo, la jornada laboral es limitada y un incremento de la duración del procesado se traduce en una inversión mayor en personal y por lo tanto en costes.

Por ello el objetivo del presente proyecto consiste en el desarrollo de una herramienta que permita una planificación eficiente de las secuencias de tratamiento químico de piezas aeronáuticas teniendo en cuenta las limitaciones de recursos y tiempo existentes, las cuales se explicarán más adelante.

1.2 Motivación

La época actual se caracteriza por la globalización, una de cuyas ventajas en el mundo empresarial hace que aumente la competitividad y por lo tanto la calidad del producto o servicio que se ofrece. Sin embargo, la globalización también es responsable de que una empresa situada en las antípodas pueda llegar a acaparar el mercado de la empresa en cuestión si ésta no se adecúa a las nuevas tecnologías.

Lo anterior conduce a las empresas a seguir las tendencias que les ayuden a reducir costes, siendo *Lean Manufacturing* una de esas tendencias que está ganando peso hoy en día en el mundo y en concreto en España, como se muestra en la siguiente gráfica obtenida de *Google Trends* y que muestra la evolución búsquedas del término “*Lean Manufacturing*” en España.



Gráfica 1: Tendencias de la búsqueda “*Lean manufacturing*”

La filosofía de esta tendencia reside en la eliminación o minimización de desperdicios, es decir, de todo aquello que no aporte nada en la cadena de valor del producto. Con esto se consigue una reducción de los tiempos de fabricación, para así poder satisfacer la demanda empleando el mínimo stock posible. Sin embargo, antiguamente los gestores se centraban en maximizar la utilización de las máquinas, pues las factorías se diseñaban para mantener las máquinas funcionando a su máxima capacidad el máximo tiempo posible. Desde el punto de vista de la máquina, eso es eficiente, pero desde el punto de vista de la productividad de la fábrica entera, no lo es siempre. Por esta razón adquieren gran importancia las herramientas que ayudan a determinar una programación de secuencia más eficiente.

La razón de la elaboración de este proyecto consiste en que la tecnología de la que se dispone hoy en día, no permite la resolución de problemas de optimización complejos en un tiempo adecuado, ya el tiempo de búsqueda del óptimo puede alargarse hasta durar varios días haciendo que sea complicado secuenciar este tipo de problemas; donde lo que se pretende es aumentar la capacidad del procesado siempre y cuando se mantengan los niveles de calidad que se requieren. Pues si el procesado la pieza no cumpliera los tiempos acotados el resultado sería la fabricación de unas piezas con peores prestaciones, de forma que los clientes dejarían de acudir a la fábrica en cuestión.

Por ello, en este proyecto se va a tratar de buscar los patrones que conduzcan a un punto de partida en el cual que se alcance una solución próxima a la óptima,

siempre y cuando se alcance en un tiempo considerablemente menor; o que proporcionen una solución mejor que la que se ha obtenido en el mismo tiempo.

1.3 Objetivo

Hoy en día los medios para resolver los problemas de optimización aún presentan ciertos problemas en cuanto al tiempo de cálculo a pesar de haber avanzado mucho en los últimos años. En concreto, el programa utilizado en este proyecto emplea varios días en encontrar la secuencia óptima que reduzca al máximo el *makespan*, entendido como el tiempo que transcurre entre que empieza la primera tarea en la fábrica y termina la última.

El objetivo de este proyecto no consiste en modelar el problema de secuenciación, porque se parte con el problema ya modelado. Sino que el objetivo es crear una herramienta computacional que sea más ágil y aporte resultados suficientemente buenos, modificando el modelado y la metodología de resolución para que el programa sea más eficaz a la hora de resolver problemas de gran tamaño y en un tiempo razonable. Es decir, que en lugar de tardar días en alcanzar el óptimo, se pueda llegar a un resultado relativamente bueno en mucho menos tiempo; pues hay que tener en cuenta que cada día la fábrica recibe un conjunto distinto de piezas a tratar.

Para cumplir con el objetivo del proyecto se va a aplicar una combinación de las técnicas existentes que se comentan en el estado del arte, pues la bibliografía que aborda este tipo de problemas puede aportar reglas de aproximación a un punto de partida que contribuya a alcanzar el objetivo.

1.4 Taxonomía general del problema

Desde que se comenzó a desarrollar la teoría de toma de decisiones tras la Segunda Guerra Mundial, hoy en día, en líneas generales se pueden distinguir hasta cuatro tipos de procesos productivos, los cuales vienen representados en la siguiente imagen.

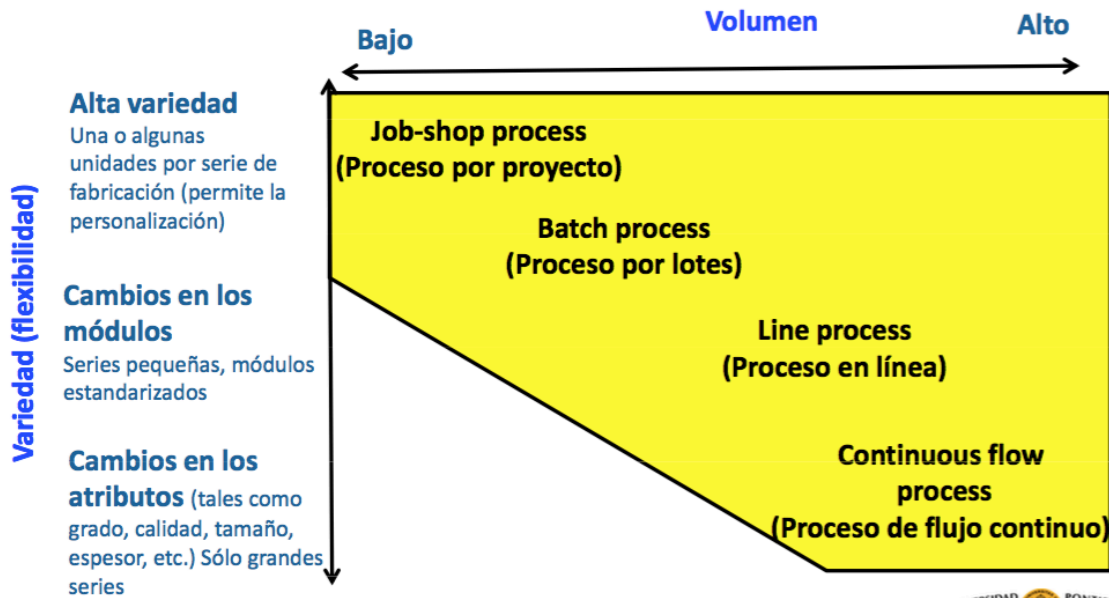


Imagen 1: Tipos de procesos productivos en la actualidad

Como se puede observar en la imagen, los criterios de distinción entre los cuatro procesos se basan en dos aspectos que van de la mano: el volumen de producción y la variedad del producto. Pues un proceso que elabore productos a medida, necesariamente va a tener un volumen mucho más reducido que otro proceso productivo que elabore un único producto.

Los cuatro procesos mencionados pueden quedar resumidos en los tres siguientes [2]

- *Job-Shop* (de ahora en adelante JS)
- Proceso por lotes o *batch process*
- Proceso continuo

De modo que el proceso productivo al que nos enfrentamos pertenece a la familia de *JS process*, cuyas características principales son las siguientes:

- Poca cantidad de producto pero mucha variedad
- Número elevado de instrucciones de trabajo (pues existen distintos tipos de piezas y cada una requiere un tratamiento diferente)
- Equipamiento de propósito general

Por *JS scheduling* se denomina a la asignación de recursos durante un periodo de tiempo determinado para ejecutar una colección de tareas. La secuenciación de tareas tipo JS tiene gran importancia debido a que permite dos cosas: por un

lado reducir los costes de forma notable, y por otro, aumentar en gran medida la producción; consiguiendo así un aumento del beneficio.[3]

Existen distintas estrategias a la hora de resolver este tipo de problemas:

- Métodos heurísticos: constituyen la base del desarrollo de la teoría de secuenciación.
- Algoritmos numéricos: intentan buscar soluciones exactas pero dejaron de usarse durante un tiempo debido a la imposibilidad de alcanzar soluciones factibles. La principal estrategia de este tipo es *Branch and Bound* (BB), que representa a modo de árbol las posibles secuencias factibles; de forma que proporciona unas reglas para ir descartando ciertas secuencias. El inconveniente es que tiene una alta carga computacional.
- Métodos de aproximación: surgen como alternativa a los métodos numéricos. Los primeros algoritmos consisten en reglas de prioridad de despacho (*Priority Dispatch Rules*), que asignan prioridades a las operaciones de forma que la secuencia se genera según dichas prioridades. Sin embargo, de entre todas las técnicas que constituyen este tipo de método, no existe una que sea claramente superior al resto.
- Técnicas de optimización: basadas en métodos de búsqueda local, a este tipo de técnicas pertenecen las redes neuronales y los algoritmos genéticos entre otras.

Sin embargo, el problema de este proyecto no es de tipo JS puro, pues la naturaleza del proceso impide que una pieza pueda esperar durante un tiempo indefinido antes de ser tratada en la siguiente cuba, como también sucede en la industria metalúrgica o la química [4]. Por tanto, se está ante un problema del tipo *no wait job shop* (de ahora en adelante NWJS).

Además, en este proyecto se da la existencia de máquinas (cubas) equivalentes, que se explicarán más adelante y no se permite la simultaneidad, es decir, no se pueden procesar dos piezas a la vez en una misma cuba. Por otro lado, el problema se caracteriza por la existencia de m máquinas o cubas y n trabajos o piezas; siendo una de las principales características la existencia del brazo transportador, que se modela como restricciones de tiempo y constituye un cuello de botella.

1.5 Estado del arte

La secuenciación de tareas en los procesos de tipo JS es complicada debido a la alta variabilidad de productos y al inevitable bajo nivel de producción, lo cual ha llevado a la elaboración de numerosos estudios durante el siglo XX.

Como se indica más adelante, la revisión del estado del arte ha sido uno de los aspectos más importantes de este proyecto porque ha permitido proponer heurísticos que sin dicha revisión no se hubiesen considerado.

A pesar de que este problema es de tipo NWJS, se revisa el estado del arte del problema JS clásico ya que: por un lado se han realizado muy pocos estudios en cuanto a este tema debido a su complejidad; y por otro, se considera que es de gran importancia conocer las características del problema básico del que se parte. [5]

Los problemas de decisión se pueden clasificar en dos grupos según el tipo de complejidad de la resolución, N y NP. Los problemas de tipo N son aquellos que pueden resolverse en un tiempo polinómico, mientras que los de tipo NP son los que no pueden resolverse en un tiempo polinómico. Como los problemas de JS *scheduling* pertenecen al tipo NP-completo, son los más complicados de formular y resolver, razón por la que durante más de 35 años se ha intentado abordar el problema de distintas formas (unas con más éxito que otras), incluyendo programación matemática, redes neuronales o algoritmos genéticos entre otros. [3]

En la literatura actual se define el problema de JS como un conjunto de n jobs o trabajos $\{J_1, J_2, \dots, J_n\}$ que han de ser procesados a través de m máquinas $\{M_1, M_2, \dots, M_m\}$ de forma que cada trabajo requiere una secuencia de tratamiento a través de las distintas máquinas. El procesado sobre un *job* J_i por parte de una máquina M_j se denomina operación o_{ij} , asumiendo que cada máquina puede ejecutar una única operación al mismo tiempo.

Cada operación o_{ij} requiere un tiempo de procesado distinto y está acotado entre un máximo y un mínimo que depende de cada operación. El momento en el que el *job* J_i está disponible para ser procesado se denomina *ready time*, R_i . [6, 7]

El problema general consiste en encontrar una secuencia que:

- Sea compatible con las restricciones tecnológicas.

- Sea óptima en relación a la medida de desempeño.

En el caso de NWJS existen un tipo de restricciones distintas a las de JS denominadas *no-wait constraint* y que limitan el problema. Por ejemplo, una restricción por la que una operación (B) de una pieza debe empezar como mínimo un tiempo fijo después de que acabe la operación anterior (A) de la misma pieza (esto se debe a la existencia del brazo transportador). Esta es la principal diferencia frente al JS, que de forma matemática y a modo de ejemplo, viene representada por las siguientes ecuaciones:

$$TF_{i1} \leq TI_{i2} \quad \forall i \text{ para el caso de JS}$$

$$TF_{i1} + s = TI_{i2} \quad \forall i \text{ para el caso de NWJS}$$

Donde:

- TF_{i1} representa el tiempo en el que finaliza la tarea i en la máquina 1, tarea A.
- TI_{i2} representa el tiempo en el que se inicia la tarea i en la máquina 2, tarea B.
- s representa el tiempo de traslado entre cubas realizado por una máquina u operario.

En este proyecto, los trabajos o *jobs* son las distintas piezas a las que se realiza el tratamiento; las operaciones son cada uno de los baños que recibe cada una de las piezas; y las máquinas son las cubas en las que se realizan los baños químicos. Además, el tiempo fijo que marca la diferencia entre el final de una operación y el comienzo de otra viene representado por el tiempo que el brazo tarda en trasladar una pieza de una cuba a otra.

Existe bibliografía que aborda el tema con sencillos ejemplos, como es el caso de [6], en donde se plantea el problema de secuenciación como un problema de distribución de periódicos entre cuatro compañeros de piso. Donde además se realiza una clasificación de los problemas a través de una simple denotación formada por cuatro parámetros, $n/m/A/B$. Donde:

- n representa el número de Jobs, por ejemplo, n (indefinido).
- m representa el número de máquinas, por ejemplo, 2.
- A representa el flujo de trabajo, por ejemplo, continuo.

- B representa la medida de desempeño o rendimiento, por ejemplo, *makespan*.

En [6] también se proporciona una lista de medidas de desempeño existentes, de las cuales la más utilizada en la bibliografía revisada es la minimización del ya mencionado *makespan*.

Otros documentos como [5] consideran problemas de secuenciación con restricciones de tiempo limitado desde el punto de vista de los 3 tipos de flujo de trabajo (JS, por lotes y continuo). En concreto, en cuanto al NWJS, estudia el problema partiendo de dos modelos de resolución que seguidamente compara.

Por otro lado, [8] aborda el estudio del NWJS haciendo una división del problema en dos subproblemas y asumiendo que si se es capaz de resolverlos, se llegará a la solución óptima del problema principal. Estos dos subproblemas son:

- *Sequencing problem*: se basa en la búsqueda de aquella secuencia de trabajos que proporcione una mejor oportunidad para encontrar una solución de mayor calidad, dando prioridad a aquellos trabajos que limiten el problema.
- *Timetabling problem*: se basa en la búsqueda de una programación factible a partir de las prioridades que proporciona el *sequencing problem*.

Además, el objetivo de ese estudio es analizar cuál de los dos subproblemas influye más a la hora de encontrar el óptimo, para lo cual combina tres métodos de secuenciación con los mejores métodos de *timetabling* que existen en la literatura. Asimismo, en [8] se hace una revisión bibliográfica del NWJS (donde como se ha mencionado antes, se apunta a la escasez de literatura frente al JS clásico). Todo ello con el objetivo de minimizar el *makespan*, cosa que como se ha indicado anteriormente, sucede en el resto de la bibliografía analizada hasta el momento.

En [3] se lleva a cabo un estudio acerca de las corrientes más importantes que han guiado a los ingenieros en la resolución del problema de *Job Shop* además de las tendencias más recientes. En cuanto a la corriente de las técnicas matemáticas, se apunta a la ya mencionada técnica de descomposición del problema general en distintos subproblemas.

1.6 Estructura del documento

En el capítulo 2 se comentan las principales características del problema que desarrolla el presente proyecto y se introducen y explican las restricciones que modelan el problema.

En el capítulo 3 se explican las herramientas usadas a lo largo del proyecto tanto para el análisis del comportamiento de la secuenciación como para las mejoras en la resolución del problema. Además se introduce el análisis del comportamiento, de donde se van a extraer las primeras conclusiones que servirán de base para el desarrollo del capítulo 4.

En el capítulo 4 se aplican las conclusiones extraídas del anterior epígrafe para desarrollar los heurísticos propuestos en el proyecto. También se explica el modo de proceder para evaluar la robustez de dichos heurísticos.

En el capítulo 5 se comentan los resultados y las conclusiones obtenidas del anterior epígrafe comparando los heurísticos a través de distintos criterios y se indican cuáles son los dos mejores heurísticos.

En el capítulo 6 se proponen posibles avances que amplían el contenido del proyecto.

2 MODELADO DEL PROBLEMA

2.1 Introducción

El problema que se describe en este proyecto se representa en la siguiente imagen, de forma que las cubas en las que tienen que ser tratadas las piezas, se disponen en línea una detrás de otra.

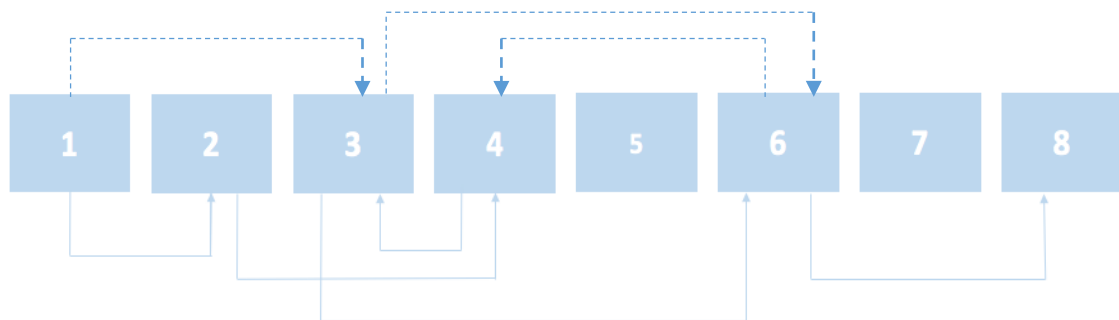


Imagen 2: Representación gráfica de la disposición de la planta

Existe un brazo transportador que es el encargado de coger la pieza que está a punto de comenzar el procesado, de llevarla de una cuba a otra, y por último de sacarla de la línea de procesado. Hay que tener en cuenta que la velocidad a la que se desplaza el brazo depende de si lleva una pieza o está libre; ya que si lleva una pieza, lógicamente va moverse a una velocidad inferior que si estuviese libre. Además, también es necesario comentar que el brazo debe esperar un tiempo breve cada vez que saca una pieza de la cuba porque tiene que dejar que esta gotee.

Dada una pieza, el brazo tiene que llevarla por todas las cubas necesarias que aparecen en los datos de la pieza, como se muestra en la imagen 2, donde se representa un ejemplo del flujo de tareas de una pieza. Esto teniendo en cuenta que para cada tarea de la pieza, ésta tiene que estar un mínimo de tiempo en cada cuba y no puede sobrepasar el máximo asignado. Tanto el mínimo como el máximo son datos propios de las tareas de las piezas.

Existen ciertas cubas equivalentes, es decir, cubas duplicadas que permiten su uso simultáneo durante el procesado con el objetivo de reducir el *makespan*. El uso de estas cubas se puede habilitar o deshabilitar a gusto del usuario.

Durante el procesado de una pieza, se aprovechan los tiempos que ésta está dentro de las cubas para comenzar el procesado de la siguiente pieza, siempre

y cuando no se violen las restricciones, como se ve representado en la imagen 2.

A continuación se describe la función objetivo así como las restricciones más relevantes a la hora de plantear la formulación del modelado del problema.

2.2 Formulación genérica

Función objetivo

Minimizar el tiempo total de secuenciación o *makespan* (pero esta vez entendido como momento en el tiempo en el que termina la última pieza).

Restricciones genéricas

- **No solapamiento de cubas:** Dada una cuba, no es posible procesar dos trabajos al mismo tiempo en la misma cuba.
- **Consecutividad de las tareas:** Dada una pieza, la secuencia de las tareas que requiere dicha pieza es un dato fijo que se introduce en el modelo y no se puede alterar.
- **Brazo transportador:**
 - Sólo existe un brazo transportador capaz de mover las piezas.
 - Debido al peso de la pieza, los tiempos de desplazamiento del brazo transportador cuando lleva carga son mayores que cuando no la lleva.
 - El brazo sólo es capaz de mover una pieza al mismo tiempo.
- **Tiempo final de cada tarea:** El tiempo final de cada tarea tiene que ser igual al tiempo de inicio de esa tarea más la duración de la misma.
- **Duración de las tareas:** Están acotadas entre un máximo y un mínimo, de forma que son datos proporcionados en el enunciado y el programa las asigna durante la búsqueda del óptimo.
- **Separación de tareas consecutivas en la misma pieza:** El tiempo final de una tarea que precede a otra tarea de la misma pieza, tiene que estar separado el tiempo de traslado del brazo transportador con carga (pues el tiempo de traslado con y sin carga es diferente), que es dato proporcionado e independiente de la distancia a recorrer.

En la siguiente imagen se muestra el zoom del diagrama de Gantt de una de las optimizaciones realizadas a lo largo del proyecto. En ella se puede observar cómo existe un tiempo entre cada tarea consecutiva de la misma pieza, de forma que para las piezas rojas se ha marcado con dos líneas verticales entre las cubas 3 y 5 para ayudar a la visualización.

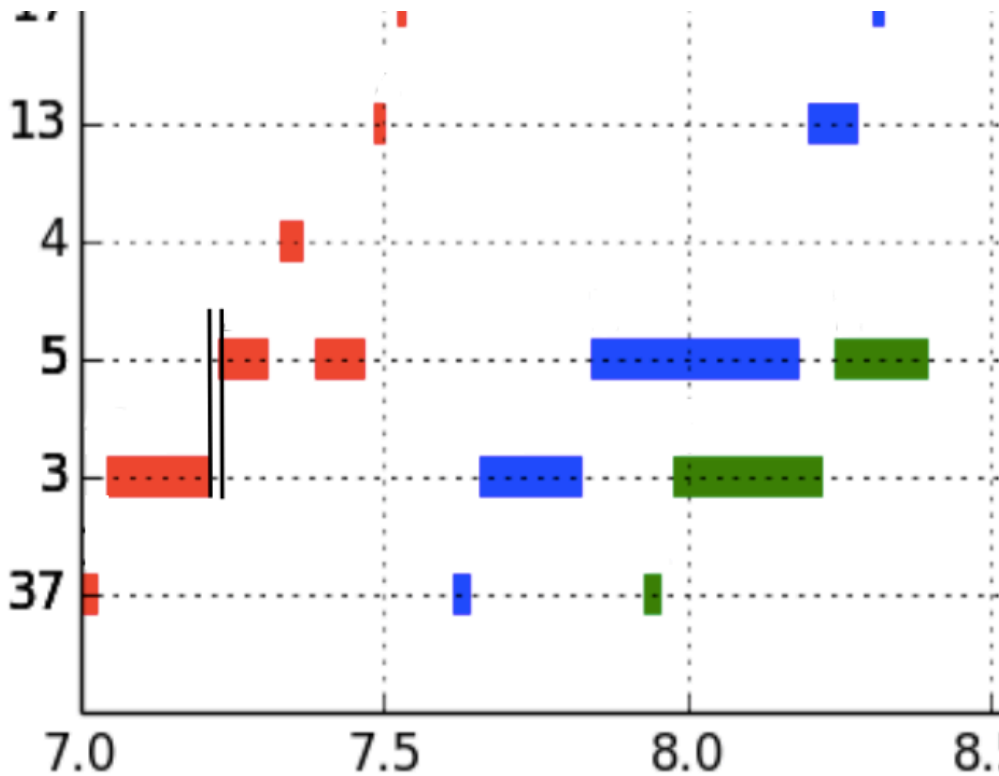


Imagen 3: Zoom del diagrama de Gantt de una optimización realizada a lo largo del proyecto

- **Precedencia entre el final y el inicio de tareas pertenecientes a distintas piezas:** Esta sigue la línea de la anterior restricción, pero se aplica en tareas de distintas piezas. Es decir, cuando se introduce una pieza en una cuba, tiene que transcurrir un tiempo hasta que el brazo pueda sacar otra pieza de otra cuba. Este tiempo es el tiempo que tarda el brazo transportador en ir a por la segunda pieza (sin carga).

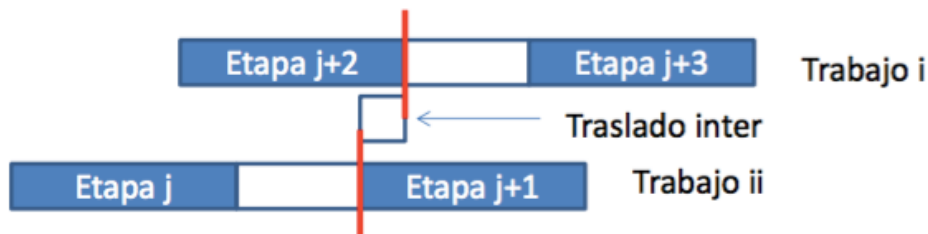


Imagen 4: Representación gráfica de la restricción de tiempo entre el final y el inicio de tareas pertenecientes a distintas piezas

- **Precedencia entre los tiempos de inicio de tareas pertenecientes a distintas piezas:** Desde que se introduce una pieza en una cuba hasta que introduce otra pieza en otra cuba, como mínimo tiene que transcurrir un tiempo igual al del traslado del brazo sin carga (para ir a por la segunda pieza) más el tiempo de traslado del brazo con carga (llevando a la segunda pieza).

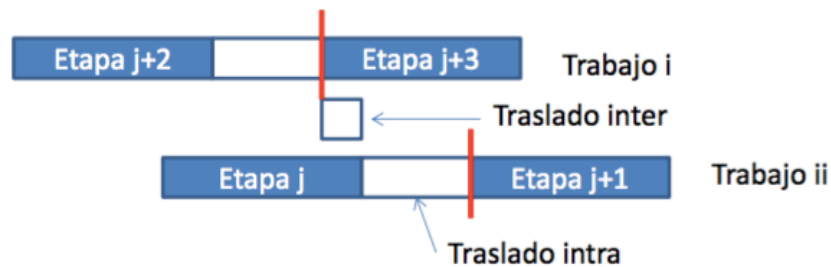


Imagen 5: Representación gráfica de la restricción de tiempo entre los inicio de tareas pertenecientes a distintas piezas

- **Precedencia entre los tiempos finales de tareas pertenecientes a distintas piezas:** Esta restricción es similar a la anterior pero aplicada a los finales de tareas. Es decir, desde que termina una tarea hasta que termina otra, como mínimo tiene que transcurrir un tiempo igual a la suma del tiempo que tarda el brazo en llevar la primera pieza a la siguiente cuba (traslado con carga) más el tiempo que tarda en ir a por la segunda pieza para sacarla de la cuba (traslado sin carga).

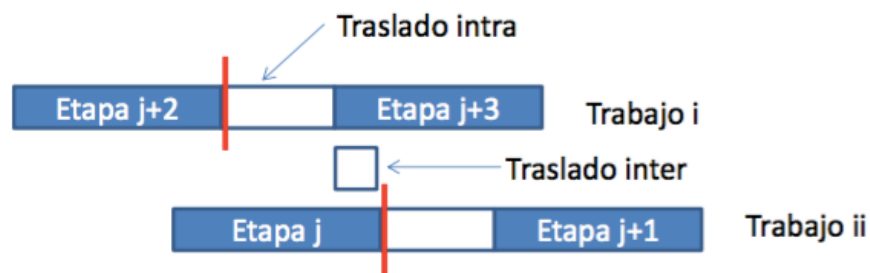


Imagen 6: Representación gráfica de la restricción de tiempo entre los finales de tareas pertenecientes a distintas piezas

- **Precedencia y separación entre tareas de distintas piezas en la misma cuba:** Cuando dos tareas de distintas piezas emplean la misma cuba, como mínimo tiene que transcurrir un tiempo igual a la suma del tiempo que tarda el brazo en llevar la primera pieza a la siguiente cuba (traslado con carga) más el tiempo que tarda el brazo en ir a por la segunda pieza (traslado sin carga) más el tiempo que tarda el brazo en llevar la segunda pieza a la cuba en cuestión (traslado con carga).

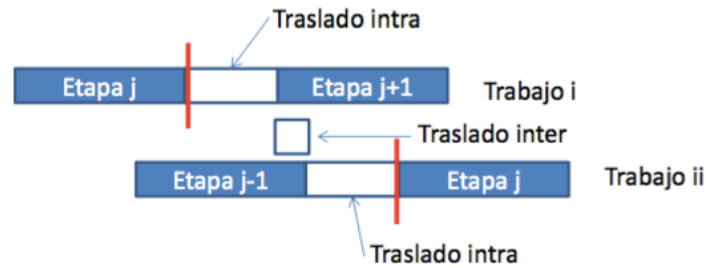


Imagen 7: Representación gráfica de la restricción de tiempo entre distintas piezas en la misma cuba

- **Precedencia por cuba k:** Entre dos tareas que usan la misma cuba, ya pertenezcan a la misma pieza o no, hay que definir cual de las dos va antes.
- **Precedencia entre el inicio de una tarea y el final de una tarea de piezas distintas:** Entre el inicio de una tarea y el final de otra, hay que definir cual de las dos va antes.
- **Precedencia entre el inicio de dos tareas de piezas distintas:** Entre el tiempo de inicio de una tarea y el tiempo de inicio de otra tarea es necesario definir cual de las dos va antes.
- **Precedencia entre el final de dos tareas de piezas distintas:** Entre el tiempo final de una tarea y el tiempo final de otra tarea hay que definir cual de las dos es precedente.
- **Asignación a cubas equivalentes:** La tarea que pueda asignarse a dos cubas que sean equivalentes tiene que optar sólo por una de ellas.

2.3 Formulación matemática

A continuación se presenta la formulación matemática de las restricciones y parámetros que se han introducido en el modelado del problema.

2.3.1 Notación

- La tupla (i,j) hace referencia a la tarea j de la pieza i.
- $vTiempoTotal$: variable que representa el tiempo total de la secuencia.
- $vTiempoFinal(i,j)$: variable que representa el tiempo en el que acaba el procesado de la tarea j de la pieza i.
- $vTiempoInicio(i,j)$: variable que representa el tiempo en el que inicia el procesado de la tarea j de la pieza i.

- $vTiempoProceso(i,j)$: variable que representa el tiempo de procesado de la tarea j de la pieza i .
- $vIndPrecedGlob1(ii,jj,i,j)$: variable binaria que indica si existe precedencia entre el inicio de la tarea jj de la pieza ii y el final de la tarea j de la pieza i .
- $vIndPrecedGlob2(i,j,ii,jj)$: variable binaria que indica si existe precedencia entre el inicio de la tarea j de la pieza i y el inicio de la tarea jj de la pieza ii .
- $vIndPrecedGlob3(i,j,ii,jj)$: variable binaria que indica si existe precedencia entre el final de la tarea j de la pieza i y el final de la tarea jj de la pieza ii .
- $vPrecedencia(i,j,ii,jj,k)$: variable binaria que indica si existe precedencia entre el final de la tarea j de la pieza i y el inicio de la tarea jj de la pieza ii en la asignación de las tareas a la cuba k .
- $vAsignaCuba(i,j,k)$: variable binaria que indica si la tarea j de la pieza i se asigna a la cuba k o no.
- $pTiempoTrasladoIntra(i,j)$: parámetro que representa el tiempo de traslado de la pieza i a la tarea j .
- $pTiempoTrasladoInter$: parámetro que representa el tiempo de traslado del brazo transportador sin carga.
- $pMinDurac(i,j)$: parámetro que representa la cota inferior de la duración del procesado de la tarea j de la pieza i .
- $pMaxDurac(i,j)$: parámetro que representa la cota superior de la duración del procesado de la tarea j de la pieza i .
- $pBigM$: Se trata de un valor elevado.

2.3.2 Función objetivo

Como ya se ha mencionado, la función objetivo consiste en reducir el tiempo total de secuenciación o *makespan*, de forma que el tiempo total va a ser mayor o igual que el mayor de los tiempos finales de todas las tareas.

$$vTiempoTotal \geq vTiempoFinal(i,j)$$

2.3.3 Restricciones genéricas

Consecutividad de las tareas

$$vTiempoFinal(i, j) + pTiempoTrasladoIntra(i, j) = vTiempoInicio(i, jj)$$

El tiempo de inicio de una tarea se calcula como la suma del tiempo final de la tarea anterior de la misma pieza más el tiempo de traslado.

Tiempo final de cada tarea

$$vTiempoFinal(i, j) = vTiempoInicio(i, j) + vTiempoProceso(i, j)$$

$$pMinDurac(i, j) \leq vTiempoProceso(i, j) \leq pMaxDurac(i, j)$$

El tiempo final de una tarea es igual a la suma del tiempo de inicio de la misma tarea más el tiempo de procesado por un lado; y por otro el tiempo de procesado de una tarea tiene que estar entre el mínimo y el máximo.

Precedencia entre el tiempo final y el tiempo de inicio de tareas pertenecientes a distintas piezas

$$\begin{aligned} vTiempoFinal(i, j) - vTiempoInicio(ii, jj) \\ \geq pTiempoTrasladoInter - pBigM * [1 \\ - vIndPrecedGlob1(ii, jj, i, j)] \end{aligned}$$

Esta ecuación expresa que el tiempo que transcurre entre el final de la tarea j de la pieza i y el inicio de la tarea jj de la pieza ii, tiene que ser como mínimo igual al tiempo de traslado del brazo sin carga; siempre y cuando el inicio de la tarea jj de la pieza ii tenga lugar antes que el final de la tarea jj de la pieza ii.

Por ejemplo, si el inicio de la tarea 3 de la pieza A_2 se produce antes que el final de la tarea 2 de la pieza A_1, $vIndPrecedGlob1=1$; de forma que quedaría:

$$vTiempoFinal(A_1,2) - vTiempoInicio(A_2,3) \geq pTiempoTrasladoInter - 0$$

Si por el contrario, el final de la tarea 2 de la pieza A_1 se produce antes que el inicio de la tarea 3 de la pieza A_2, $vIndPrecedGlob1=0$; de forma que quedaría:

$$vTiempoFinal(A_1,2) - vTiempoInicio(A_2,3) \geq pTiempoTrasladoInter - 1000$$

De esta forma la restricción se relajaría porque el lado izquierdo de la ecuación va a ser positivo o negativo, pero en cualquier caso siempre va a ser mayor que el lado derecho, porque va a ser un número negativo muy grande en valor absoluto comparado con el lado izquierdo.

Precedencia entre los tiempos de inicio de tareas pertenecientes a distintas piezas

$$\begin{aligned}
 &vTiempoInicio(i, j) + pTiempoTrasladoInter \\
 &\quad + pTiempoTrasladoIntra(ii, jj - 1) \\
 &\leq pTiempoInicio(ii, jj) + pBigM * [1 \\
 &\quad - vIndPrecedGlob2(i, j, ii, jj)]
 \end{aligned}$$

El tiempo que transcurre entre el inicio de la tarea j de la pieza i y el inicio de la tarea jj de la pieza ii tiene que ser como mínimo igual al tiempo de traslado del brazo sin carga (para ir a por la segunda) más el tiempo de traslado con carga (para llevar la segunda); siempre y cuando el inicio de la tarea j de la pieza i se produzca antes que el inicio de la tarea jj de la pieza ii.

Precedencia entre los tiempos finales de tareas pertenecientes a distintas piezas

$$\begin{aligned}
 &vTiempoFinal(i, j) + pTiempoTrasladoInter + pTiempoTrasladoIntra(i, j) \\
 &\leq pTiempoFinal(ii, jj) + pBigM * [1 \\
 &\quad - vIndPrecedGlob3(i, j, ii, jj)]
 \end{aligned}$$

El tiempo que transcurre entre el final de la tarea j de la pieza i y el final de la tarea jj de la pieza ii tiene que ser como mínimo igual al tiempo de traslado del brazo con carga (para que pueda llevar la pieza i a la siguiente tarea) más el tiempo de traslado del brazo sin carga (para que pueda ir a sacar la pieza ii); siempre y cuando el final de la tarea j de la pieza i se produzca antes que el final de la tarea jj de la pieza ii.

Precedencia y separación entre tareas de distintas piezas en la misma cuba

$$\begin{aligned}
 &vTiempoFinal(i, j) + pTiempoTrasladoInter + pTiempoTrasladoIntra(i, j) \\
 &\quad + pTiempoTrasladoIntra(ii, jj - 1) \\
 &\leq pTiempoInicio(ii, jj) + pBigM \\
 &\quad * [1 - vIndPrecedencia(i, j, ii, jj, k)] + 0.5 \\
 &\quad * [pBigM[2 - vAsignaCuba(i, j, k) - vAsignaCuba(ii, jj, k)]]
 \end{aligned}$$

El tiempo que transcurre entre el final de la tarea j de la pieza i y el inicio de la tarea jj de la pieza ii tiene que ser como mínimo igual al tiempo de traslado del brazo con carga (para llevar la pieza i a la siguiente tarea) más el tiempo de traslado del brazo sin carga (para ir a por la pieza ii) más el tiempo de traslado del brazo con carga (para llevar la pieza ii); siempre y cuando el final de la tarea j de la pieza i se produzca antes que el inicio de la tarea jj de la pieza ii en la cuba k y tanto la tarea j de la pieza i como la tarea jj de la pieza ii se realicen en la cuba k.

Precedencia por cuba k

$$vIndPrecedencia(i, j, ii, jj, k) + vIndPrecedencia(ii, jj, i, j, k) = 1$$

Dado que solamente una de las dos tareas puede preceder a la otra, la suma de las variables binarias tiene que ser 1.

Precedencia entre el inicio de una tarea y el final de una tarea de piezas distintas

$$vIndPrecedenciaGolb1(i, j, ii, jj) + vIndPrecedenciaGolb1(ii, jj, i, j) = 1$$

De igual forma que en la anterior restricción, solo puede darse una precedencia y ambas variables binarias solamente pueden sumar 1.

Precedencia entre el inicio de dos tareas de piezas distintas

$$vIndPrecedenciaGolb2(i, j, ii, jj) + vIndPrecedenciaGolb2(ii, jj, i, j) = 1$$

Análogamente a las anteriores restricciones, solo se permite una precedencia de forma que la suma de ambas variables binarias solo puede ser 1.

Precedencia entre el final de dos tareas de piezas distintas

$$vIndPrecedenciaGolb3(i, j, ii, jj) + vIndPrecedenciaGolb3(ii, jj, i, j) = 1$$

Una vez más, esta restricción sigue la línea de las 3 anteriores restricciones permitiendo solamente una precedencia y reflejándose en que la suma de ambas variables binarias tiene que ser 1.

Asignación de cubas equivalentes

$$vAsignaCuba(i, j, k) + vAsignaCuba(i, j, kk) = 1$$

A diferencia de las anteriores restricciones, ésta no es una restricción de precedencia en el tiempo sino que dada una tarea, a ésta se le asigna una cuba en caso de que pudiera realizarse en dos cubas equivalentes.

3 ANÁLISIS DEL COMPORTAMIENTO DE LA SECUENCIACIÓN

3.1 INTRODUCCIÓN

En este apartado se va a realizar un análisis previo del comportamiento de las secuencias partiendo de los datos proporcionados de tres piezas distintas para tal caso. En concreto dicho análisis se ha llevado a cabo realizando 39 optimizaciones, cuyas características pueden verse reflejadas de forma resumida en la tabla 1. La razón de que se realizasen tantas secuencias, entre otras cosas, es que como punto de partida se quiso realizar un análisis del comportamiento de la secuenciación.

La metodología seguida en este proyecto para realizar el análisis del impacto de los diferentes heurísticos que se proponen, se basa en la asignación de prioridades de precedencia en las distintas cubas a través de la modificación de los parámetros disponibles en la herramienta GAMS, que permiten fijar los valores de determinadas variables usadas en el modelado del problema.

La mayoría de las optimizaciones que se llevaron a cabo fueron sin asignación de cubas equivalentes, ya que la asignación de dichas cubas puede alterar los resultados de los heurísticos, como se verá más adelante.

Nombre	Limite tiempo	Tareas
Secuenciacion20160125	2 minutos	2 piezas, limpieza y anodizado
Secuenciacion20160126	2 minutos	3 piezas, una de cada
Secuenciacion20160126-2	2 minutos	3 piezas, una de cada
Secuenciacion20160126-3	2 minutos	3 piezas, una de cada
Secuenciacion20160126-4	2 minutos	3 piezas, una de cada
Secuenciacion20160128	2 minutos	3 piezas, una de cada
Secuenciacion20160128-2	2 minutos	3 piezas, una de cada
Secuenciacion20160129	2 minutos	3 piezas, una de cada
Secuenciacion20160129-2	2 minutos	3 piezas, una de cada
Secuenciacion20160202	2 minutos	6 piezas, 2 de cada
Secuenciacion20160202-2	2 minutos	6 piezas, 2 de cada
Secuenciacion20160204	2 minutos	6 piezas, 2 de cada
Secuenciacion20160204-2	2 minutos	6 piezas, 2 de cada
Secuenciacion20160204-3	2 minutos	
Secuenciacion20160204-4	5 minutos	6 piezas, 3 de anod y limp
Secuenciacion20160204-5	sin limite	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160205	sin limite	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160208	sin limite	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160208-2	sin limite	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160208-3	sin limite	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160208-4	20 minutos	7 piezas, 3 anod y lim; 1 desen
Secuenciacion20160209	20 minutos	7 piezas, 3 anod y lim; 1 desen
Secuenciacion20160209-2	20 minutos	7 piezas, 3 anod y lim; 1 desen
Secuenciacion20160209-3	20 minutos	7 piezas, 3 anod y lim; 1 desen
Secuenciacion20160209-4	20 minutos	7 piezas, 3 anod y lim; 1 desen
Secuenciacion20160209-5	1 hora	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160209-6	1 hora	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160210	6 horas	Todas
Secuenciacion20160210-2	1 hora	Todas
Secuenciacion20160210-3	1 hora	Todas
Secuenciacion20160616	1 hora	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160616-2	1 hora	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160616-3	1 hora	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160616-4	1 hora	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160616-5	1 hora	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160617	1 hora	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160617-2	1 hora	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160617-3	1 hora	8 piezas, 3 anod y lim; 2 desen
Secuenciacion20160617-4	1 hora	8 piezas, 3 anod y lim; 2 desen

Tabla 1: Resumen de las optimizaciones del caso único

A continuación se presentan las principales herramientas empleadas que han posibilitado la realización tanto del análisis de la secuenciación como de las mejoras en la resolución (capítulo 4).

3.2 GAMS

Para la realización de este proyecto se ha hecho uso de la herramienta informática de optimización GAMS, creada en 1987 en EEUU y que emplea un lenguaje algebraico de modelado con el mismo nombre que la herramienta (*General Algebraic Modelling System*). [9]

El programa recibe los datos de todas las piezas involucradas en la secuencia, de forma que durante la optimización, el programa va explorando y mostrando en pantalla numerosos nodos y estableciendo una cota superior y otra inferior así como el *gap* que hay entre ambos valores, entre otros parámetros. A continuación se muestra una captura de pantalla de una optimización cualquiera.

Nodes		Current Node			Objective Bounds			Work	
Expl	Unexpl	Obj	Depth	IntInf	Incumbent	BestBd	Gap	It/Node	Time
0	0	1.95000	0	1375	18.05000	1.95000	89.2%	-	11s
0	0	2.85833	0	1615	18.05000	2.85833	84.2%	-	51s
0	0	2.85833	0	1512	18.05000	2.85833	84.2%	-	86s
0	0	2.85833	0	775	18.05000	2.85833	84.2%	-	106s
0	0	2.85833	0	1013	18.05000	2.85833	84.2%	-	118s
0	0	2.85833	0	1180	18.05000	2.85833	84.2%	-	132s
0	0	2.85833	0	996	18.05000	2.85833	84.2%	-	144s
0	0	2.85833	0	414	18.05000	2.85833	84.2%	-	196s
H	0				17.5500000	2.85833	83.7%	-	570s
0	0	2.85833	0	851	17.55000	2.85833	83.7%	-	621s
0	0	2.85833	0	675	17.55000	2.85833	83.7%	-	863s
0	0	2.85833	0	713	17.55000	2.85833	83.7%	-	882s
0	0	2.85833	0	662	17.55000	2.85833	83.7%	-	896s
0	0	2.85833	0	649	17.55000	2.85833	83.7%	-	908s
0	0	2.85833	0	709	17.55000	2.85833	83.7%	-	927s
0	0	2.85833	0	757	17.55000	2.85833	83.7%	-	949s
0	0	2.85833	0	746	17.55000	2.85833	83.7%	-	1006s
0	2	2.85833	0	746	17.55000	2.85833	83.7%	-	2341s
1	2	2.85833	1	741	17.55000	2.85833	83.7%	3596	2376s
2	3	2.85833	1	758	17.55000	2.85833	83.7%	2175	2420s
3	4	2.85833	2	659	17.55000	2.85833	83.7%	1573	2436s
4	4	2.85833	3	626	17.55000	2.85833	83.7%	1570	2460s
5	5	2.85833	3	591	17.55000	2.85833	83.7%	1400	2469s

Imagen 8: Captura de pantalla de una optimización de GAMS

Las dos columnas encuadradas en la imagen representan de izquierda a derecha, las cotas superior e inferior. Tanto el valor final obtenido de la optimización como las cotas de cada nodo, representan la duración (en horas) del procesado de la secuencia en la fábrica. Es decir, si el valor es de 17.5, significa que para poder procesar todas las piezas del problema, la fábrica debería estar funcionando 17 horas y media. Como se puede observar en esta captura de pantalla, a lo largo de la optimización el límite superior va bajando y el límite inferior va subiendo de forma que el óptimo se alcanzaría cuando ambos valores coincidiesen.

No obstante, GAMS da la posibilidad de fijar un límite de tiempo para cada optimización de forma que en caso de alcanzar el límite de tiempo fijado, el resultado será el que marca el límite superior.

Por otro lado, entre los archivos de salida del programa se encuentran los tiempos de inicio y final de cada tarea, que permiten su representación gráfica.

3.3 DIAGRAMAS DE GANTT

En este proyecto es muy importante la visualización de los datos, ya que si no se pudiese observar cómo son las secuencias resultantes tanto de los casos originales como de los heurísticos, el análisis y la comparación de los resultados serían muy complicados. De modo que la forma elegida para representar todos los datos mencionados que genera GAMS es a través de diagramas de Gantt.

Los diagramas de Gantt son una herramienta gráfica desarrollada por Henry Laurence Gantt que sirven para representar actividades ordenadas en forma de barras horizontales a lo largo de una secuencia de tiempo concreta. En la siguiente imagen se puede ver un ejemplo.

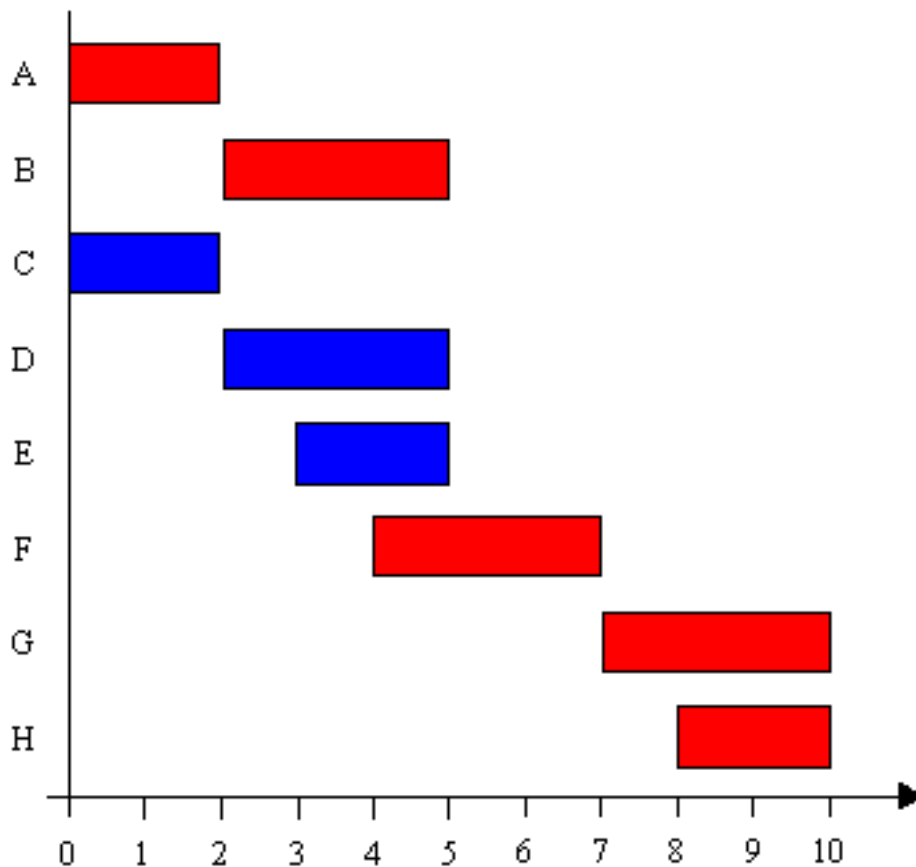


Imagen 9: Ejemplo de diagrama de Gantt. [10]

Para desarrollar dichos diagramas a partir de los archivos generados por GAMS se han seguido dos formas distintas. La primera fue a través de Excel, aprovechando los archivos de salida o outputs que genera GAMS en cada optimización, se copiaban los resultados a una hoja de cálculo sobre la que hacer el diagrama, modificando los diferentes parámetros de Excel que permiten

construir el diagrama. En la siguiente imagen se puede ver un ejemplo de diagrama de Gantt a partir de Excel.

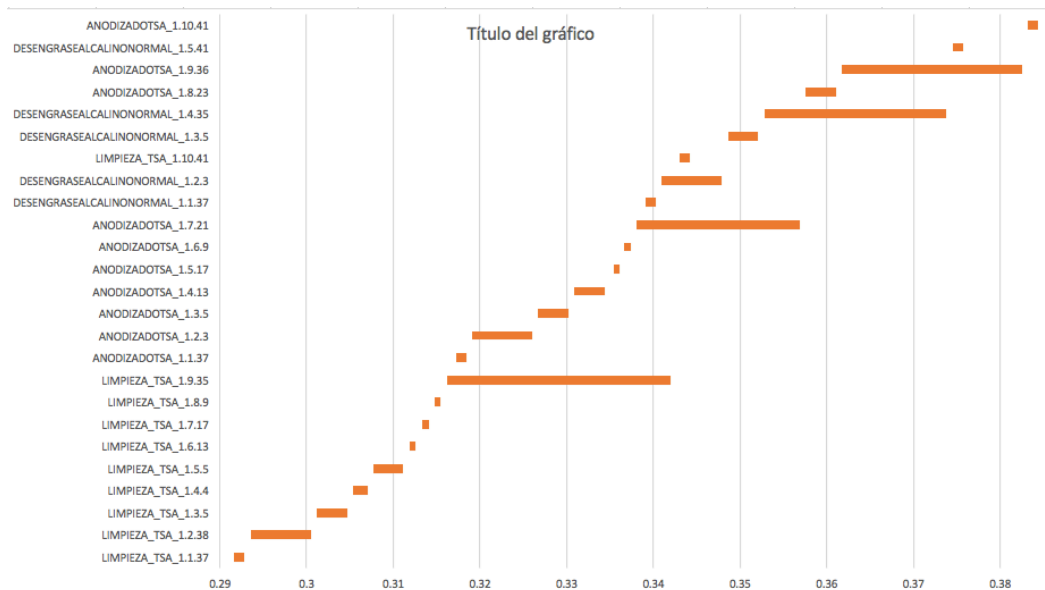


Imagen 10: Ejemplo de diagrama de Gantt generado a partir de Excel

Esta forma se descartó enseguida ya que el proceso de generar el diagrama requería un tiempo muy elevado además de que visualmente no estaba muy completo, de modo que se tenía que buscar un método alternativo.

Así pues, se aplicaron los conocimientos en el lenguaje Python para generar los diagramas de Gantt. Los resultados obtenidos no tienen nada que ver en cuanto al tiempo empleado, ya que usando Python es prácticamente automático y tan sólo hay que preparar los datos en documentos de texto y ejecutar el script. A continuación se muestra un ejemplo de diagrama de Gantt a partir de Python.

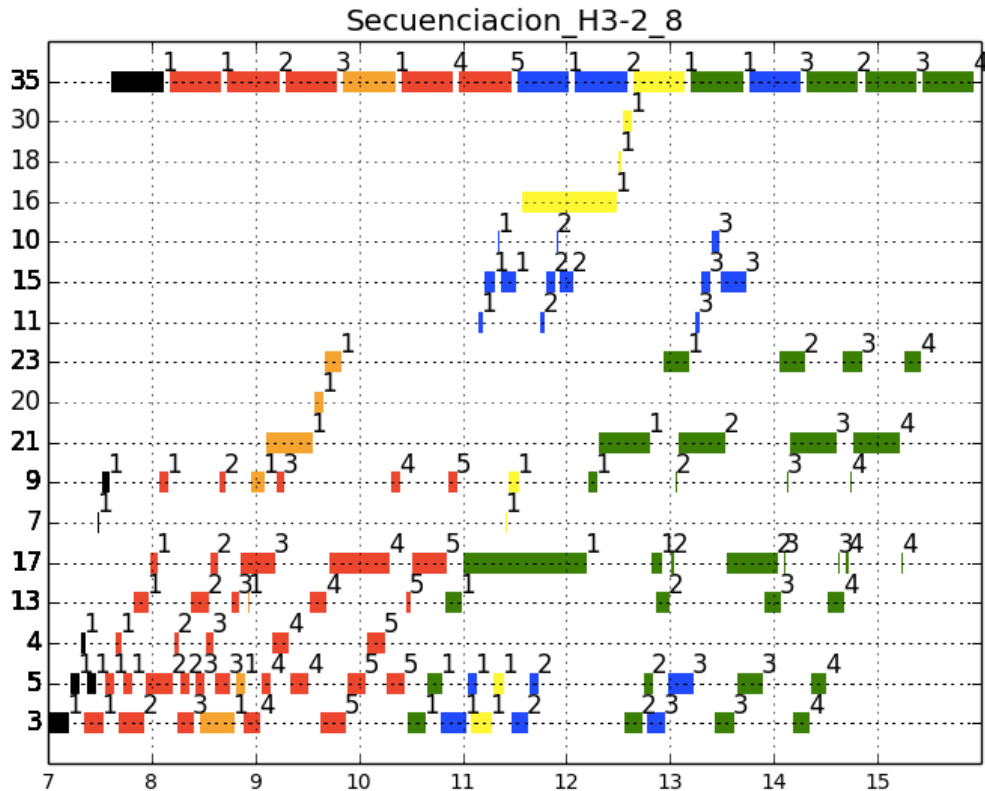


Imagen 11: Ejemplo de diagrama de Gantt generado a partir de Python

En el diagrama se representa una secuencia al azar generada a lo largo del proyecto. El eje horizontal representa el tiempo, de forma que a las 7 de la mañana es el momento en el que comenzaría el procesado de las piezas y terminaría, en este caso, un poco antes de las 16. El eje vertical representa las cubas, pero se ha modificado el código para que tan sólo aparezcan las cubas involucradas en la secuencia que se analice en cada caso. Las barras horizontales representan las diferentes tareas que aparecen en esta secuencia. El color de las barras representa cada tipo de pieza, por ejemplo las barras verdes representan las tareas de los trabajos tipo TRABAJO_C. El número que está en la esquina superior derecha de cada barra representa la pieza a la que pertenece cada tarea, de forma que las barras verdes acompañadas por un 1 representan las tareas de la pieza TRABAJO_C_1.

Además, a parte de la ventaja en cuanto al tiempo de generación del diagrama, se consigue una presentación de los datos mucho más compacta y manejable.

3.4 VARIABLES

Como ya se sabe, para cumplir el objetivo de este proyecto, es necesario aportar ciertos puntos de partida que ayuden a la optimización del problema. Dichos

puntos de partida se consiguen modificando la variable adecuada de entre las decenas que modelan el problema, teniendo cuidado de no modificar aquellas que llevan a absurdos.

Cabe destacar que al principio se cometió un error en algunas secuencias iniciales en las que se trató de imponer una solución inicial: modificar la variable “vIndPrecedGlob2”. De modo que al asignar prioridades con esta variable, se asignaba una prioridad total, impidiendo cualquier tipo de adelantamiento a lo largo de toda la secuencia. Es decir, en cuanto entraba una pieza en la secuencia, no podía entrar otra hasta que la anterior hubiese terminado y salido de la secuencia. En la siguiente imagen se muestra un ejemplo en el que se modificó la mencionada variable.

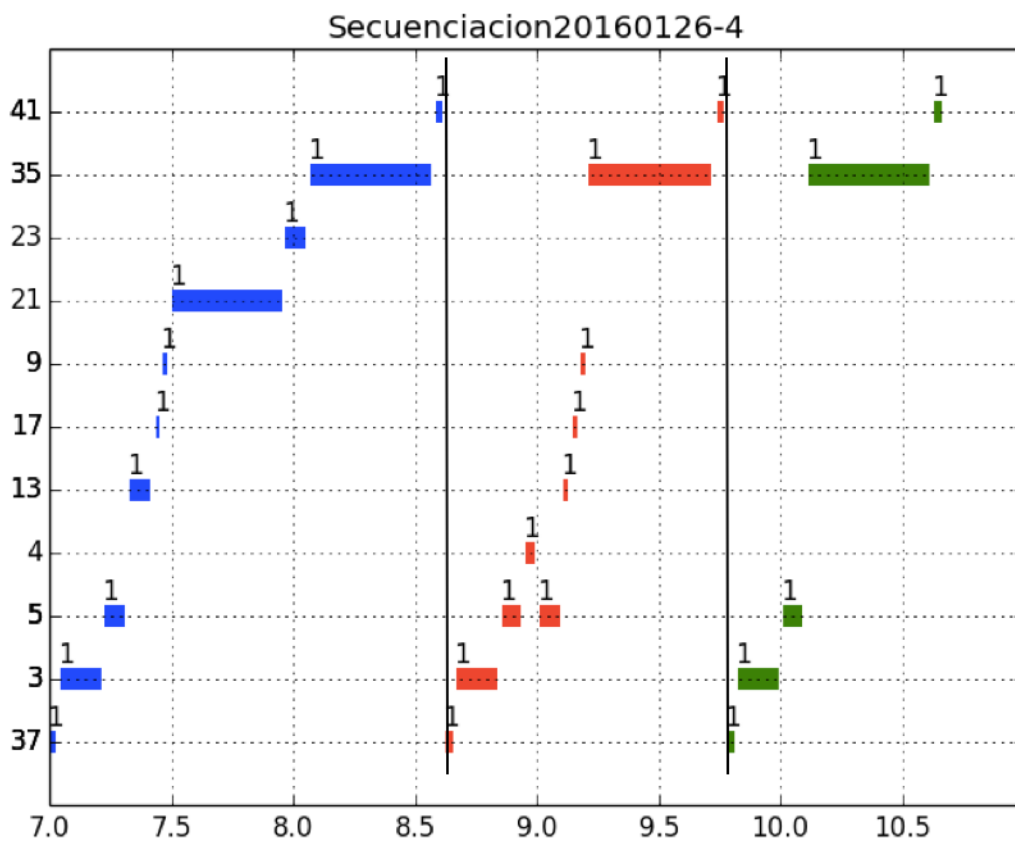


Imagen 12: Ejemplo de secuencia modificada bajo la variable vIndPrecedGlob2

Como se puede observar, la primera tarea de la pieza roja (cuba 37) no entra en la secuencia hasta que ha terminado la última tarea de la pieza azul (cuba 41); y lo mismo sucede con la pieza roja y la verde. Se han incluido dos líneas verticales que ayudan a la visualización de este hecho.

La variable que realmente se tiene que modificar se llama: “vIndPrecedenciak(i,j,ii,jj,k)”. La cual recibe cinco parámetros y marca la prioridad

de una tarea, j , de una pieza, i , frente a otra tarea, jj , de otra pieza, ii , en una cuba k . En la siguiente imagen se muestra la misma secuencia que en la de la imagen 12, pero modificando la variable adecuada.

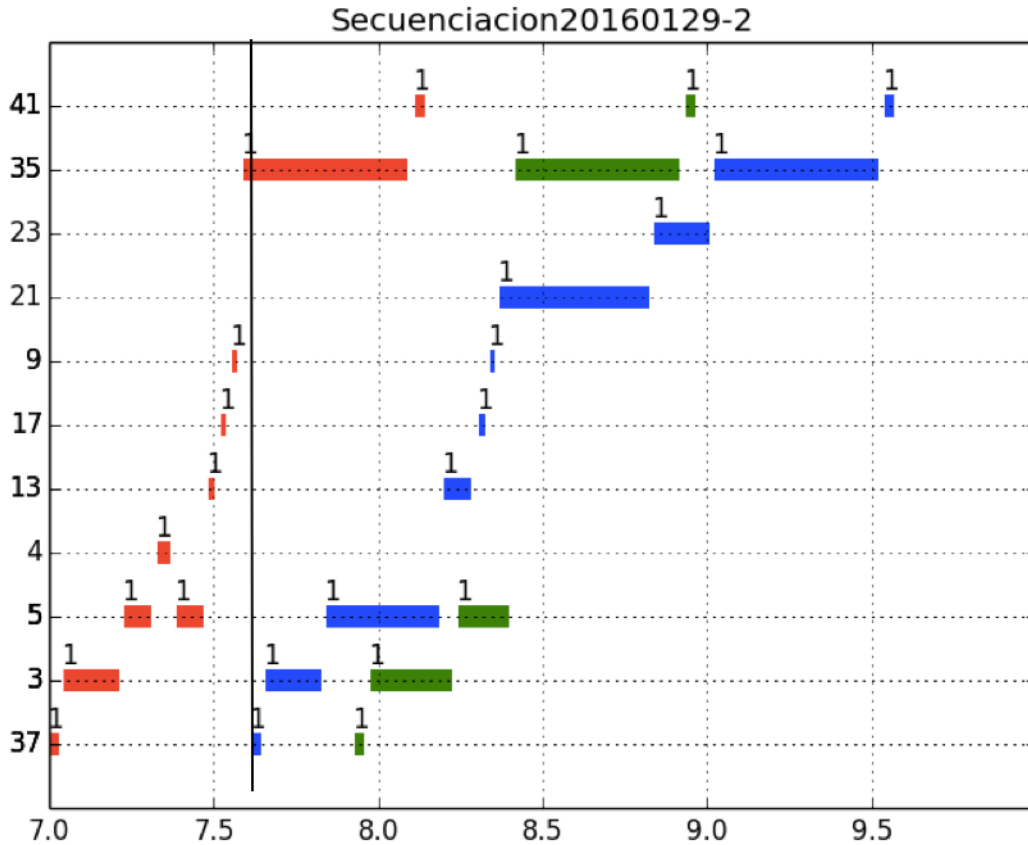


Imagen 13: Ejemplo de secuencia modificada bajo la variable $vIndPrecedenciaK$

Como se puede observar, en este caso la primera tarea de la pieza azul (cuba 37) no espera a que la última tarea de la pieza roja (cuba 41) haya terminado, sino que aprovecha el tiempo de procesado de la tarea de la cuba 35 de la pieza roja para comenzar el procesado de la pieza azul.

3.5 HEURÍSTICOS – ESTADO DEL ARTE

Como ya se ha mencionado, la bibliografía relacionada con este proyecto es muy escasa y tan sólo se pueden apreciar dos heurísticos que se puedan aplicar a este proyecto y que constituyen la base de la creación de todos los heurísticos del presente proyecto. Principalmente son [4]:

- *Shortest/Longest Processing Time (SPT/LPT)*. Consiste en ordenar las piezas según el tiempo agregado de cada tarea. Primero aquellas de menor duración (SPT), o aquellas de mayor duración (LPT). Pero tiene el

inconveniente de que no se conoce el tiempo real, ya que los datos de tiempos de los que se disponen constan de un rango admisible. Es decir, dada una tarea, la pieza debe estar un mínimo de tiempo en una determinada cuba, pero sin exceder el límite fijado para dicha tarea.

- Shortest/Longest Operation Time (SOT/LOT). Consiste en ordenar las piezas dando prioridad según la duración de la tarea más larga (LPT) o más corta (SPT). A parte de tener el mismo inconveniente que la anterior, además tiene el inconveniente de que la variabilidad de tiempos de las tareas es muy pequeña, es decir, la mayoría de las tareas tienen duraciones parecidas si no iguales.

3.6 TIEMPOS DE PROCESADO

Como se ha mencionado anteriormente, una de las peculiaridades de este problema es que los tiempos de procesado proporcionados consisten en rangos de tiempo admisibles y por lo tanto se desconocen los tiempos reales que usa GAMS. Esto obliga a usar un parámetro alternativo que ayude a comparar y asignar prioridades a las distintas piezas y tareas.

Dicho parámetro se denomina t_{med_pieza} y como su nombre indica, es característico de cada pieza. A continuación se muestra cómo se ha procedido en el caso del análisis de comportamiento de la secuenciación, cuyos datos han sido proporcionados para la realización de este proyecto.

Trabajo\cuba	37	3	5	4	5	13	17	9	21	23	35	41
ANODIZADO	1.7	(10-15)	(5-30)	0	0	5.0	(1-2)	(1-5)	27.1	(5-10)	(30-90)	1.7
LIMPIEZA	1.7	(10-15)	(5-90)	2.4	5.0	1.0	(1-2)	(1-5)	0	0	(30-90)	1.7
DESENGRASE	1.7	(10-15)	(5-15)	0	0	0	0	0	0	0	(30-90)	1.7

Tabla 2: Rangos de tiempos de procesado del análisis de comportamiento

Partiendo de los rangos de tiempos proporcionados mostrados en la tabla1, se calculan los t_{med_tarea} , que representa la media de tiempo entre el máximo y mínimo de cada tarea. A continuación se muestran los t_{med_tarea} del análisis del comportamiento de la secuenciación.

Trabajo\cuba	37	3	5	4	5	13	17	9	21	23	35	41
ANODIZADO	1.7	12.5	17.5	0	0	5.0	1.5	3.0	27.1	7.5	60.0	1.7
LIMPIEZA	1.7	12.5	47.5	2.4	5.0	1.0	1.5	3.0	0	0	60.0	1.7
DESENGRASE	1.7	12.5	10.0	0	0	0	0	0	0	0	60.0	1.7

Tabla 3: t_{med_tarea} de las tareas del análisis de comportamiento

Una vez calculados los t_{med_tarea} de cada pieza, se calcula t_{med_pieza} como la suma de los t_{med_tarea} que constituyen dicha pieza. Lo que lleva a los resultados que se muestran en la siguiente tabla.

Trabajo	Duracion
ANODIZADO	137.5
LIMPIEZA	136.3
DESENGRASE	85.9

Tabla 4: t_{med_pieza} de las piezas del análisis de comportamiento

Así pues, de ahora en adelante, cada vez que en una optimización propia del análisis del comportamiento se vaya a introducir alguna prioridad basada en la duración del tiempo de procesado agregado de las tareas, se hará usando los parámetros t_{med_pieza} que están en la tabla 4. Es preciso incidir en que se podría definir como un atributo a priori, es decir, que realmente no se conocen las duraciones que resultarán de cada secuenciación; razón por la cual, en las secuencias que se muestran más adelante los tiempos que se muestran y ayudan al análisis son los reales obtenidos de la optimización y calculados gracias a un script de Python que se ha desarrollado.

3.7 PUNTOS DE PARTIDA

Como se ha comentado ya, en este proyecto hay que aportar los puntos de partida adecuados que permitan asignar prioridades de forma heurística.

Para asignar las variables de precedencia que rigen las prioridades, resultaron de ayuda los conocimientos en el lenguaje de programación Python para crear un script que permitiese generar el extenso texto que hay que introducir en el archivo de GAMS, con el objetivo de establecer ciertas prioridades que conduzcan al punto de partida deseado. Es decir, a lo largo de todo el proyecto se han desarrollado diferentes scripts cuya función ha sido generar un archivo de texto con las instrucciones de asignación de variable. En la siguiente imagen se puede ver una fracción de un ejemplo que justifica la necesidad de un método externo (Python en este caso) que genere el texto, que puede rondar las 300 líneas o más. También se puede observar que el cierto grado de repetitividad del texto es lo que ha permitido la ayuda externa.

```

vIndPrecedenciak.fx('TRABAJO_A_1','1','TRABAJO_A_5','1','3')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','2','TRABAJO_A_2','2','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','2','TRABAJO_A_2','4','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','2','TRABAJO_A_3','2','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','2','TRABAJO_A_3','4','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','2','TRABAJO_A_4','2','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','2','TRABAJO_A_4','4','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','2','TRABAJO_A_5','2','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','2','TRABAJO_A_5','4','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','3','TRABAJO_A_2','3','4')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','3','TRABAJO_A_3','3','4')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','3','TRABAJO_A_4','3','4')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','3','TRABAJO_A_5','3','4')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','4','TRABAJO_A_2','2','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','4','TRABAJO_A_2','4','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','4','TRABAJO_A_3','2','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','4','TRABAJO_A_3','4','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','4','TRABAJO_A_4','2','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','4','TRABAJO_A_4','4','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','4','TRABAJO_A_5','2','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','4','TRABAJO_A_5','4','5')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','5','TRABAJO_A_2','5','13')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','5','TRABAJO_A_3','5','13')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','5','TRABAJO_A_4','5','13')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','5','TRABAJO_A_5','5','13')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','6','TRABAJO_A_2','6','17')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','6','TRABAJO_A_3','6','17')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','6','TRABAJO_A_4','6','17')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','6','TRABAJO_A_5','6','17')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','7','TRABAJO_A_2','7','9')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','7','TRABAJO_A_3','7','9')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','7','TRABAJO_A_4','7','9')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','7','TRABAJO_A_5','7','9')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','10','TRABAJO_A_2','10','35')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','10','TRABAJO_A_3','10','35')=1;
vIndPrecedenciak.fx('TRABAJO_A_1','10','TRABAJO_A_4','10','35')=1;

```

Imagen 14: Fragmento de un input empleado en GAMS para una optimización cualquiera

Dichos scripts han ido variando a lo largo del proyecto a medida que se iba refinando el proceso de desarrollo. Es decir, los scripts han pasado de hacer el cálculo automáticamente en función de los tiempos que se asignase a cada tarea (como ya se ha mencionado, los datos de los que se disponen son rangos de tiempo admisibles), a generar el archivo en función del orden que se le asigne manualmente. Puede parecer que se ha retrocedido en cuanto a automatización del proceso, pero hay que tener en cuenta que con el script que se ha utilizado al final, se puede aplicar cualquier criterio sin tener que hacer modificaciones en el código a parte del propio orden que se desea introducir. Es decir, se pueden generar todos los archivos de texto a partir de un único script de Python. Además, el cálculo del orden de las piezas es bastante sencillo en los diferentes heurísticos y se puede hacer a mano sin complicaciones. Por el contrario, la línea que siguió al principio obligaba a cambiar de script radicalmente en función del

algoritmo de ordenación que se eligiese, a parte de que dependía de los tiempos que se asignasen a mano a cada tarea.

3.8 SECUENCIACIONES

Al principio del desarrollo del proyecto se hicieron numerosas pruebas sobre un único caso proporcionado por el director para el análisis del comportamiento. Dicho caso contaba con 3 tipos de piezas, cada una de ellas con distintas tareas según se ve en la tabla 2, donde se pueden ver los rangos admisibles de cada tarea.

El número de piezas que estaban disponibles de cada tipo en el análisis del comportamiento se muestra en la siguiente tabla.

Piezas	Número de piezas
ANODIZADO	4
LIMPIEZA	4
DESENGRASE	2

Tabla 5: Número de piezas de cada tipo en análisis de comportamiento

Por otro lado, en este punto se llevó a cabo el primer contacto con la herramienta GAMS y algunas de las secuencias que se llevaron a cabo carecen de utilidad más allá de la ayuda al aprendizaje de la herramienta.

Además, junto a la realización de estas pruebas se fueron desarrollando los scripts de Python que permitían el uso de los archivos de texto como inputs así como el script que permite la visualización de los resultados en forma de diagrama de Gantt.

A continuación se comentan las optimizaciones más relevantes que se llevaron a cabo así como las conclusiones de las mismas. En la siguiente tabla se puede ver un resumen de dichas optimizaciones.

Secuencias	Limite tiempo	Tareas	Comentarios
Secuenciacion20160204	2 minutos	2 lim, 2 anod, 2 desen	Anodizado sin tarea exclusiva
Secuenciacion20160204-2	2 minutos	2 lim, 2 anod, 2 desen	Limpieza con atrea adicional
Secuenciacion20160204-3	2 minutos	2 lim, 2 anod	Sin desengrasado
Secuenciacion20160204-4	5 minutos	3 lim, 3 anod	Sin desengrasado
Secuenciacion20160204-5	sin limite	3 lim, 3 anod, 2 desen	Sin fijar ninguna
Secuenciacion20160205	sin limite	3 lim, 3 anod, 2 desen	Fijo de menor a mayor sin tener en cuenta desengrasado
Secuenciacion20160208	sin limite	3 lim, 3 anod, 2 desen	Igual que antes pero de mayor a menor
Secuenciacion20160208-3	sin limite	3 lim, 3 anod, 2 desen	Amplio el tiempo maximo
Secuenciacion20160210	6 horas	4 lim, 4 anod, 2 desen	Original
Secuenciacion20160210-2	1 hora	4 lim, 4 anod, 2 desen	Asignacion creciente de todas las piezas
Secuenciacion20160210-3	1 hora	4 lim, 4 anod, 2 desen	Fijo de menor a mayor sin desengrasado
Secuenciacion20160616	1 hora	3 lim, 3 anod, 2 desen	Asignacion aleatoria
Secuenciacion20160616-2	1 hora	3 lim, 3 anod, 2 desen	Asignacion aleatoria
Secuenciacion20160616-3	1 hora	3 lim, 3 anod, 2 desen	Asignacion aleatoria
Secuenciacion20160616-4	1 hora	3 lim, 3 anod, 2 desen	Asignacion creciente de todas las piezas menos limpieza
Secuenciacion20160616-5	1 hora	3 lim, 3 anod, 2 desen	Asignacion creciente de todas las piezas menos anodizado
Secuenciacion20160617	1 hora	3 lim, 3 anod, 2 desen	Asignacion creciente de todas las piezas menos 2 de limpieza
Secuenciacion20160617-2	1 hora	3 lim, 3 anod, 2 desen	Asignacion creciente de todas las piezas menos 2 de anodizado
Secuenciacion20160617-3	1 hora	3 lim, 3 anod, 2 desen	Asignacion creciente de todas las piezas
Secuenciacion20160617-4	1 hora	3 lim, 3 anod, 2 desen	Asignacion decreciente de todas las piezas

Tabla 6: Resumen de las optimizaciones más relevantes

Como se comentó anteriormente, en los diagramas de Gantt utilizados cada color representa un tipo de pieza. En los diagramas de Gantt que se muestran a continuación, y que pertenecen al análisis del comportamiento, se ha empleado la siguiente relación cromática.

Piezas	Color
ANODIZADO	Azul
LIMPIEZA	Rojo
DESENGRASE	Verde

Tabla 7: Relación de colores para los diagramas de Gantt del análisis de comportamiento

Secuenciacion20160204

Como se puede ver en la siguiente tabla, en secuenciacion20160202 (formada por dos piezas de cada tipo) las piezas de anodizado son las que resultaron tener la mayor duración y en la secuencia se situó una detrás de la otra, como se puede ver en la imagen.

Secuenciacion20160202	
D2	58.4
L2	58.8
D1	63.4
L1	90.2
A1	91.7
A2	109.2

Tabla 8: Tiempos de procesamiento de secuenciacion20160202

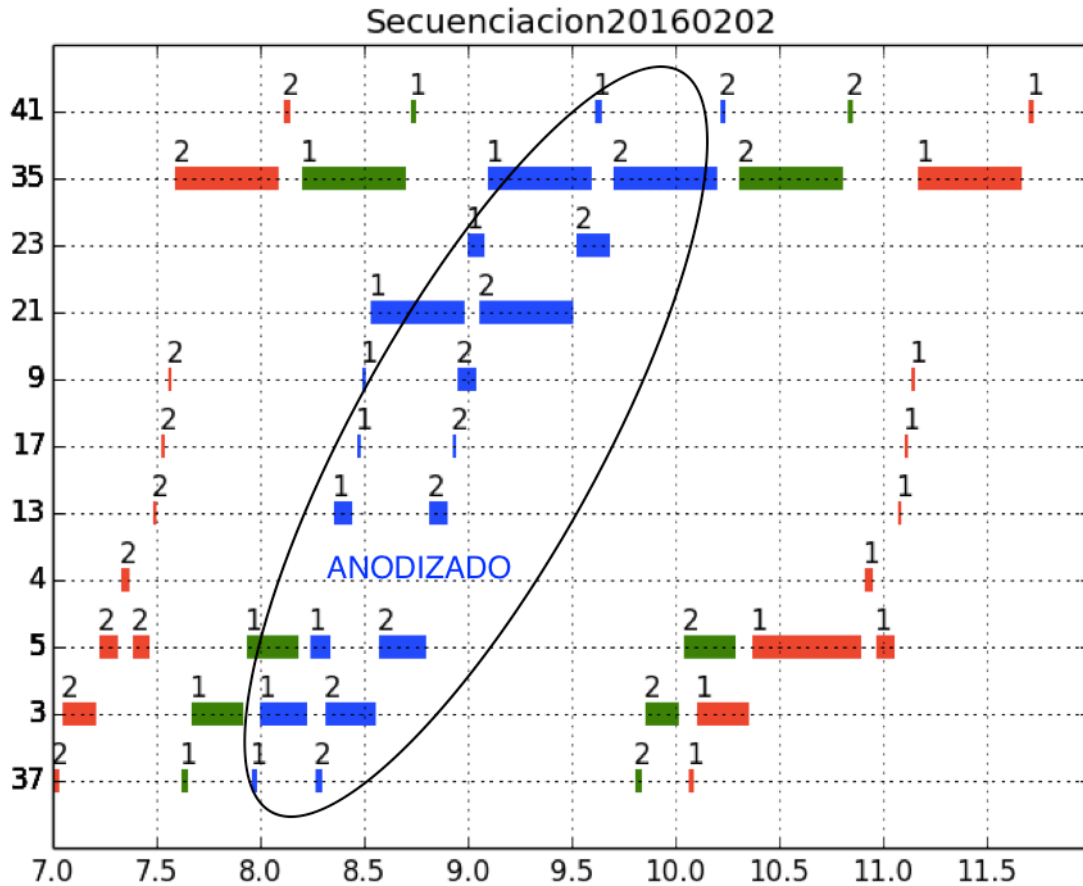


Imagen 15: Diagrama de Gantt secuenciacion20160202

Dado que las piezas de anodizado (azul) tienen dos tareas exclusivas (cuba 23 y cuba 21) mientras que las piezas de limpieza (rojo) solo tienen una (cuba 4) y las de desengrasado (verde) ninguna, en secuenciación20160204 se ha optado por eliminar una de las tareas exclusivas de anodizado para ver su influencia. En concreto se eliminó la correspondiente a la cuba 21 por ser la que más tiempo requería en la secuenciacion20160202.

Como se muestra en la siguiente tabla, el resultado es que las piezas de anodizado (azul) volvieron a ser las de mayor duración y volvieron a estar juntas en la secuencia.

Secuenciacion20160204	
L1	58.8
D1	63.4
D2	63.4
L2	74.3
A2	75.5
A1	93.0

Tabla 9: Tiempos de procesamiento de secuenciacion20160204

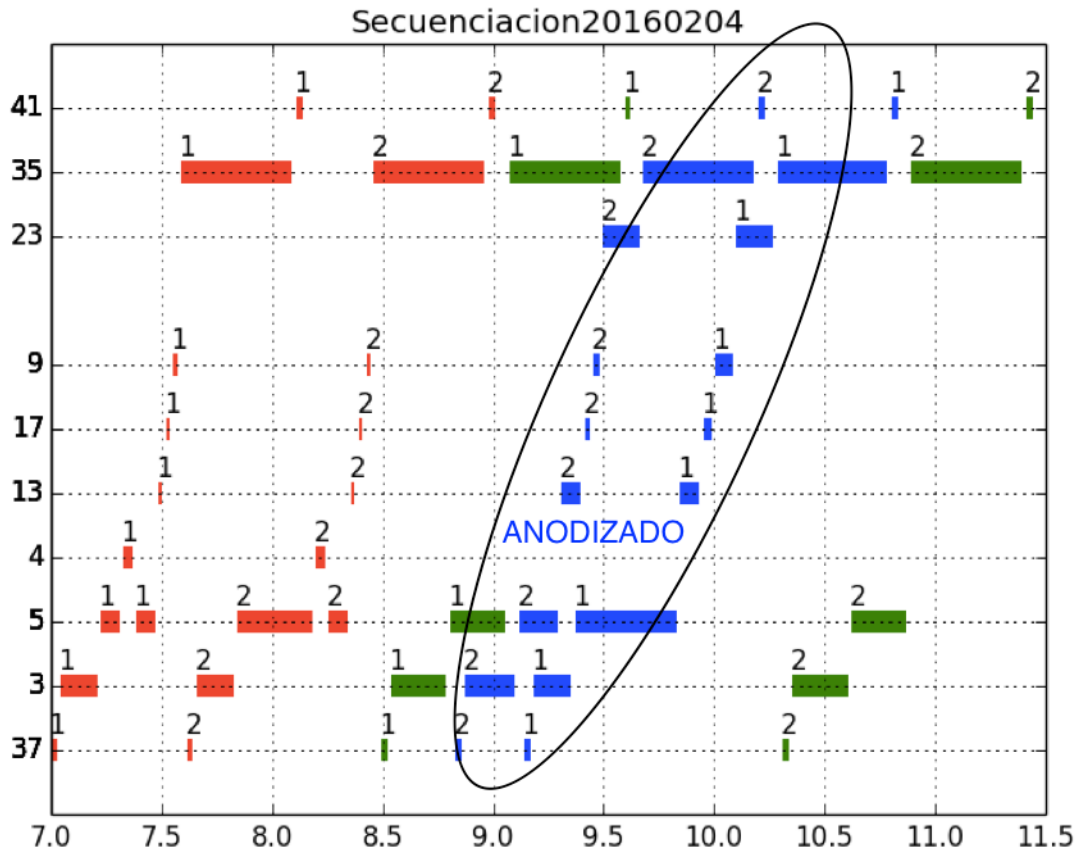


Imagen 16: Diagrama de Gantt secuenciacion20160204

Por otro lado, en las secuencias realizadas hasta el momento se ha apreciado cierta tendencia a una ordenación de menor a mayor duración. También es destacable que si no se tuviesen en cuenta las piezas de desengrasado (verde), toda la secuencia estaría ordenada de menor a mayor duración del tiempo agregado de cada pieza.

Secuenciacion20160204-2

En este caso se quiere analizar la influencia de las piezas con tareas exclusivas, para lo cual se ha partido de la secuencia anterior en la que se eliminó la tarea de la cuba 21 de las piezas de anodizado (azul), pero en este caso se ha añadido una nueva tarea exclusiva a las piezas de limpieza (rojo). De forma que las piezas de limpieza (rojo) ahora tienen dos tareas exclusivas mientras que las de anodizado (azul) solo tienen una y las de desengrasado (verde) ninguna.

Secuenciacion20160204-2	
D1	58.4
L2	59.8
D2	63.4
A1	71.9
L1	74.8
A2	75.5

Tabla 10: Tiempos de procesado de secuenciacion20160204-2

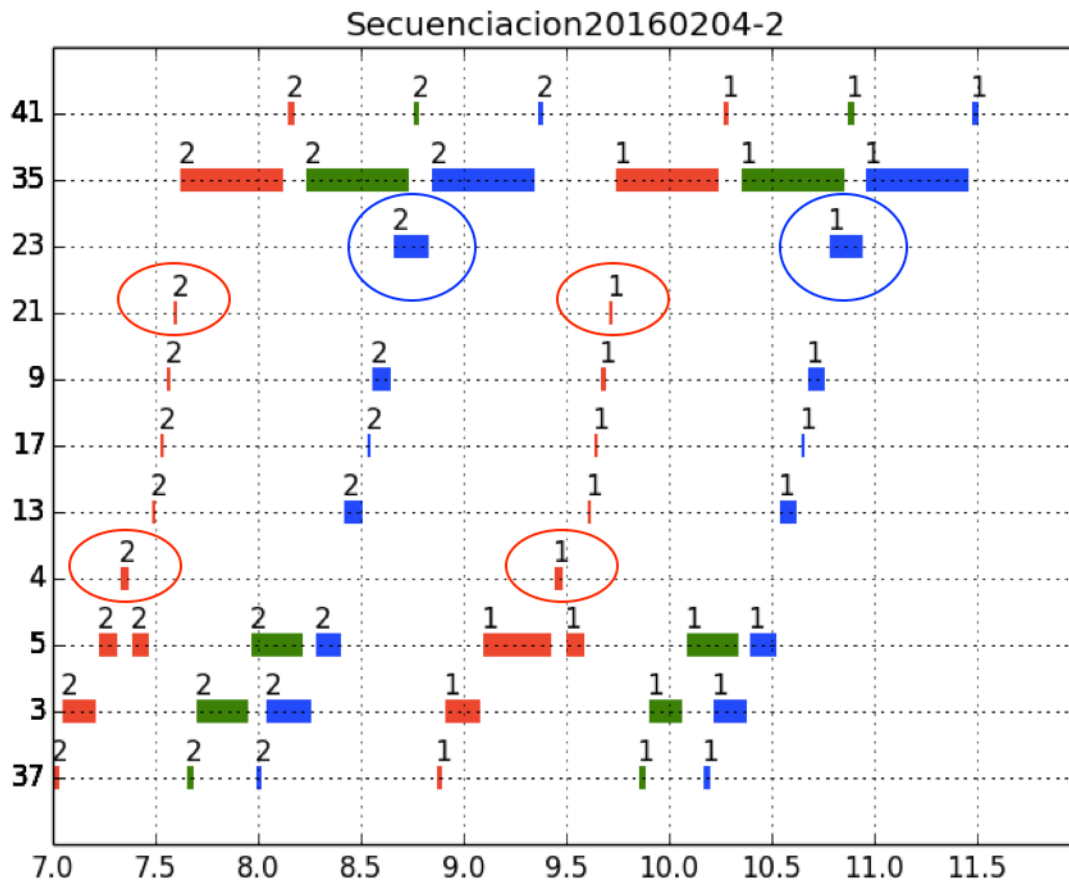


Imagen 17: Diagrama de Gantt secuenciacion20160204-2

En la imagen se han marcado las tareas exclusivas de las piezas. Como se observa, cada pieza de limpieza tiene dos tareas exclusivas mientras que cada pieza de anodizado tiene una tarea exclusiva.

El resultado es que las dos piezas con mayor duración, que en este caso son Anodizado2 (azul) y Limpieza1 (rojo), vuelven a estar juntas y en el medio de la secuencia. Por otro lado, las piezas de desengrasado (verde) vuelven a estar custodiando a las dos piezas con mayor duración.

Secuenciacion20160204-3 y 4

El objetivo de esta optimización es tratar de dar respuesta a los resultados de la secuenciacion20160204, donde se observó la ordenación de todas las piezas en

orden creciente salvo las de desengrasado. Los resultados obtenidos se muestran en las siguientes imágenes, donde se puede observar la disposición de las piezas en orden creciente según la duración agregada de las tareas de cada pieza.

Por ello, en secuenciacion20160204-3 se ha llevado a cabo la optimización libre (sin fijar ninguna prioridad) de 2 piezas de cada tipo sin incluir las piezas de desengrasado (verde). En secuenciacion20160204-4 se ha seguido la misma línea, pero empleando 3 piezas de cada tipo en lugar de 2.

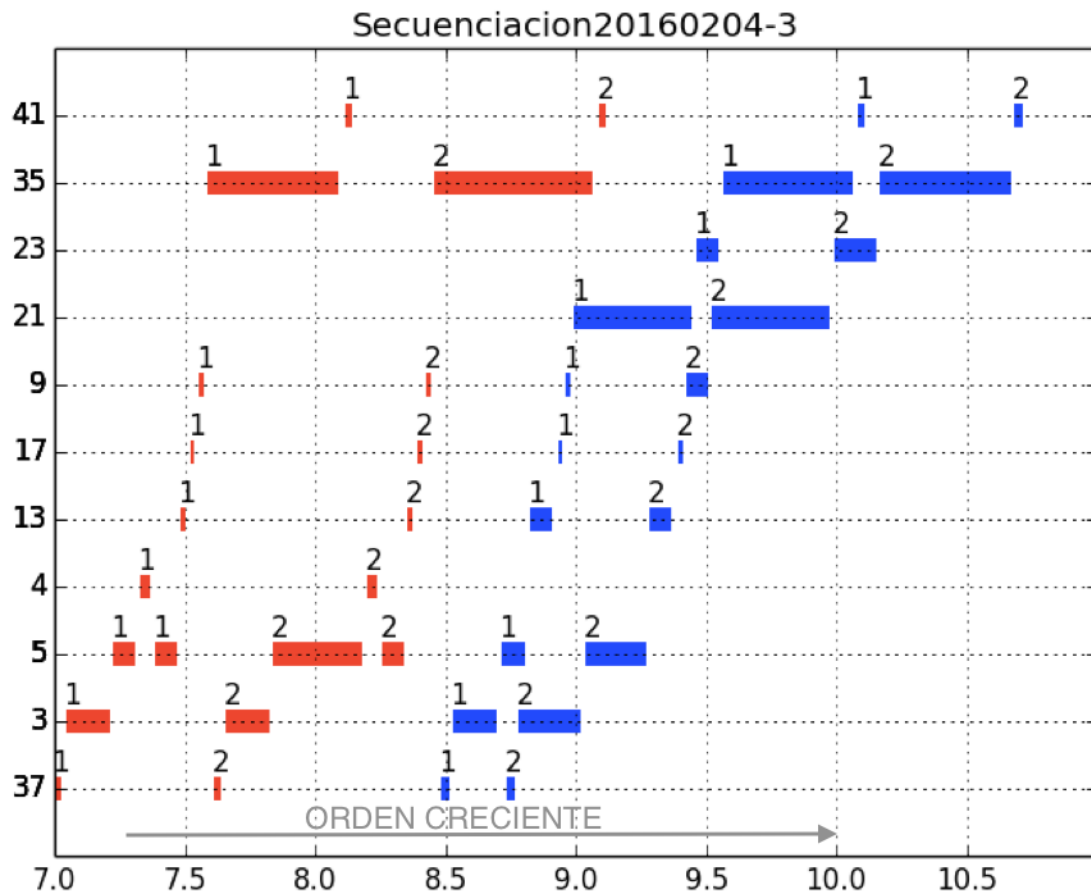


Imagen 18: Diagrama de Gantt de secuenciacion20160204-3

Secuenciacion20160204-3	
L1	58.8
L2	80.5
A1	88.2
A2	109.2

Tabla 11: Tiempos de procesado de secuenciacion20160204-3

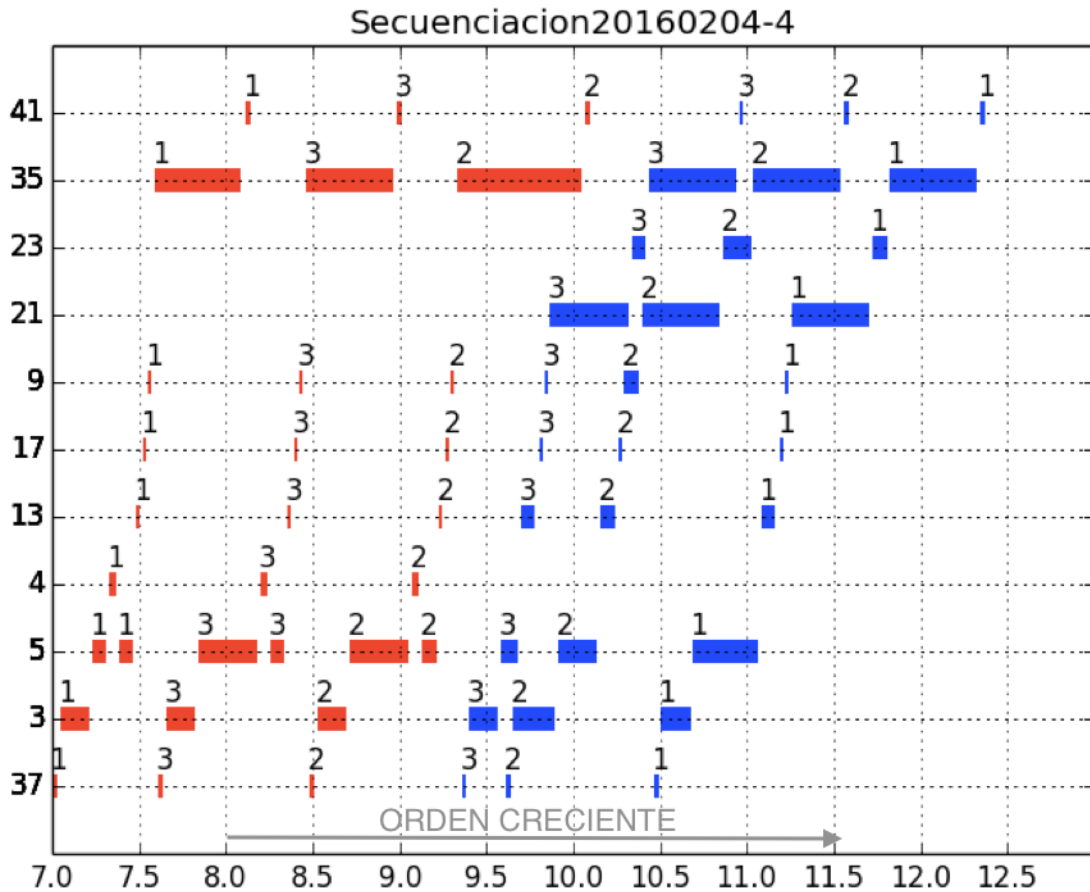


Imagen 19: Diagrama de Gantt de secuenciacion20160204-4

Secuenciacion20160204-4	
L1	58.8
L3	74.3
L2	87.1
A3	88.2
A1	105.0
A2	109.2

Tabla 12: Tiempos de procesado de secuenciacion20160204-4

Ante estos resultados, se decidió realizar una secuenciación en la que se asignase una prioridad a todas las piezas salvo a las de desengrasado. Pero para poder determinar cómo de buenos son los resultados, primero es necesario realizar una optimización libre de todas las piezas con la que poder comparar, es decir, sin introducir ninguna solución inicial.

Secuenciacion20160204-5 y secuenciacion20160208-3

De modo que en secuenciacion20160204-5 se realizó la secuenciación de 3 piezas de anodizado, 3 de limpieza y 2 de desengrasado con un límite de 1 hora. Como se llegó al límite de tiempo, se volvió a realizar la secuenciación, pero esta vez con un límite de 2 horas (secuenciacion20160208-3). El resultado de la

optimización fue exactamente el mismo en las dos secuencia, de modo que se puede asumir este resultado como el óptimo.

	Final solve	Elapsed
Secuenciacion20160204-5	6.361667	1:00:01.804
Secuenciacion20160208-3	6.361667	2:00:01.330

Tabla 13: Comparación de los resultados obtenidos en una misma secuencia para dos límites de tiempo distintos

No obstante, a pesar de haber obtenido el mismo valor existe la posibilidad de que la secuencia resultante sea distinta. Sin embargo, analizando el diagrama de Gantt de ambas optimizaciones se observa que la secuencia obtenida fue la misma.

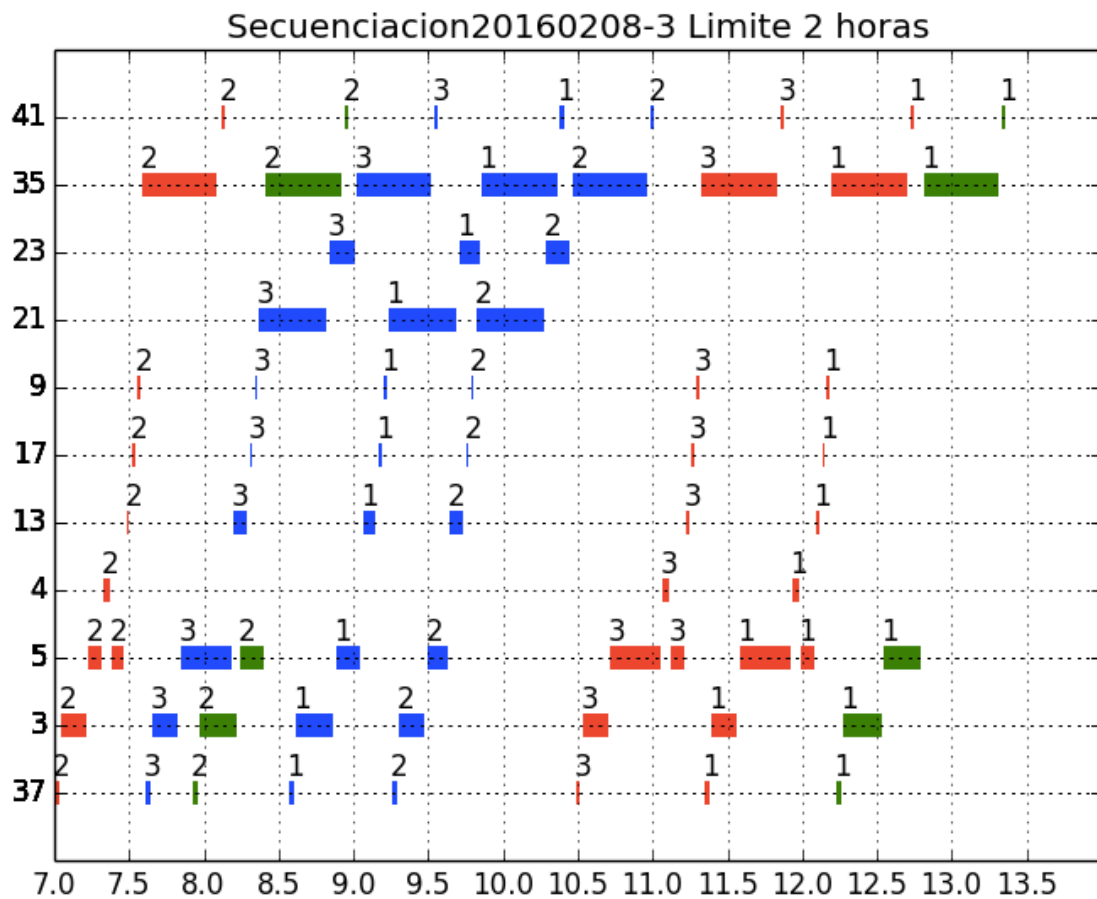


Imagen 20: Resultado de la secuenciacion20160208-3 con límite de 2 horas

Secuenciación20160205 y secuenciación20160208

Una vez realizada la secuenciación libre, se han llevado a cabo la secuenciaciones 20160617-3 y 4, donde se realiza una asignación de prioridad de todas las piezas en orden creciente y decreciente respectivamente. Como se

puede observar en la tabla 14, los resultados en estas dos secuencias son peores que en el caso en el que se dejaron libres todas las piezas.

Tras haber realizado la secuencia libre, se realizó la secuenciación20160205 en la que se fijaron de menor a mayor todas las piezas salvo las de desengrasado, que se dejaron libres. El resultado es llamativo, pues es exactamente igual al obtenido en la secuenciacion20160208-3 (que se asume óptimo) pero esta vez se obtuvo en tan sólo 35 segundos.

En la secuenciacion20160208 se siguió el mismo camino que en la secuenciacion20160205 pero esta vez ordenando las piezas de mayor a menor (salvo las de desengrasado). El resultado vuelve a ser llamativo porque una vez más se obtuvo exactamente el mismo resultado que en el caso óptimo.

	Final solve	Elapsed
Secuenciacion20160204-5	6.361667	1:00:01.804
Secuenciacion20160208-3	6.361667	2:00:01.330
Secuenciacion20160205	6.361667	0:00:35.594
Secuenciacion20160208	6.361667	0:00:41.157
Secuenciacion20160617-3	6.586667	0:00:06.403
Secuenciacion20160617-4	6.845	0:00:04.114

Tabla 14: Comparación de una misma secuencia sin ordenación versus ordenación total versus ordenación parcial

Se observa lo siguiente:

- Cuando se asignan prioridades a todas las piezas, los resultados empeoran frente al óptimo y posiblemente frente a cualquier heurístico. Pues no se le da libertad a GAMS para poder encontrar una secuencia adecuada, sino que está predeterminada por el usuario.
- Cuando se asignan prioridades a todas las piezas salvo a las de desengrasado, no solo se obtienen mejores resultados que cuando se asignan prioridades a todas las piezas, sino que se consigue el óptimo.

Secuenciacion20160616 , 2 y 3

Con el propósito de descubrir la peculiaridad de los resultados obtenidos en secuenciacion20160205 y secuenciacion20160208; en la secuenciacion20160616, 2 y 3 se ha seguido la línea anterior, pero en este caso en lugar de ordenar los trabajos bajo un criterio determinado, se ha hecho de forma aleatoria y también sin fijar el desengrasado. En los tres casos se han seguido las ordenaciones aleatorias que se muestran a continuación.

20160616	20160616-2	20160616-3
A1	L3	L3
L3	A1	A3
A3	L1	A2
L2	A2	L1
A2	A3	L2
L1	L2	A1

Tabla 15: Ordenaciones aleatorias de las secuencias 20160616, 2 y 3

Los resultados obtenidos han sido los siguientes

	Final solve	Elapsed
Secuenciacion20160616	6.901667	0:00:15.000
Secuenciacion20160616-2	6.415	0:00:21.360
Secuenciacion20160616-3	6.415	0:00:23.426

Tabla 16: Comparación de los resultados de 3 secuencias con orden aleatorio

Como se puede observar comparando con la tabla13, los resultados han empeorado en cierta medida, pues el óptimo es 6.361667. No obstante el factor *Elapsed Time* sigue siendo muy pequeño. En la siguiente tabla se trata de cuantificar el empeoramiento de las tres secuencias aleatorias en minutos respecto al óptimo.

	Desviación [min]
Secuenciacion20160616	32.4
Secuenciacion20160616-2	3.19998
Secuenciacion20160616-3	3.19998

Tabla 17: Desviación en minutos respecto al óptimo

Esto puede llevar a pensar que la ordenación previa bajo el criterio de los tiempos, ya sea en sentido creciente o decreciente, es el factor clave que conduce a unos resultados mejores. Sin embargo, si se observan las duraciones reales del procesado de cada pieza (con la ayuda del script de Python que se desarrolló) tanto en secuenciacion20160205 como en secuenciacion20160208, y asumiendo una ordenación de menor a mayor y de mayor a menor respectivamente, el orden aplicado no es exactamente igual al que se debería haber introducido. A continuación se muestra una tabla que ayuda a comprender este aspecto.

Secuenciacion20160205		Secuenciacion20160208	
Presupuesto	Real	Presupuesto	Real
L1	L1	A1	L1
L2	L3	A2	A2
L3	L2	A3	A1
A1	A3	L1	A3
A2	A2	L2	L3
A3	A1	L3	L2

Tabla 18: Comparación de la ordenación basada en tiempos a priori frente a tiempos reales

En la columna de “Presupuesto” se ve reflejado el orden introducido basado en el parámetro del tiempo medio agregado: de menor a mayor para 20160205 y de mayor a menor para 20160208; y en la columna “Real” se ve reflejada la ordenación real que tendría que se tendría que haber introducido bajo el mismo criterio una vez que se han conocido los tiempos reales.

De modo que en realidad, es posible que se hayan obtenido mejores resultados debido al hecho de que se ordenan las piezas por familias y no debido a una asignación por orden decreciente/creciente. Sin embargo, no es posible demostrar esta hipótesis porque nunca se conocen los tiempos reales que GAMS va a emplear en cada optimización y en cualquier caso, cuando se introducen las asignaciones de prioridad de las piezas ya sea por orden creciente o decreciente, siempre se introducirán por familias de piezas.

Secuenciacion20160616-4 y 5

No obstante, a pesar de que la intuición ha llevado a aislar la asignación de prioridad a las piezas de desengrasado, hasta ahora no se ha demostrado que estén piezas sean especiales. Es decir, no se ha demostrado que los resultados obtenidos en la secuenciacion20160205, donde no se asignaba prioridad a las piezas de desengrasado, sean característicos de dicha pieza. Es necesario comprobar que no se obtienen los mismos resultados cuando se aíslan las demás piezas.

De modo que siguiendo el estilo de secuenciacion20160205, se van a ordenar todas las piezas de menor a mayor salvo las de limpieza (secuenciacion20160616-4). De igual manera, en secuenciacion201606-5 se han ordenado todas las piezas de menor a mayor salvo las de anodizado. Basándose en las conclusiones extraídas en relación a la ordenación de las piezas, en estas dos secuencias tan solo se ha probado ordenando de menor a mayor.

A continuación se muestran los resultados obtenidos en los tres casos.

	Sin fijar	Final solve	Elapsed
Secuenciacion20160205	Desengrasado	6.361667	0:00:35.594
Secuenciacion20160616-4	Limpieza	6.38	0:04:25.785
Secuenciacion20160616-5	Anodizado	6.38	0:04:18.912

Tabla 19: Comparación de los resultados cuando no se fija ninguna pieza de un tipo

Observando la tabla anterior se puede llegar a la prematura conclusión de que esto se traduce en una indiscutible peculiaridad de las piezas de desengrasado.

Sin embargo, es preciso remarcar que en el caso de la secuencia en la que se deja libre el desengrasado, se han dejado libres dos piezas mientras que en las secuencias correspondientes a anodizado y limpieza se han dejado libres 3 piezas y esto puede influir en los resultados.

Secuenciaion20160617 y 2

Por este motivo, en tanto en la secuenciaion20160617 como en la secuenciaion20160617-2, se han dejado libres 2 piezas de limpieza y desengrasado respectivamente en lugar de 3.

	Sin fijar	Final solve	Elapsed
Secuenciacion20160205	Desengrasado	6.361667	0:00:35.594
Secuenciacion20160617	Limpieza	6.586667	0:00:56.181
Secuenciacion20160617-2	Anodizado	6.586667	0:00:34.888

Tabla 20: Comparación de los resultados cuando no se fijan dos piezas de un tipo

Como se puede observar en la tabla anterior y en la **tabla19**, los resultados obtenidos cuando no se fijan las piezas de limpieza o de anodizado son peores, en cualquier caso, que cuando no se fijan las piezas de desengrasado. Ahora sí se puede afirmar que las piezas de desengrasado tienen cierta peculiaridad de la que carecen las demás piezas de este enunciado.

Secuenciacion20160210, 2 y 3

Con el objetivo de proporcionar mayor fiabilidad a los resultados obtenidos anteriormente, se han desarrollado 3 optimizaciones que siguen la línea de las anteriores, pero esta vez partiendo de una secuencia que contaba con 4 piezas de limpieza, 4 de anodizado y 2 de desengrasado. Es decir, se introdujeron todas las piezas que fueron proporcionadas para el análisis del comportamiento.

Del mismo modo que en la secuencia anterior, primero se realizó la optimización libre (secuenciacion20160210), sin imponer ninguna condición sobre el orden de las piezas, con un límite de 6 horas para asegurarse de que el valor obtenido se puede considerar como óptimo.

Tras esto, se realizó una optimización (secuenciacion20160210-2) con la imposición de un orden creciente sobre todas las piezas (incluidas las de desengrasado) bajo el criterio de tiempo medio agregado. Y también se realizó una optimización igual que la anterior (secuenciacion20160210-3) pero sin asignar ningún orden a las piezas de desengrasado.

A continuación se muestran los resultados.

	Final solve	Elapsed
Secuenciacion20160210	7.910	6:00:06.269
Secuenciacion20160210-2	8.117	0:00:16.725
Secuenciacion20160210-3	7.910	0:01:55.567

Tabla 21: Comparación de una secuencia libre, con todas las piezas fijadas...

Como se puede observar, el resultado cuando se asigna un orden a todas las piezas incluidas las de desengrasado, empeora en:

$$8.117 - 9.1 = 0.207 h$$

Lo que significa que el procesado de las piezas tardaría 12 minutos ($0.207 \cdot 60$) más en acabar. Este aspecto es muy relevante porque a pesar de no haber alcanzado el óptimo, sino un punto cercano, se ahorran 6 horas de cálculo frente a una pérdida de 12 minutos en el procesado.

Sin embargo, y al igual que sucedió en la secuencia de 8 piezas, con la optimización que impone un orden en todas las piezas salvo las de desengrasado (secuenciacion20160210-3), el resultado obtenido es exactamente igual que el óptimo (secuenciacion20160210). La diferencia es que en vez de alcanzarlo en 6 horas lo alcanza en 2 minutos aproximadamente.

3.9 CONCLUSIONES

Tras este análisis de comportamiento se pueden sacar dos conclusiones:

- Cuando se asignan prioridades a todas las piezas se obtienen los peores resultados. En el siguiente apartado se compararán las diferencias entre asignar un orden a todas las piezas y hacerlo solo a la mitad aproximadamente.
- Las piezas de desengrasado son especialmente flexibles porque son las únicas piezas con las que se consigue llegar al óptimo cuando no se les asigna ninguna prioridad, y esto puede deberse a 2 motivos
 - Por ser las piezas con menos tareas
 - Por ser las piezas que menos aparecen en cada secuencia

4 MEJORAS EN LA RESOLUCIÓN DEL PROBLEMA

4.1 INTRODUCCIÓN

Debido a las limitaciones comentadas en el análisis del comportamiento, en este apartado se van a aplicar las mejoras comentadas en dicho análisis (capítulo 3) con el objetivo de encontrar nuevos métodos que ayuden a la resolución y comprobar su robustez. Para ello se han creado 10 casos distintos partiendo de los datos proporcionados para el desarrollo de las mejoras en la resolución, sobre los que se van a aplicar los heurísticos que se han creado aplicando un límite de tiempo de media hora para cada optimización. De esta forma se van a poder comparar los resultados a lo largo de los distintos escenarios.

A continuación se presenta la forma de proceder para crear los casos originales sobre los que se comparan los resultados así como los distintos heurísticos que se han creado y sus principales características.

4.2 CASOS ORIGINALES

En los datos proporcionados para el desarrollo de las mejoras en la resolución se encuentran los datos de 6 trabajos distintos. Del mismo modo que en la tabla 2, en la siguiente tabla se muestra un resumen de los trabajos junto con los datos de sus respectivas tareas, cubas donde se deben realizar las tareas, así como los rangos de tiempos admisibles.

Herramienta de planificación de secuencias de tratamiento químico sin esperas de piezas aeronáuticas

Piezas/cubas	N piezas	3	5	4	5	13	17	7	9	21	17	20	23	11	15	10	15	16	18	30	35	
TRABAJO_A	9	(10-15)	(5-15)	(2.5-10)	(5-10)	(1-10)	(1-90)		(1-5)													(30-60)
TRABAJO_B	3	(10-15)	(5-15)											(2.5-3)	(5-15)	(1-5)	(5-15)					(30-60)
TRABAJO_C	9	(10-15)	(5-15)			(1-10)	(1-90)		(1-5)	(27-29)	(1-90)		(5-15)									(30-90)
TRABAJO_D	2	(12-20)	(5-15)					(1-5)	(5-10)									(51-60)	(1-5)	(5-10)		(30-90)
TRABAJO_E	2	(12-20)	(5-10)			(1-5)			(5-10)	(26-27)		(1-5)	(5-10)									(30-90)
TRABAJO_F	2	(12-15)	(5-10)	(3-5)	(5-10)			(1-2)	(5-10)													(30-90)

Tabla 22: Datos del enunciado

Con el objetivo de crear distintos heurísticos y probar la robustez de cada uno de ellos, se optó por la realización de 10 casos o escenarios partiendo de los datos que se muestran en la tabla anterior. Para ello primero se crearon los 10 casos de la forma más representativa posible, es decir, intentando crear diferentes escenarios teniendo en cuenta la limitación del número de piezas de cada tipo (primera columna de la tabla anterior) y asignando 15 piezas a cada caso. A continuación se muestra la distribución de los trabajos en cada caso.

	TRABAJO_A	TRABAJO_B	TRABAJO_C	TRABAJO_D	TRABAJO_E	TRABAJO_F
Caso1	3	3	3	2	2	2
Caso2	5	3	5	2	0	0
Caso3	5	3	5	0	2	0
Caso4	5	3	5	0	0	2
Caso5	9	0	0	2	2	2
Caso6	0	0	9	2	2	2
Caso7	5	0	4	2	2	2
Caso8	5	3	4	1	1	1
Caso9	0	3	7	2	1	2
Caso10	3	0	8	0	2	2

Tabla 23: Distribución de las piezas a lo largo de los casos

- Caso1: Se busca un caso homogéneo con el mayor número posible de piezas de cada tipo, siempre que quede equilibrado
- Caso2: Influencia de las piezas tipo D
- Caso3: Influencia de las piezas tipo E
- Caso4: Influencia de las piezas tipo F
- Caso5: Influencia de las piezas tipo A
- Caso6: Influencia de las piezas tipo C
- Caso7 y caso8: influencia de las piezas tipo B
- Caso9 y caso10: aleatorios

Del mismo modo que en el análisis del comportamiento de la secuenciación, para el las mejoras en la resolución del problema se han calculado las duraciones medias agregadas de cada trabajo. Las cuales se van a utilizar como criterio de comparación para asignar prioridades entre las piezas; a pesar de que, como ya se ha mencionado, no es un parámetro que represente la realidad de los tiempos empleados.

A continuación se muestra una tabla con los tiempos agregados de las piezas en la mejora en la resolución del problema.

Trabajo	T_med
TRABAJO_B	93.25
TRABAJO_F	101.50
TRABAJO_E	131.00
TRABAJO_A	135.25
TRABAJO_D	162.50
TRABAJO_C	220.00

Tabla 24: Tiempo agregado de las piezas involucradas en las mejoras en la resolución

Una vez que se crearon los casos, fue necesario realizar la optimización de cada uno de ellos sin ninguna modificación, es decir, de los casos originales (nombre con el que se les va a designar a lo largo del proyecto). Con esto se consigue una referencia con la que comparar cada heurístico posteriormente.

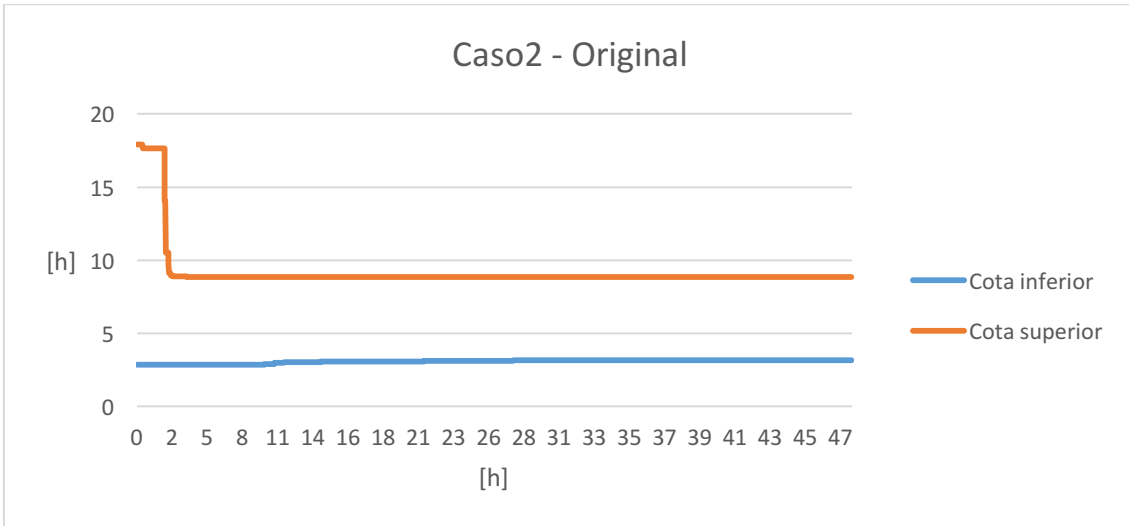
En un primer momento se dejaron corriendo los casos con un límite de 48 horas (el cual se alcanzaba antes de encontrar un punto óptimo), pero analizando los resultados se llegó a la conclusión de que no era necesario dejarlo tanto tiempo, pues transcurridas 24 horas la única modificación que tenía lugar durante la optimización era un incremento de la cota inferior.

Durante la optimización de los casos originales, transcurridas 24 horas, la columna del límite superior no solo no mejoraba sustancialmente, sino que se mantenía constante; y como se ha explicado anteriormente, el resultado final en caso de llegar al límite de tiempo es igual a la cota superior. En concreto se ejecutaron los casos 1 y 2 con límite tanto de 24 horas como de 48 horas y se obtuvieron exactamente los mismos resultados en cuanto al óptimo. En la siguiente tabla se muestra cómo en el caso2, el resultado final es igual en ambas optimizaciones. Por otro lado, como era de esperar, el valor de la cota inferior se acerca más al valor final en el caso de 48 horas, pero esto no influye en el resultado final.

Elapsed	Best possible	Final Solve
24:00:35.241	2.858333	8.858333
48:14:23.171	3.158333	8.858333

Tabla 25: Resultado de la optimización del caso2 para 24 y 48 horas

En la siguiente gráfica se muestra la variabilidad tanto de la cota inferior como de la cota superior frente al tiempo para el caso2. Como se puede observar, a partir de las 5 horas de optimización la cota superior no sufre ninguna variación. Ante estos resultados es más conveniente aceptar el límite de 24 horas en lugar del de 48 teniendo en cuenta que como mínimo se han de ejecutar 10 casos originales.



Gráfica 2: Representación de las cotas superior e inferior en el caso2 a lo largo de la optimización

Sin embargo, podría existir la posibilidad de que aun obteniendo el mismo valor de la optimización, el orden de la secuencia final fuese diferente. Por ello se crearon los diagramas de Gantt de ambas secuencias y se comprobó que eran exactamente iguales tanto con el límite de 24h como con el límite de 48h.

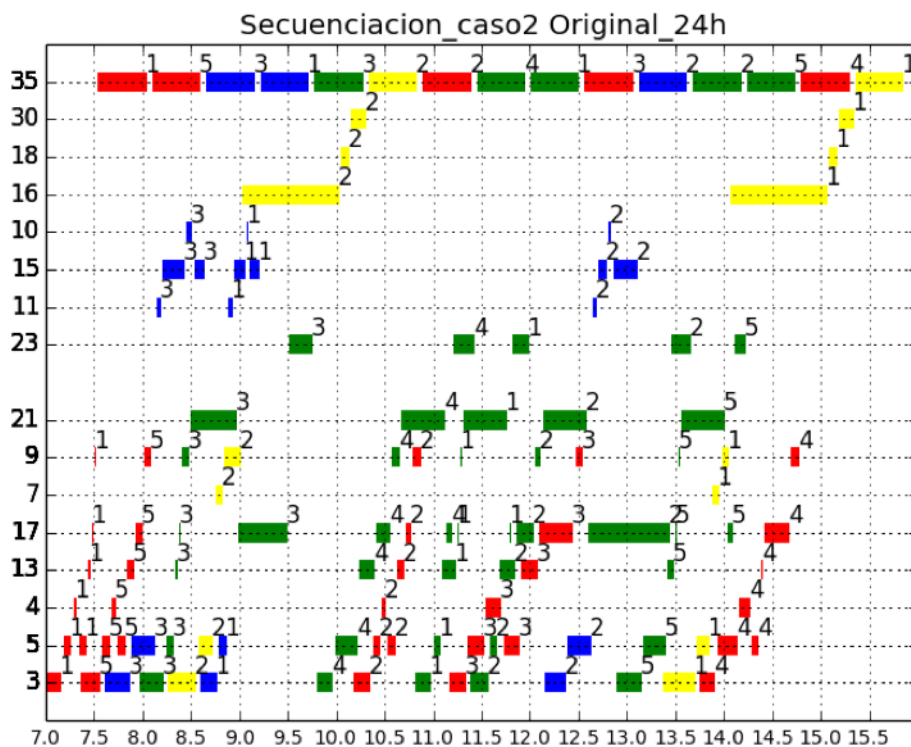


Imagen 21: Resultado del caso2 para 24h

Por último, y para demostrar que el resultado en ambas optimizaciones fue exactamente igual, se calcularon los tiempos de procesamiento agregado de cada pieza en las dos optimizaciones, usando el script de Python ya mencionado.

A continuación se muestran los resultados de las optimizaciones de todos los casos originales, que se supondrán como óptimos para la posterior comparación con los heurísticos.

Caso	Final Solve	Best possible	Gap	Gap %	Elapsed
Caso1	8.858333	2.8583	6.000	67.73%	24:01:14.778
Caso2	8.858333	2.8583	6.000	67.73%	24:00:35.241
Caso3	8.858333	2.6676	6.191	69.90%	48:04:26.121
Caso4	8.858333	3.1583	5.700	64.35%	24:01:05.513
Caso5	8.858333	2.9083	5.950	67.17%	25:00:52.968
Caso6	9.133333	2.8583	6.275	68.70%	24:00:25.397
Caso7	8.858333	2.8583	6.000	67.73%	24:00:28.170
Caso8	8.858333	2.6000	6.258	70.65%	24:00:19.745
Caso9	8.891667	2.8631	6.029	67.80%	24:01:42.445
Caso10	8.858333	2.6500	6.208	70.08%	24:00:44.055

Tabla 26: Resultados de la optimización de los casos originales

Lo llamativo de los resultados que se muestran en la tabla es que en todos los casos salvo en dos se obtienen los mismos resultados, que como ya se explicó, es el valor de la cota superior y que se supone como valor final en caso de que se alcance el límite de tiempo.

Por otro lado, es cierto que el valor del *gap* es elevado. Pero como ya se ha explicado, la única opción de que se reduzca consiste en un aumento de la cota inferior, pues en la gráfica 2 se ha visto cómo la cota superior se mantenía constante durante más de 24 horas.

4.3 HEURISTICOS

Los heurísticos que se describen a continuación se han centrado en los siguientes aspectos:

- **Influencia del parámetro duración media agregada:** Como ya se ha explicado, a cada pieza se le asigna un tiempo igual a la suma de las medias de los rangos de las tareas que componen dicha pieza.
- **Influencia del parámetro amplitud agregada:** A cada pieza se le asigna un tiempo igual a la suma de las amplitudes de las tareas que componen dicha pieza.
- **Influencia de la exclusividad/no exclusividad de las piezas:** Se entiende por no exclusivas aquellas piezas cuyas tareas se realicen en cubas que sean comunes a las demás piezas de otra familia del mismo caso; y por

exclusivas, lo contrario. A continuación se muestra una imagen que ayude a entender este criterio.

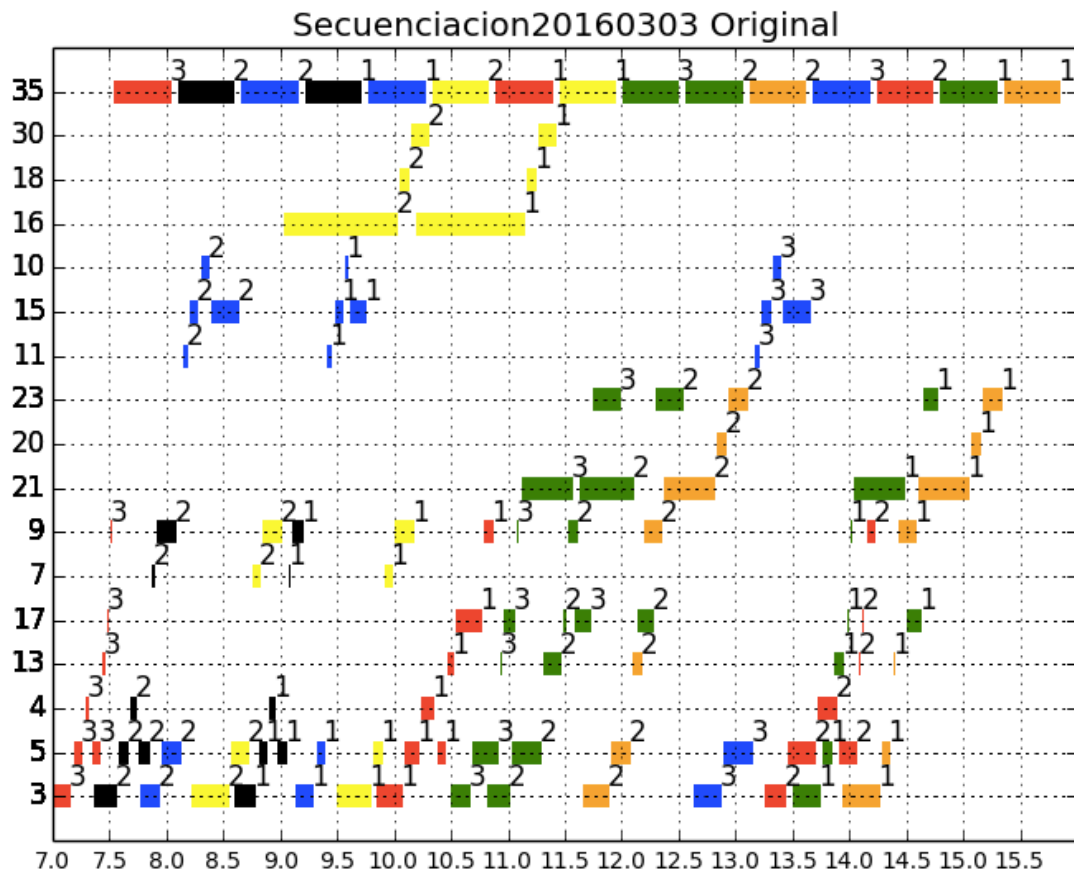


Imagen 22: Distribución de la secuencia original del caso1

La imagen pertenece a la secuencia original del caso1. En ella se puede observar cómo las piezas rojas (A), negras (F) y verdes (C) realizan todas sus tareas en cubas que como mínimo coinciden con otro trabajo; a diferencia del resto de piezas, por ejemplo, las amarillas son las únicas que emplean las cubas 16, 18 y 30.

- **Influencia de la repetitividad de las piezas:** Se asignarán prioridades a aquellas piezas que aparezcan más veces en cada caso.
- **Influencia de la aleatoriedad**
- **Influencia de la asignación de cubas equivalentes**

Basándose en estos aspectos, primero se seleccionan las piezas sobre las que se asigna prioridad y a continuación, basándose en los parámetros de tiempo, se asignan prioridades entre las distintas piezas que se han seleccionado anteriormente.

Es preciso mencionar que, igual que en el en el análisis del comportamiento de la secuenciación, la mayoría de los heurísticos se han realizado sin habilitar el uso de cubas equivalentes. En concreto, tan solo se ha habilitado en los heurísticos H18 y H19 con el objetivo de analizar su impacto sobre los heurísticos analizados.

4.3.1 H1 y H2. Duración agregada sobre el total de las piezas

Es necesario recordar que cuando se asignan prioridades entre las distintas piezas, se hace asignando un valor a las variables que modelan el problema.

H1: Orden creciente

Este heurístico ordena todos los trabajos de **menor a mayor** duración agregada de las tareas

- Caso1: Azul (B), negro (F), naranja (E), rojo (A), amarillo (D), verde (C)
- Caso2: Azul (B), rojo (A), amarillo (D), verde (C)
- Caso3: Azul (B), naranja (E), rojo (A), verde (C)
- Caso4: Azul (B), negro (F), rojo (A), verde (C)
- Caso5: Negro (F), naranja (E), rojo (A), amarillo (D),
- Caso6: Negro (F), naranja (E), amarillo (D), verde (C)
- Caso7: Negro (F), naranja (E), rojo (A), amarillo (D), verde (C)
- Caso8: Azul (B), negro (F), naranja (E), rojo (A), amarillo (D), verde (C)
- Caso9: Azul (B), negro (F), naranja (E), amarillo (D), verde (C)
- Caso10: Negro (F), naranja (E), rojo (A), verde (C)

A continuación y para cada heurístico, se muestra una tabla de resultados donde se recogen tanto el error (en porcentaje) del heurístico para cada caso respecto al óptimo calculado, como la diferencia en minutos (cuánto empeora cada caso), también respecto al óptimo.

H1	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	13.55	12.14	4.80	4.80	14.11	6.93	14.30	7.62	8.53	9.22	9.60
Diferencia min	72	65	26	26	75	38	76	41	45	49	51

Tabla 27: Resultados del heurístico H1 para cada caso

H2: Orden decreciente

Siguiendo la línea de H1, en este caso se ordenan todas las piezas de **mayor a menor** duración agregada de las tareas.

- Caso1: Verde (C), amarillo (D), rojo (A), naranja (E), negro (F), azul (B)
- Caso2: Verde (C), amarillo (D), rojo (A), azul (B)
- Caso3: Verde (C), rojo (A), naranja (E), azul (B)
- Caso4: Verde (C), rojo (A), negro (F), azul (B)
- Caso5: Amarillo (D), rojo (A), naranja (E), negro (F)
- Caso6: Verde (C), amarillo (D), naranja (E), negro (F)
- Caso7: Verde (C), amarillo (D), rojo (A), naranja (E), negro (F)
- Caso8: Verde (C), amarillo (D), rojo (A), naranja (E), negro (F), azul (B)
- Caso9: Verde (C), amarillo (D), naranja (E), negro (F), azul (B)
- Caso10: Verde (C), rojo (A), naranja (E), negro (F)

H2	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	10.25	9.13	9.97	7.71	12.51	11.59	10.82	9.41	13.50	11.67	10.66
Diferencia min	55	49	53	41	67	64	58	50	72	62	57

Tabla 28: Resultados del heurístico H2 para cada caso

4.3.2 H3, H4, H5 y H6 No exclusividad de las tareas

En los siguientes heurísticos se va a realizar una asignación de prioridades basada en la no exclusividad de las tareas, que como ya se sabe, consiste en asignar prioridades a aquellas piezas que no sean exclusivas, es decir, a aquellas piezas cuyas tareas se realicen en cubas que sean comunes a las demás piezas de otra familia del mismo caso.

H3: Orden creciente

De forma similar a H1, en este caso se ordenan las piezas de **menor a mayor**, pero solo se le asignan prioridades a aquellas piezas que no sean exclusivas. Resultados:

- Caso1: Negro (F), rojo (A), verde (C)
- Caso2: Ninguna
- Caso3: Verde (C)
- Caso4: Rojo (A)

- Caso5: Negro (F)
- Caso6: Ninguna
- Caso7: Negro (F), rojo (A), verde (C)
- Caso8: Negro (F), rojo (A), verde (C)
- Caso9: Ninguna
- Caso10: Rojo (A) y verde (C)

Como se puede observar, no todos los casos tienen piezas que no sean exclusivas, de modo que este heurístico no se puede aplicar a todos los casos. Además, los casos en los que solo hay una pieza no exclusiva no se le puede asignar prioridad frente a ninguna pieza. Sin embargo, como se comentará más adelante, a lo largo del desarrollo de este proyecto se ha descubierto que aun asignando prioridades entre las piezas de un mismo tipo se pueden obtener buenos resultados. En concreto en la siguiente tabla se puede ver cómo en el caso4 se ha obtenido un error de tan solo 0.38% siendo un caso en el que solo se le ha asignado prioridad a las piezas del tipo A.

H3	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	0.38	- *	15.80	0.38	5.64	- *	0.85	0.85	-*	4.23	4.02
Diferencia min	2	- *	84	2	30	- *	4	4	-*	23	21

Tabla 29: Resultados del heurístico H3 para cada caso

* Casos donde no procede aplicar el heurístico

H4: Orden creciente menos restrictivo

Aquí se ha intentado crear un heurístico que suponga la evolución del heurístico H3 adoptando una postura menos restrictiva para solventar el problema de H3. Por ello, partiendo del conjunto de piezas a las que se les asignaba prioridad en H3, se les añade aquellas piezas que tengan como máximo una tarea exclusiva. Con esto se consigue una aplicación en el 100% de los casos obteniendo unos resultados muy buenos también, con una media de error del 1.83%; que en este caso es más representativo al poder aplicarlo sobre más casos

Para entender este criterio mejor, observando la imagen anterior se puede ver cómo los trabajos naranjas solo tienen una tarea exclusiva, que es la que se realiza en la cuba 20; a diferencia de las piezas amarillas, que como se mencionó anteriormente, tienen más de una tarea exclusiva. Los resultados son los siguientes ordenados de **menor a mayor**:

- Caso1: Negro (F), naranja (E), rojo (A), verde (C)
- Caso2: Rojo (A)
- Caso3: Naranja (E), rojo (A), verde (C)
- Caso4: Negro (F), rojo (A)
- Caso5: Negro (F), rojo (A)
- Caso6: Negro (F), naranja (E), verde (C)
- Caso7: Negro (F), naranja (E), rojo (A), verde (C)
- Caso8: Negro (F), naranja (E), rojo (A), verde (C)
- Caso9: Negro (F), naranja (E), verde (C)
- Caso10: Negro (F), naranja (E), rojo (A), verde (C)

H4	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	0.85	2.45	1.22	0.38	0.85	0.73	1.88	0.38	0.94	9.22	1.89
Diferencia min	4	13	7	2	4	4	10	2	5	49	10

Tabla 30: Resultados del heurístico H4 para cada caso

H5: Orden decreciente

Igual que H3, es decir, asignando prioridades a las piezas que no sean exclusivas; pero ordenando de **mayor a menor**. Resultados:

- Caso1: Verde (C), rojo (A) y negro (F)
- Caso2: Ninguna
- Caso3: Verde (C)
- Caso4: Rojo (A)
- Caso5: Negro (F)
- Caso6: Ninguna
- Caso7: Verde (C), rojo (A), negro (F)
- Caso8: Verde (C), rojo (A), negro (F)
- Caso9: Ninguna
- Caso10: Verde (C), rojo (A)

H5	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	6.30	- *	15.80	0.56	7.62	- *	5.64	0.38	-*	3.39	5.67
Diferencia min	34	- *	84	3	41	- *	30	2	-*	18	30

Tabla 31: Resultados del heurístico H5 para cada caso

* Casos donde no procede aplicar el heurístico

H6: Orden decreciente menos restrictivo

De la misma forma que sucede con H3, H5 no tiene una aplicación sobre el 100% de los casos. De modo que este heurístico se vuelve a basar en H5 (de mayor a menor) pero siendo menos restrictivo. Resultados:

- Caso1: Verde (C), rojo (A), naranja (E), negro (F)
- Caso2: Rojo (A)
- Caso3: Verde (C), rojo (A), naranja (E)
- Caso4: Rojo (A), negro (F)
- Caso5: Rojo (A), negro (F)
- Caso6: Verde, naranja y negro
- Caso7: Verde (C), rojo (A), naranja (E), negro (F)
- Caso8: Verde (C), rojo (A), naranja (E), negro (F)
- Caso9: Verde (C), naranja (E), negro (F)
- Caso10: Verde (C), rojo (A), naranja (E), negro (F)

H6	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	0.38	16.18	1.22	14.58	0.75	5.02	6.02	1.22	10.87	11.67	5.75
Diferencia min	2	86	7	78	4	28	32	7	58	62	31

Tabla 32: Resultados del heurístico H6 para cada caso

4.3.3 H7, H8, H9 y H10. Exclusividad de las tareas

Para comparar la influencia de la ordenación de piezas no exclusivas frente a las exclusivas, en estos heurísticos se van a ordenar las piezas asignando prioridades a aquellas piezas a las que no se les asignó en la anterior batería de heurísticos. Es decir, se van a ordenar las piezas que posean al menos una tarea exclusiva.

H7: Orden creciente

Los resultados de este heurístico se compararán con los de H3 ya que se asignan prioridades a las piezas a las que no se les asignó en dicho heurístico.

- Caso1: Azul (B), naranja (E), amarillo (D)
- Caso2: Azul (B), rojo (A), amarillo (D), verde (C)
- Caso3: Azul (B), naranja (E), rojo (A)
- Caso4: Azul (B), negro (F), verde (C)

- Caso5: Naranja (E), rojo (A), amarillo (D)
- Caso6: Negro (F), naranja (E), amarillo (D), verde (C)
- Caso7: Naranja (E), amarillo (D)
- Caso8: Azul (B), naranja (E), amarillo (D)
- Caso9: Azul (B), negro (F), naranja (E), amarillo (D), verde (C)
- Caso10: Negro (F), naranja (E)

H7	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	2.54	12.14	0.38	7.06	6.49	6.93	-*	-*	8.53	-*	6.29
Diferencia min	14	65	2	38	35	38	-*	-*	45	-*	34

Tabla 33: Resultados del heurístico H7 para cada caso

* Casos donde no se ha llegado a ninguna solución

H8: Orden creciente más restrictivo

Este heurístico da prioridad a aquellas piezas a las que no se les asignó ninguna prioridad en H4. Al contrario de lo que sucedió anteriormente, en H7 se le asigna prioridad a demasiadas piezas y esto puede empeorar los resultados según se comentó en el apartado de análisis del comportamiento. Tanto que como se verá mas adelante, en 3 casos no se llega a ninguna solución. De modo que en este heurístico se van a ordenar de **menor** a **mayor** aquellas piezas que posean más de una tarea exclusiva.

- Caso1: Azul (B), amarillo (D)
- Caso2: Azul (B), amarillo (D), verde (C)
- Caso3: Azul (B)
- Caso4: Azul (B), verde (C)
- Caso5: Naranja (E), Amarillo (D)
- Caso6: Amarillo (D)
- Caso7: Amarillo (D)
- Caso8: Azul (B), amarillo (D)
- Caso9: Azul (B), amarillo (D)
- Caso10: Ninguna

H8	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	7.90	0	9.88	0	2.26	10.31	7.34	3.48	89.69	-*	14.54
Diferencia min	42	0	53	0	12	57	39	19	479	-*	78

Tabla 34: Resultados del heurístico H8 para cada caso

* Casos donde no procede aplicar el heurístico

H9: Orden decreciente

Con este heurístico vamos a comparar los resultados de H5 ya que damos prioridad a aquellas piezas a las que no se le dio en el mencionado heurístico. Se ordenan las piezas que posean al menos una tarea exclusiva, resultando igual que H7 pero de **mayor a menor** duración agregada

- Caso1: Amarillo (D), Naranja (E), Azul (B)
- Caso2: Verde (C), Amarillo (D), Rojo (A), Azul (B),
- Caso3: Rojo (A), Naranja (E), Azul (B)
- Caso4: Verde (C), Negro (F), Azul (B),
- Caso5: Amarillo (D), Rojo (A), Naranja (E),
- Caso6: Verde (C), Amarillo (D), Naranja (E), Negro (F),
- Caso7: Amarillo (D), Naranja (E),
- Caso8: Amarillo (D), Naranja (E), Azul (B)
- Caso9: Verde (C), Amarillo (D), Naranja (E), Negro (F), Azul (B)
- Caso10: Naranja (E), Negro (F)

H9	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	1.32	9.13	0	0	4.14	11.59	107.62	0.94	13.50	16.56	16.48
Diferencia min	7	49	0	0	22	64	572	5	72	88	88

Tabla 35: Resultados del heurístico H9 para cada caso

H10: Orden decreciente menos restrictivo

Una vez más, se asignan prioridades a las piezas que posean más de una tarea exclusiva, resultando igual que H8 pero ordenando las piezas de **mayor a menor** duración agregada

- Caso1: Amarillo (D), Azul (B)
- Caso2: Verde (C), Amarillo (D), Azul (B),
- Caso3: Azul (B)
- Caso4: Verde (C), Azul (B),
- Caso5: Amarillo (D), Naranja (E),
- Caso6: Amarillo (D)
- Caso7: Amarillo (D)
- Caso8: Amarillo (D), Azul (B)
- Caso9: Amarillo (D), Azul (B)

- Caso10: Ninguna

H10	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	1.03	3.39	9.88	0	3.29	10.31	5.93	2.82	5.62	0	4.70
Diferencia min	6	18	53	0	18	57	32	15	30	0	25

Tabla 36: Resultados del heurístico H10 para cada caso

4.3.4 H11 y H12. Duración agregada sobre la mitad de las piezas

A lo largo del proyecto se ha observado que cuando se asignan prioridades a demasiadas piezas el resultado empeora, de modo que en este caso se van a realizar dos heurísticos basados en H1 y H2, pero donde se asignen prioridades a la mitad de las piezas. El objetivo es llegar a 7 piezas empleando distintas familias: de forma que si utilizando la primera pieza se sobrepasan las 7, habría que seleccionar una pieza más; pero si por el contrario, se seleccionan las dos primeras piezas y no se llega a 7, pero seleccionando la tercera sí se sobrepasa la cifra de 7, tan solo habría que quedarse con las 2 primeras piezas.

H11: Orden creciente

- Caso1: Azul (B), negro (F), naranja (E)
- Caso2: Azul (B), rojo (A)
- Caso3: Azul (B), naranja (E)
- Caso4: Azul (B), negro (F)
- Caso5: Negro (F), naranja (E)
- Caso6: Negro (F), naranja (E), amarillo (D)
- Caso7: Negro (F), naranja (E)
- Caso8: Azul (B), negro (F), naranja (E)
- Caso9: Azul (B), negro (F), naranja (E)
- Caso10: Negro (F), naranja (E), rojo (A)

H11	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	0	3.95	25.87	0.85	1.32	- *	5.17	-*	0	-*	5.31
Diferencia min	0	21	138	4	7	- *	28	-*	0	-*	28

Tabla 37: Resultados del heurístico H11 para cada caso

- * Casos donde no se ha llegado a ninguna solución

H12: Orden decreciente

Este heurístico sigue la línea de H11 pero ordenando las piezas de **mayor a menor** duración agregada

- Caso1: Naranja (E) , negro (F), azul (B)
- Caso2: Rojo (A), azul (B)
- Caso3: Naranja (E), azul (B)
- Caso4: Negro (F), azul (B)
- Caso5: Naranja (E), negro (F)
- Caso6: Amarillo (D) , naranja (E), negro (F)
- Caso7: Naranja (E), negro (F)
- Caso8: Naranja (E), negro (F), azul (B)
- Caso9: Naranja (E), negro (F), azul (B)
- Caso10: Rojo (A), naranja (E), negro (F)

H12	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	1.69	7.34	0.38	0.38	1.60	3.83	0	1.03	1.97	1.13	1.93
Diferencia min	9	39	2	2	9	21	0	6	11	6	10

Tabla 38: Resultados del heurístico H12 para cada caso

4.3.5 H13 y H14. Amplitud agregada sobre el total de las piezas

Analizando los datos de las piezas se llegó a la conclusión de la posibilidad de aplicar otro parámetro como método de comparación de los distintos trabajos. Se trata de la amplitud de tiempos agregada de cada pieza.

Duracion	B	F	E	A	D	C
Amplitud	B	F	E	D	A	C

Tabla 39: Secuencia creciente para el caso de amplitud agregada y duración media agregada

Como se puede observar, la única diferencia entre los dos criterios reside en la permutación de los trabajos A y D.

H13: Orden creciente

En este heurístico se van a ordenar todas las piezas en orden creciente bajo el criterio de amplitud agregada con el objetivo de poder comparar más adelante el resultado con el del H1. Sin embargo, dado que la única diferencia reside en el cambio de A por D, este heurístico solo va a tener sentido aplicarlo en aquellos casos en los que aparezcan ambas piezas, es decir: 1, 2, 5, 7 y 8. Porque en el

resto se obtendrían los mismos resultados que en H1 (y H2 para el heurístico H14)

- Caso1: Azul (B), negro (F), naranja (E) , amarillo (D), rojo (A), verde (C)
- Caso2: Azul (B), amarillo (D), rojo (A), verde (C)
- Caso3: Igual que H1
- Caso4: Igual que H1
- Caso5: Negro (F), naranja (E), amarillo (D), rojo (A)
- Caso6: Igual que H1
- Caso7: Negro (F), naranja (E), amarillo (D), rojo (A), verde (C)
- Caso8: Azul (B), negro (F), naranja (E), amarillo (D), rojo (A), verde (C)
- Caso9: Igual que H1
- Csao10: Igual que H1

H13	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	6.68	7.71	4.80	4.80	3.86	6.93	7.71	6.49	8.53	9.22	6.67
Diferencia min	36	41	26	26	21	38	41	35	45	49	36

Tabla 40: Resultados del heurístico H13 para cada caso

H14: Orden decreciente

De igual forma que en el anterior, en este caso se van a ordenar todas las piezas en orden decreciente bajo el criterio de amplitud agregada para poder comparar los resultados con los de H2.

- Caso1: Verde (C), amarillo (D), rojo (A), naranja (E), negro (F), azul (B)
- Caso2: Verde (C), amarillo (D), rojo (A), azul (B)
- Caso3: Igual que H2
- Caso4: Igual que H2
- Caso5: Amarillo (D), rojo (A), naranja (E), negro (F)
- Caso6: Igual que H2
- Caso7: Verde (C), amarillo (D), rojo (A), naranja (E), negro (F)
- Caso8: Verde (C), amarillo (D), rojo (A), naranja (E), negro (F), azul (B)
- Caso9: Igual que H2
- Caso10: Igual que H2

H14	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	17.50	18.63	9.97	7.71	10.91	11.59	18.16	14.11	13.50	11.67	13.37
Diferencia min	93	99	53	41	58	64	97	75	72	62	71

Tabla 41: Resultados del heurístico H14 para cada caso

4.3.6 H15. Aleatoriedad

Este heurístico realiza una asignación de prioridades de forma totalmente aleatoria sobre todas las piezas. Se ha desarrollado este heurístico para poder asegurar que el resto de heurísticos tienen sentido y que se obtienen por lo general mejores resultados que de forma aleatoria.

H15	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	10.82	9.31	8.47	6.40	16.46	6.11	16.27	7.53	11.81	11.19	10.44
Diferencia min	58	50	45	34	88	34	87	40	63	60	56

Tabla 42: Resultados del heurístico H15 para cada caso

4.3.7 H16 y H17. Repetitividad de las piezas

Basándose en los resultados del análisis del comportamiento de la secuenciación, aquí se pretende realizar un heurístico que diese prioridad a aquellas piezas que tuviesen un elevado número de tareas. Sin embargo, en los casos de la mejora de la resolución, todas las piezas tienen 7, 8 o 9 tareas, de modo que no tendría sentido diferenciar las piezas bajo este criterio. En su lugar, se ha desarrollado un heurístico que da prioridad a las piezas que más se repiten.

H16: Orden creciente

Según lo comentado, en este heurístico en cada caso se ha dado prioridad a todas aquellas piezas que estuviesen repetidas más de 2 veces (es decir, 3 o más), y se han ordenado de **menor a mayor** duración agregada.

- Caso1: Azul (B), rojo (A) , verde (C)
- Caso2: Azul (B), rojo (A) , verde (C)
- Caso3: Azul (B), rojo (A) , verde (C)
- Caso4: Azul (B), rojo (A) , verde (C)
- Caso5: Rojo (A)
- Caso6: Verde (C)
- Caso7: Rojo (A) , verde (C)
- Caso8: Azul (B), rojo (A) , verde (C)
- Caso9: Azul (B), verde (C)
- Caso10: Rojo (A) , verde (C)

En este heurístico se ha descubierto que, en contra de lo que se pensaba, cuando se tiene que aplicar un heurístico a un caso en el que solo es necesario asignar prioridad a una pieza, sí es útil realizar la optimización. Al principio se dejaban dichos casos en blanco alegando que el heurístico no era de aplicación. Sin embargo, en H16_5 se ha aplicado la ordenación de las piezas tipo A para hacer la prueba: los resultados no solo son mejores que dejándolo libre, sino que como se ve en la siguiente tabla, se llega al óptimo en el caso5

H16	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	0.38	0.38	0.38	0.38	0	3.10	0.09	0.38	0	0	0.51
Diferencia min	2	2	2	2	0	17	1	2	0	0	3

Tabla 43: Resultados del heurístico H16 para cada caso

Se podría llegar a pensar que en el caso5 original tal vez se llegase al óptimo en media hora, pero echando un vistazo al registro de GAMS de dicho caso original (imagen17), se puede ver cómo esto no sucede, sino que la cota superior (resultado de la optimización en caso de llegar al límite de tiempo) tenía un valor de 10.65833 mientras que el óptimo tiene un valor de 8.853333. Este descubrimiento hizo que se tuviesen que realizar las optimizaciones que se habían descartado anteriormente.

912	554	2.85833	11	1173	10.65833	2.85833	73.2%	327	1285s
914	555	4.88333	35	549	10.65833	2.85833	73.2%	327	1310s
915	556	7.54167	75	915	10.65833	2.85833	73.2%	326	1321s
916	557	4.89167	50	374	10.65833	2.85833	73.2%	326	1446s
917	557	4.18333	12	554	10.65833	2.85833	73.2%	325	1472s
918	558	3.41667	18	545	10.65833	2.85833	73.2%	325	1496s
919	559	4.37500	25	658	10.65833	2.85833	73.2%	325	1507s
920	559	9.62500	144	650	10.65833	2.85833	73.2%	324	1527s
921	560	2.85833	8	693	10.65833	2.85833	73.2%	324	1585s
922	561	4.11667	18	680	10.65833	2.85833	73.2%	324	1591s
923	561	10.25833	218	680	10.65833	2.85833	73.2%	323	1617s
924	564	2.85833	14	668	10.65833	2.85833	73.2%	388	1692s
925	565	2.85833	14	737	10.65833	2.85833	73.2%	390	1708s
926	565	2.85833	15	1056	10.65833	2.85833	73.2%	393	1732s
927	566	2.85833	15	895	10.65833	2.85833	73.2%	395	1738s
928	567	2.85833	16	976	10.65833	2.85833	73.2%	395	1741s
930	566	2.85833	17	980	10.65833	2.85833	73.2%	396	1746s
931	570	2.85833	17	911	10.65833	2.85833	73.2%	397	1752s
934	578	2.85833	18	906	10.65833	2.85833	73.2%	401	1782s
974	596	3.62500	26	520	10.65833	2.85833	73.2%	414	2224s
977	592	3.45000	27	501	10.65833	2.85833	73.2%	415	2441s

Imagen 23: Registro de la optimización del caso5 original

Como se puede observar, al cabo de media hora o 1800 segundos (recuadro de la derecha) el valor de la cota superior (recuadro de la izquierda) era de 10.65833.

H17: Orden decreciente

Según lo comentado, en este heurístico en cada caso se ha dado prioridad a todas aquellas piezas que estuviesen repetidas más de 2 veces (es decir, 3 o más), y se han ordenado de **mayor a menor** duración agregada.

- Caso1: Verde (C), rojo (A), azul (B)
- Caso2: Verde (C), rojo (A), azul (B)
- Caso3: Verde (C), rojo (A), azul (B)
- Caso4: Verde (C), rojo (A), azul (B)
- Caso5: Rojo (A) (mismo resultado que H16)
- Caso6: Verde (C) (mismo resultado que H16)
- Caso7: Verde (C), rojo (A)
- Caso8: Verde (C), rojo (A), azul (B)
- Caso9: Verde (C), azul (B)
- Caso10: Verde (C), rojo (A)

H17	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	3.20	6.02	5.17	1.98	0	3.10	3.20	3.20	3.09	1.13	3.01
Diferencia min	17	32	28	11	0	17	17	17	17	6	16

Tabla 44: Resultados del heurístico H17 para cada caso

4.3.8 H18 y H19. Cubas equivalentes

H18: Repetitividad de las piezas en orden creciente y con cubas equivalentes

En este heurístico se van a aplicar las mismas prioridades que en H16 pero esta vez habilitando el uso de las cubas equivalentes con el objetivo de analizar la influencia de estas.

H19: Amplitud agregada sobre el total de las piezas en orden decreciente

De igual forma que el heurístico anterior toma las prioridades del H16, con este se van a tomar las prioridades del heurístico en el que se han obtenido los peores resultados siempre y cuando se pueda aplicar en el 100% de los casos o se hayan obtenido resultados en el 100% de ellos.

Como se comentará más adelante, la peor media se ha obtenido con H9. Pero en este heurístico existe un *outlier* que dispara la media, de forma que si no se tuviese en cuenta este valor, H9 no sería el heurístico con la peor media de todas; la siguiente peor media se obtiene con H8, pero no permite una aplicación sobre el 100% de los resultados; teniendo en cuenta lo anterior, los peores resultados se obtienen con H14, razón por la cual H19 va a aplicar las mismas prioridades pero habilitando el uso de cubas equivalentes.

4.3.9 H20. Priorización entre las piezas del mismo tipo

Analizando los buenos resultados obtenidos en H16, se llegó a la conclusión de la necesidad de crear un método de asignación de prioridad entre las piezas del mismo tipo. Es decir, si se tienen 3 piezas del tipo A y 2 del tipo B, consiste en asignar prioridades dentro de A y dentro de B de forma que no se asigne ninguna prioridad de A frente a B. El resultado sería:

- Trabajo_A_1 - Trabajo_A_2 –Trabajo_A_3
- Trabajo_B_1 – Trabajo_B_2

H20	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media
Error %	5.36	2.63	0.00	0.00	0.00	1.28	4.99	2.73	1.59	0.85	1.94
Diferencia min	29	14	0	0	0	7	27	15	8	4	10

Tabla 45: Resultados del heurístico H20 para cada caso

5 RESULTADOS Y CONCLUSIONES

A lo largo del desarrollo de los heurísticos, ha habido algunos casos en los que no ha sido posible aplicar ciertos heurísticos o incluso no se ha llegado a un resultado. En la comparación de los resultados, estos casos se van a suponer como peores frente a aquellos que sí han obtenido una solución.

5.1 Orden creciente vs decreciente con criterio de tiempo agregado

A continuación se van a comparar los resultados de los heurísticos que se han desarrollado tanto en orden creciente como en orden decreciente según el tiempo agregado de procesado. En la siguientes tablas:

- La columna **N** representa el número de casos en los que se obtienen mejores resultados
- **DT** es la desviación típica
- Para cada caso, se resaltan en verde aquél heurístico que proporcione mejores resultados

Cuando la desviación típica es pequeña, significa que la media es representativa porque todos los valores son cercanos a la media. De modo que si se obtiene una media elevada y con desviación típica pequeña, el heurístico se puede asumir como malo; por el contrario, si la media es pequeña y la desviación típica también lo es, se puede asumir el heurístico como bueno. Esto siempre y cuando el heurístico tenga una aplicación sobre el 100% de los casos. En aquellos heurísticos en los que se obtenga una desviación típica elevada, será necesario analizar si esto se debe a un *outlier* o a que realmente los resultados son muy dispersos.

5.1.1 H1 vs H2. Total de las piezas

A continuación se muestra la comparación entre una ordenación creciente (H1) frente a una ordenación decreciente (H2). Como se puede observar, en 6 de los 10 casos se obtienen mejores resultados con H1 frente a 4 y la media de este heurístico es mejor que la de H2. No obstante es cierto que la desviación típica

de H2 es más pequeña que la de H1, pero en cualquier caso ambas son pequeñas.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H1	13.55	12.14	4.80	4.80	14.11	6.93	14.30	7.62	8.53	9.22	9.60	3.70	6
H2	10.25	9.13	9.97	7.71	12.51	11.59	10.82	9.41	13.50	11.67	10.66	1.72	4

Tabla 46: Comparación H1 vs H2

Es preciso indicar que los resultados de ambos heurísticos son muy malos. Ya que la media de los errores en minutos se traduce en una media de error de 51 minutos para H1 y 57 minutos para H2, lo que significa que la secuencia se retrasaría una hora respecto al óptimo.

5.1.2 H3 vs H5. No exclusividad

En la siguiente tabla se muestra la comparación entre H3 (creciente) y H5 (decreciente). En estos heurísticos se han asignado prioridades a aquellas piezas sin exclusividad, pero como se comentó anteriormente, el problema de estos dos heurísticos es que no tiene una aplicación sobre el 100% de los casos sino que existen 3 casos sobre los que no se puede aplicar. En este caso, H3 consigue mejores resultados en 4 casos frente a los 2 que consigue H5; porque como se puede observar, a parte de los casos que no son de aplicación, en el caso3 se obtienen los mismos resultados con los 2 heurísticos.

Si solo se le diese importancia a la media obtenida, ambos heurísticos podrían considerarse buenos, además la elevada desviación típica (en minutos 30 min para H3 y 28 min para H5) de los dos heurísticos se debe al resultado del caso3, que podría considerarse como un *outlier*. Pero hay que tener en cuenta dos cosas:

- Solo es de aplicación en el 70% de los casos
- La comparación solo es posible en el 60% de los casos, de modo que las conclusiones que se saquen de esta comparación no pueden constituir parte de los fundamentos de las conclusiones finales; en todo caso podrían apoyarlas

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H3	0.38	-	15.80	0.38	5.64	-	0.85	0.85	-	4.23	4.02	5.60	4
H5	6.30	-	15.80	0.56	7.62	-	5.64	0.38	-	3.39	5.67	5.26	2

Tabla 47: Comparación H3 vs H5

5.1.3 H4 vs H6. No exclusividad menos restrictiva

Como ya se sabe, el objetivo de H4 (creciente) y H6 (decreciente) es dar solución a la escasa aplicabilidad de los heurísticos H3 y H5 creando un heurístico menos restrictivo que los anteriores y que siga dando prioridad a las piezas que no poseen tareas exclusivas. Esto se puede ver reflejado en la siguiente tabla, donde se observa que se ha conseguido una aplicación sobre el 100% de los casos.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H4	0.85	2.45	1.22	0.38	0.85	0.73	1.88	0.38	0.94	9.22	1.89	2.66	7
H6	0.38	16.18	1.22	14.58	0.75	5.02	6.02	1.22	10.87	11.67	5.75	6.18	2

Tabla 48: Comparación H4 vs H6

Una vez más se ha obtenido un mayor número de casos con mejores resultados en el heurístico de asignación en orden creciente (7 frente a 2). Además, con H4 se ha obtenido una media muy buena acompañada de una desviación típica reducida (14 minutos), haciendo que éste sea uno de los mejores heurísticos que se ha analizado.

5.1.4 H7 vs H9. Exclusividad

A diferencia de los heurísticos 3, 4, 5 y 6, con los heurísticos 7, 8, 9 y 10 se pretende dar prioridad a las piezas con tareas exclusivas.

En contra de lo que venía sucediendo hasta ahora, comparando H7 (creciente) frente a H9 (decreciente) se obtiene un mayor número de casos con mejores resultados en el heurístico con asignación en orden decreciente (8 frente a 2). Además, a diferencia de lo que se ha mostrado hasta ahora, en H7 existen 3 casos en los que no se ha llegado a ninguna solución (hasta ahora los casos sin resultados se debían a que los heurísticos no eran de aplicación).

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H7	2.54	12.14	0.38	7.06	6.49	6.93	-	-	8.53	-	6.29	3.86	2
H9	1.32	9.13	0.00	0.00	4.14	11.59	107.62	0.94	13.50	16.56	16.48	32.59	8

Tabla 49: Comparación H7 vs H9

A pesar de que N es mucho mayor en H9, tanto la media obtenida como la desviación típica son excesivamente elevadas debido al resultado obtenido en el caso7, que sin duda se trata de un *outlier*. Sin embargo, si se sustituye ese resultado por otro igual al máximo obtenido en este heurístico, por ejemplo 16.56%, la media sería de 7.37% y la desviación típica de 6.87%. Lo que significa que seguiría siendo elevada tanto la media como la desviación típica. Además,

si se observa la tabla 48, se puede ver cómo en los casos 3 y 4 se alcanza el óptimo para H9.

De modo que se pueden afirmar dos cosas:

- H9 es un heurístico muy volátil
- Aun siendo un heurístico muy volátil, con H9 se obtienen mejores resultados que en orden creciente.

5.1.5 H8 vs H10. Exclusividad más restrictiva

El problema de los dos heurísticos anteriores es que han realizado una asignación de prioridades sobre demasiadas piezas (al contrario que H3 y H4, donde se hacía una asignación sobre un número escaso de piezas). De modo que tanto en H8 (creciente) como H10 (decreciente) se busca crear heurísticos más restrictivos, para lo cual se hace una asignación sobre aquellas piezas que posean más de una tarea exclusiva.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H8	7.90	0.00	9.88	0.00	2.26	10.31	7.34	3.48	89.69	-	14.54	26.83	2
H10	1.03	3.39	9.88	0.00	3.29	10.31	5.93	2.82	5.62	0.00	4.70	3.73	5

Tabla 50: Comparación H8 vs H10

Al igual que en la comparación anterior, en esta, se obtiene un mayor número de resultados mejores cuando la asignación se hace en orden decreciente (5 frente a 2). Sin embargo es necesario analizar los resultados de H8 porque existe un *outlier* que hace que la media sea excesivamente elevada.

Si se sustituyese el valor del caso9 por el más elevado del mismo heurístico, se obtendría una media de 6.05% (32 min) y una desviación típica de 4.53% (24 min), lo que significa que seguirían siendo malos resultados.

De igual forma que en la comparación anterior, comparando estos dos heurísticos se pueden afirmar dos cosas:

- H10 es un heurístico muy volátil
- A pesar de que sea muy volátil, se siguen obteniendo mejores resultados asignando un orden decreciente que con uno creciente.

5.1.6 H11 vs H12. Duración agregada sobre la mitad de piezas

En este caso se ha realizado una asignación de prioridades a la mitad de las piezas tanto en orden creciente (H11) como en orden decreciente (H12).

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H11	0.00	3.95	25.87	0.85	1.32	-	5.17	-	0.00	-	5.31	9.28	4
H12	1.69	7.34	0.38	0.38	1.60	3.83	0.00	1.03	1.97	1.13	1.93	2.19	6

Tabla 51: Comparación H11 vs H12

En estos heurísticos se vuelve a obtener un mayor número de casos con mejores resultados en orden decreciente que en creciente (6 frente a 4), además para H11 ha habido hasta 3 casos en los que no se ha alcanzado ninguna solución.

Por otro lado, se observa que en H12 se ha obtenido una media muy buena así como una desviación típica reducida, haciendo que éste sea uno de los mejores heurísticos, como se verá más adelante.

5.1.7 H13 vs H14. Amplitud agregada

Hasta ahora se ha utilizado el criterio de duración media agregada para asignar las prioridades a las distintas piezas. Sin embargo, como se comentó anteriormente, existe la posibilidad de asignar prioridad según la amplitud agregada de las tareas. Así pues, H13 asigna una prioridad en orden creciente mientras que H14 lo hace en orden decreciente.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H13	6.68	7.71	4.80	4.80	3.86	6.93	7.71	6.49	8.53	9.22	6.67	1.74	10
H14	17.50	18.63	9.97	7.71	10.91	11.59	18.16	14.11	13.50	11.67	13.37	3.71	0

Tabla 52: Comparación H13 vs H14

En esta comparación se puede ver cómo claramente se obtienen mejores resultados en el 100% de los casos cuando se emplea el heurístico que asigna prioridades en orden creciente bajo el criterio de amplitud agregada. De hecho, la media de H13 es inferior a la mitad de la media de H14 y además la desviación típica de H13 es reducida y también es inferior a la de H14.

5.1.8 H16 vs H17. Repetitividad de las piezas

En este apartado se comparan los resultados obtenidos en los heurísticos que asignan prioridades a las piezas que más se repiten en cada secuencia, tanto en orden creciente (H16) como en orden decreciente (H17).

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H16	0.38	0.38	0.38	0.38	0.00	3.10	0.09	0.38	0.00	0.00	0.51	0.93	8
H17	3.20	6.02	5.17	1.98	0.00	3.10	3.20	3.20	3.09	1.13	3.01	1.75	0

Tabla 53: Comparación H16 vs H17

Como se puede observar, existen 2 casos en los que se obtienen los mismos resultados, pero en el resto de casos se obtienen mejores resultados en todos ellos cuando se asignan prioridades en orden creciente (8 frente a 0).

Además, el heurístico H16 es el mejor de todos los que se ha probado. Pues como se puede observar, la media es muy baja y la desviación típica es inferior a 1.

5.1.9 Conclusiones

Analizando la cantidad de casos en los que se obtienen mejores o peores resultados en orden creciente frente a decreciente, se puede concluir que se obtienen mejores resultados en más casos cuando se asignan prioridades en sentido creciente. En concreto se obtienen en 43 casos frente a 27, lo que significa 1.59 veces más.

Sin embargo, esto no quiere decir que siempre sea mejor aplicar un sentido creciente, sino que como se ha visto, depende del criterio que se elija para asignar prioridades. En concreto, se pueden extraer la siguientes conclusiones basándose en el mencionado parámetro N:

- Mejores resultados en orden creciente en los siguientes casos:
 - Ordenación de todas las piezas bajo criterio de duración agregada (H1 vs H2)
 - Ordenación bajo el criterio de no-exclusividad (H3 vs H5 y H4 vs H6)
 - Ordenación de todas las piezas bajo el criterio de amplitud agregada (H13 vs H14)
 - Ordenación sobre todas las piezas salvo las que se repiten menos (H16 vs H17)
- Mejores resultados en orden decreciente en los siguientes casos:
 - Ordenación bajo el criterio de exclusividad (H7 vs H9 y H8 vs H10)
 - Ordenación sobre la mitad del total de las piezas (H11 vs H12)
- Cuando se asignan prioridades a demasiadas piezas, los resultados empeoran o incluso no llegan a solución

- Cuando se emplea un criterio demasiado restrictivo no permite su aplicación sobre el 100% de los casos

5.2 Exclusividad vs no exclusividad

A continuación se van a comparar los heurísticos equivalentes en los que se ha seguido un criterio de exclusividad y no exclusividad por parejas y bajo el mismo criterio de asignación de prioridades en cuanto a la duración agregada de las tareas.

5.2.1 H3 vs H7. Orden creciente

En ambos heurísticos se ha llevado a cabo una asignación de las prioridades en sentido creciente bajo el criterio de duración agregada de las tareas. Pero en H3 se han priorizado aquellas piezas que no poseían ninguna tarea exclusiva frente a H7, donde se a priorizado aquellas que poseían como mínimo una tarea exclusiva.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H3	0.38	-	15.80	0.38	5.64	-	0.85	0.85	-	4.23	4.02	5.60	6
H7	2.54	12.14	0.38	7.06	6.49	6.93	-	-	8.53	-	6.29	3.86	4

Tabla 54: Comparación H3 vs H7

Aquí se puede ver la demostración del problema que tienen H3 y H7. Cuando se emplea H3 existe el inconveniente de que hay varios casos en los que no se puede aplicar el heurístico, al contrario que sucede con H7, cuyo problema es que asigna prioridades a demasiadas piezas.

5.2.2 H4 vs H8. Orden creciente con distintas restricciones

Aquí se puede observar cómo con H4, el heurístico que prioriza la no exclusividad, se obtienen mejores resultados frente al que prioriza la exclusividad (8 frente a 2). Ambos heurísticos asignan prioridades en sentido creciente bajo el criterio de la duración agregada de cada tarea pero con distintas restricciones para no saturar ni carecer de asignación de piezas con prioridades

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H4	0.85	2.45	1.22	0.38	0.85	0.73	1.88	0.38	0.94	9.22	1.89	2.66	8
H8	7.90	0.00	9.88	0.00	2.26	10.31	7.34	3.48	89.69	-	14.54	26.83	2

Tabla 55: Comparación H4 vs H8

5.2.3 H5 vs H9. Orden decreciente

En esta comparación se sigue la línea de la comparación H3 vs H7 pero con una asignación de prioridades en sentido decreciente.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H5	6.30	-	15.80	0.56	7.62	-	5.64	0.38	-	3.39	5.67	5.26	3
H9	1.32	9.13	0.00	0.00	4.14	11.59	107.62	0.94	13.50	16.56	16.48	32.59	7

Tabla 56: Comparación H5 vs H9

Como se puede observar, a diferencia de la comparación anteriormente mencionada, aquí el heurístico que prioriza la exclusividad obtiene mejores resultados que el que prioriza la no-exclusividad (7 frente a 3).

5.2.4 H6 vs H10. Orden decreciente con distintas prioridades

La diferencia de esta comparación frente a la que se hace con H4 vs H8 es que en esta, ambos heurísticos siguen una asignación de prioridades en sentido decreciente.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H6	0.38	16.18	1.22	14.58	0.75	5.02	6.02	1.22	10.87	11.67	5.75	6.18	5
H10	1.03	3.39	9.88	0.00	3.29	10.31	5.93	2.82	5.62	0.00	4.70	3.73	5

Tabla 57: Comparación H6 vs H10

Según estos resultados no se puede sacar ninguna conclusión sobre cuál de los dos heurísticos es mejor ya que obtienen el mismo número de casos con mejores resultados, 5 en los dos heurísticos. Sin embargo, cabe destacar que H10 tiene una media inferior que H6, así como una desviación típica también inferior y además, en dos casos consigue obtener un error del 0%. De modo que se podría decir que H10 es mejor que H6.

5.2.5 Análisis

Basándose en los resultados de las comparaciones anteriores a primera vista parece que es difícil determinar un criterio único con el que se puedan obtener los mejores resultados. Pero si se hace una comparación más, según se muestra en la siguiente tabla, se puede observar que en caso de ordenar las piezas en sentido creciente, se obtienen mejores resultados cuando se priorizan las piezas exclusivas.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T
H3	0.38	-	15.80	0.38	5.64	-	0.85	0.85	-	4.23	4.02	5.60
H4	0.85	2.45	1.22	0.38	0.85	0.73	1.88	0.38	0.94	9.22	1.89	2.66
H7	2.54	12.14	0.38	7.06	6.49	6.93	-	-	8.53	-	6.29	3.86
H8	7.90	0.00	9.88	0.00	2.26	10.31	7.34	3.48	89.69	-	14.54	26.83

Tabla 58: Comparación del criterio de exclusividad en caso de una ordenación creciente

En cambio, si se analizan los resultados en caso de una ordenación en sentido decreciente, se observa que los resultados son similares (incluso en el caso de que no apareciesen los *outliers*). Sin embargo, se puede observar que se obtienen 4 casos con un error del 0% cuando se prioriza la exclusividad mientras que no se obtienen ninguno cuando se prioriza la no-exclusividad.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T
H5	6.30	-	15.80	0.56	7.62	-	5.64	0.38	-	3.39	5.67	5.26
H6	0.38	16.18	1.22	14.58	0.75	5.02	6.02	1.22	10.87	11.67	6.79	6.18
H9	1.32	9.13	0.00	0.00	4.14	11.59	107.62	0.94	13.50	16.56	16.48	32.59
H10	1.03	3.39	9.88	0.00	3.29	10.31	5.93	2.82	5.62	0.00	4.23	3.73

Tabla 59: Comparación del criterio de exclusividad en caso de una ordenación decreciente

A continuación se muestra una tabla que recoge los heurísticos involucrados en el criterio de la exclusividad.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	Media
H3	0.38	-	15.80	0.38	5.64	-	0.85	0.85	-	4.23	4.02	5.60	4.59
H4	0.85	2.45	1.22	0.38	0.85	0.73	1.88	0.38	0.94	9.22	1.89	2.66	
H5	6.30	-	15.80	0.56	7.62	-	5.64	0.38	-	3.39	5.67	5.26	
H6	0.38	16.18	1.22	14.58	0.75	5.02	6.02	1.22	10.87	11.67	6.79	6.18	
H7	2.54	12.14	0.38	7.06	6.49	6.93	-	-	8.53	-	6.29	3.86	10.38
H8	7.90	0.00	9.88	0.00	2.26	10.31	7.34	3.48	89.69	-	14.54	26.83	
H9	1.32	9.13	0.00	0.00	4.14	11.59	107.62	0.94	13.50	16.56	16.48	32.59	
H10	1.03	3.39	9.88	0.00	3.29	10.31	5.93	2.82	5.62	0.00	4.23	3.73	

Tabla 60: Comparación de los heurísticos que priorizan la exclusividad frente a los que priorizan la no-exclusividad

Se puede observar que en 6 casos en los que se prioriza la exclusividad se consigue un error del 0%. Pero por otro lado, hasta en 6 casos se consiguen mejores resultados cuando se prioriza la no-exclusividad frente a 4 y además el error medio en los heurísticos que priorizan la no-exclusividad (4.59%) es mejor que en los de exclusividad (10.38%). Si se analiza la media del error de cada caso como se muestra en la siguiente tabla, se puede observar que en 7 casos se obtiene una media mejor cuando se prioriza la no exclusividad frente a 3.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10
No-Ex	1.98	9.31	8.51	3.97	3.72	2.87	3.60	0.71	5.90	7.13
Ex	3.20	6.16	5.03	1.76	4.05	9.79	40.29	2.41	29.33	8.28

Tabla 61: Comparación de los errores medios de cada caso bajo el criterio de exclusividad y no-exclusividad

5.2.6 Conclusiones

De modo que se llegan a las siguientes conclusiones

- Cuando se prioriza la exclusividad se obtienen *outliers* tanto en sentido creciente como decreciente y los resultados son muy volátiles, pues en estos heurísticos se han obtenido los dos casos con peores resultados y hasta 6 casos con errores del 0%
- El criterio de no-exclusividad es más robusto que el de exclusividad
- Ante una ordenación en sentido creciente, se obtienen mejores resultados cuando se prioriza la no-exclusividad

5.3 *Duración agregada vs amplitud*

Este parámetro es difícil de comparar porque no es aplicable en todos los casos ya que solo se diferencia por el orden entre dos piezas y no en todos los heurísticos aparecen ambas piezas.

De hecho, tan solo es posible aplicar este criterio en la ordenación total de todas las piezas tanto en sentido creciente como decreciente y en 3 casos más, que se corresponden con el H7_2, H9_2 y H9_5

5.3.1 H1 vs H13. Orden creciente

Comparación en orden creciente entre la asignación bajo el criterio de duración media agregada (H1) y amplitud agregada (H13)

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H1	13.55	12.14	4.80	4.80	14.11	6.93	14.30	7.62	8.53	9.22	9.60	3.70	0
H13	6.68	7.71	4.80	4.80	3.86	6.93	7.71	6.49	8.53	9.22	6.67	1.74	6

Tabla 62: Comparación H1 vs H13

Como se puede observar, en todos los casos en los que se obtienen resultados diferentes, éstos son mejores cuando se aplica H13.

5.3.2 H2 vs H14. Orden decreciente

A diferencia de cuando se ordena en sentido creciente, se obtienen mejores resultados con una ordenación basada en el criterio de duración agregada decreciente.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	D.T	N
H2	10.25	9.13	9.97	7.71	12.51	11.59	10.82	9.41	13.50	11.67	10.66	1.72	5
H14	17.50	18.63	9.97	7.71	10.91	11.59	18.16	14.11	13.50	11.67	13.37	3.71	1

Tabla 63: Comparación H2 vs H14

5.3.3 Conclusiones

Lo ideal hubiera sido poder crear más heurísticos basados en el criterio de la amplitud agregada, pero no ha sido posible porque con los heurísticos creados y con los datos proporcionados, como se ha mencionado ya, tan solo ha sido posible comparar 3 casos, cuyos resultados se muestran a continuación.

Error [%]	Caso2
H_amp	7.71
H7	64.50

Tabla 64: Comparación caso2 bajo criterio de amplitud vs duración agregada

Al igual que sucedía antes, y como se observa en la tabla anterior, cuando se aplica el criterio de la amplitud agregada en un caso de ordenación en sentido creciente, se obtienen resultados mejores. En concreto en este caso el resultado es casi 10 veces mejor.

Error [%]	Caso2	Caso5
H_amp	18.63	6.49
H9	9.13	4.14

Tabla 65: Comparación caso2 y caso5 bajo criterio de amplitud vs duración agregada

En este caso vuelve a suceder lo que se había observado en la comparación H2 vs H14, los resultados empeoran ante una ordenación en sentido decreciente bajo el criterio de amplitud agregada.

Se llegan a las siguientes conclusiones:

- Ante una asignación de prioridades en sentido creciente, se obtienen mejores resultados bajo el criterio de amplitud agregada en el 100% de los casos comprobados.
- En cambio, ante una asignación de prioridades en sentido decreciente, se obtienen peores resultados bajo el criterio de amplitud agregada en el 100% de los casos comprobados.

5.4 Cubas equivalentes

A continuación se comentan los resultados de la comparación entre los heurísticos H18 y H19.

Mejora [min]	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10
H18	47.5	60.5	62.5	76.5	55	14	4	111	46.5	2
H19	78	8	58	93.5	90	-10.5	32	76	18.5	11

Tabla 66: Comparación H18 vs H19

H18 aplica el mismo criterio que H16, que como se comentará más adelante, es el mejor heurístico de todos los que se ha probado; y H19 aplica el mismo criterio que H14, pues es el peor heurístico de los analizados. Por tanto se esperaba que la mejora de H18 frente H19 fuese mayor en todos los casos, sin embargo, como se muestra en la tabla 65 (donde se representa la mejora en minutos de cada caso respecto al óptimo en el que no había cubas equivalentes), las mayores mejoras se producen indistintamente en en los dos heurísticos. No obstante hay que recalcar que en el caso6 del heurístico H19 es el único en el que empeoran los resultados.

5.4.1 Conclusiones

Los resultados obtenidos en el proyecto tras el análisis sin cubas equivalentes no son aplicables ante la existencia de cubas equivalentes. Ya que si no fuera tal caso, con H18 se deberían haber obtenido mejores resultados que con H19 en el 100% de los casos, y no ha sucedido así.

5.5 Priorización entre piezas del mismo tipo

H20 resulta ser uno de los heurísticos analizados que proporcionan mejores resultados. Esto se debe a que se reduce la carga computacional asignando prioridades que a priori parecen innecesarias para el usuario, pero que ayudan al programa a reducir considerablemente el número de operaciones a realizar. Es decir, reduce el tiempo sin perder optimalidad ya que no asignan ninguna prioridad que no sea natural.

5.6 Resumen resultados

A continuación se muestra una tabla con los resultados, que contiene el error de todos los heurísticos para cada caso expresado en tanto por ciento respecto al óptimo, la media de cada heurístico y la desviación típica de los mismos. No se incluyen los heurísticos H18, H19 porque no tiene sentido compararlo con el resto de heurísticos.

Error %	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	DT
H1	13.55	12.14	4.80	4.80	14.11	6.93	14.30	7.62	8.53	9.22	9.60	3.70
H2	10.25	9.13	9.97	7.71	12.51	11.59	10.82	9.41	13.50	11.67	10.66	1.72
H3	0.38	-	15.80	0.38	5.64	-	0.85	0.85	-	4.23	4.02	5.60
H4	0.85	2.45	1.22	0.38	0.85	0.73	1.88	0.38	0.94	9.22	1.89	2.66
H5	6.30	-	15.80	0.56	7.62	-	5.64	0.38	-	3.39	5.67	5.26
H6	0.38	16.18	1.22	14.58	0.75	5.02	6.02	1.22	10.87	11.67	5.75	6.18
H7	2.54	12.14	0.38	7.06	6.49	6.93	NO SOL	NO SOL	8.53	NO SOL	6.29	3.86
H8	7.90	0.00	9.88	0.00	2.26	10.31	7.34	3.48	89.69	-	14.54	26.83
H9	1.32	9.13	0.00	0.00	4.14	11.59	107.62	0.94	13.50	16.56	16.48	32.59
H10	1.03	3.39	9.88	0.00	3.29	10.31	5.93	2.82	5.62		4.70	3.73
H11	0.00	3.95	25.87	0.85	1.32	NO SOL	5.17	NO SOL	0.00	NO SOL	5.31	9.28
H12	1.69	7.34	0.38	0.38	1.60	3.83	0.00	1.03	1.97	1.13	1.93	2.19
H13	6.68	7.71	4.80	4.80	3.86	6.93	7.71	6.49	8.53	9.22	6.67	1.74
H14	17.50	18.63	9.97	7.71	10.91	11.59	18.16	14.11	13.50	11.67	13.37	3.71
H15	10.82	9.31	8.47	6.40	16.46	6.11	16.27	7.53	11.81	11.19	10.44	3.68
H16	0.38	0.38	0.38	0.38	0.00	3.10	0.09	0.38	0.00	0.00	0.51	19.42
H17	3.20	6.02	5.17	1.98	0.00	3.10	3.20	3.20	3.09	1.13	3.01	0.00
H20	5.36	2.63	0.00	0.00	0.00	1.28	4.99	2.73	1.59	0.85	1.94	1.98

Tabla 67: Error de cada heurístico para cada caso en %

En la siguiente tabla se muestra la diferencia que existe en cada caso respecto al óptimo, pero expresados en minutos, pues el tiempo es el recurso más importante de este proyecto.

Diferencia min	Caso1	Caso2	Caso3	Caso4	Caso5	Caso6	Caso7	Caso8	Caso9	Caso10	Media	DT
H1	72	65	26	26	75	38	76	41	45	49	51	20
H2	55	49	53	41	67	64	58	50	72	62	57	9
H3	2	-	84	2	30	-	4	4	-	23	21	30
H4	4	13	7	2	4	4	10	2	5	49	10	14
H5	34	-	84	3	41	-	30	2	-	18	30	28
H6	2	86	7	78	4	28	32	7	58	62	31	33
H7	14	65	2	38	35	38	NO SOL	NO SOL	45	NO SOL	34	21
H8	42	0	53	0	12	57	39	19	479		78	143
H9	7	49	0	0	22	64	572	5	72	88	88	173
H10	6	18	53	0	18	57	32	15	30		25	20
H11	0	21	138	4	7	NO SOL	28	NO SOL	0	NO SOL	28	49
H12	9	39	2	2	9	21	0	6	11	6	10	12
H13	36	41	26	26	21	38	41	35	45	49	36	9
H14	93	99	53	41	58	64	97	75	72	62	71	20
H15	58	50	45	34	88	34	87	40	63	60	56	19
H16	2	2	2	2	0	17	1	2	0	0	3	5
H17	17	32	28	11	0	17	17	17	17	6	16	9
H20	29	14	0	0	0	7	27	15	8	4	10	10

Tabla 68: Error de cada heurístico para cada caso en minutos

En la siguiente tabla se muestran los heurísticos en orden creciente según la media de desviación en minutos respecto al óptimo.

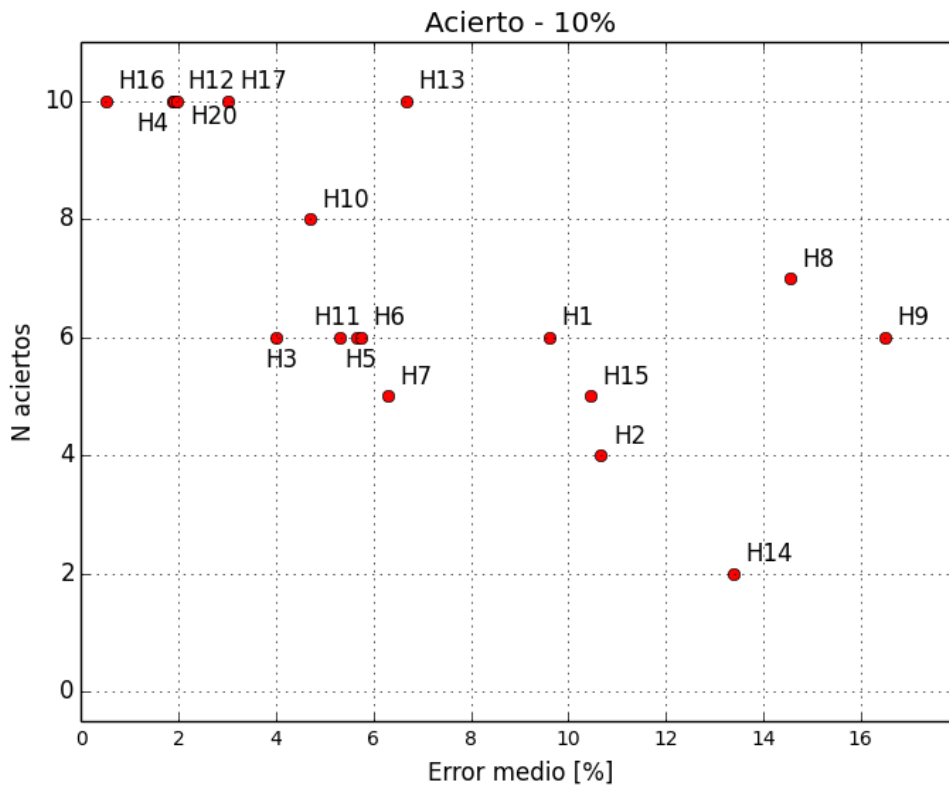
POSICION	Heurístico	MEDIA
1	H16	3
2	H4	10
3	H20	10
4	H12	10
5	H17	16
6	H3	21
7	H10	25
8	H11	28
9	H5	30
10	H6	31
11	H7	34
12	H13	36
13	H1	51
14	H15	56
15	H2	57
16	H14	71
17	H8	78
18	H9	88

Tabla 69: Posiciones de los heurísticos según la desviación media en minutos de los mismos

Con el objetivo de expresar los resultados de una forma más visual, a continuación se muestran cuatro gráficas (desarrolladas con la ayuda de Python) de forma que en el eje horizontal se representa el error medio de cada heurístico y en el eje vertical el número de aciertos. Se entiende número de aciertos como el número de casos en los que se ha obtenido un error inferior al error límite que fije el usuario.

Un error del 10% puede suponer un empeoramiento de hasta 50 minutos respecto al óptimo. Dado que el error límite lo marca el usuario, lo que puede ser inadmisibile para una fábrica (como el empeoramiento de 50 minutos), para otra puede ser aceptable. Por ello se han muestran dos gráficas, una con un error límite del 10% y otra con un error límite del 5%.

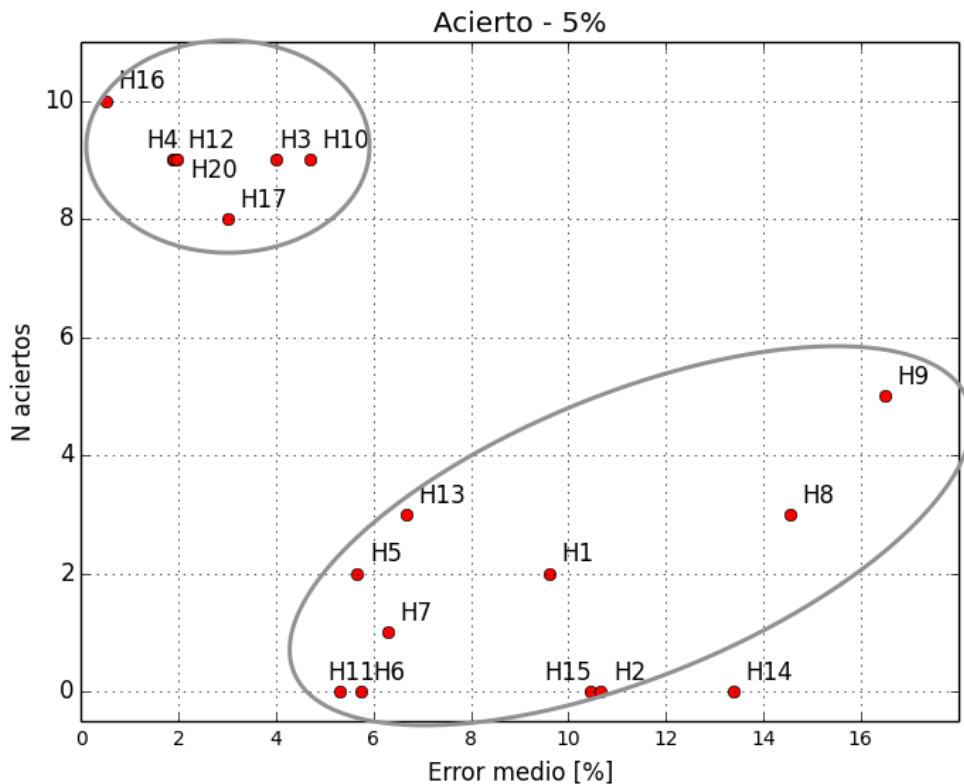
Es preciso destacar que en aquellos casos en los que no es de aplicación el heurístico, no se ha considerado que este falle. Pero en los casos en los que no se ha llegado a solución, sí se ha considerado que el heurístico falla.



Gráfica 3: Representación gráfica de los resultados con error límite 10%

Esta gráfica nos permite realizar la comparación entre todos los heurísticos para determinar cuales son mejores. Por ejemplo, entre los heurísticos H3, H11, H5, H6, H1 y H9, se aprecia cómo los 6 han obtenido el mismo número de aciertos, pero el error medio es muy diferente entre ellos.

En la siguiente gráfica se ha fijado el error límite en 5%. A diferencia de la gráfica 3, en la gráfica 4 el número de aciertos de todos los heurísticos salvo el de H16 se ve reducido. En concreto, se pasa de tener 6 heurísticos con 10 aciertos a tener tan solo un heurístico con 10 aciertos. Además en este caso se pueden diferenciar claramente dos conjuntos o *clusters* de resultados, como se ve en la siguiente imagen.

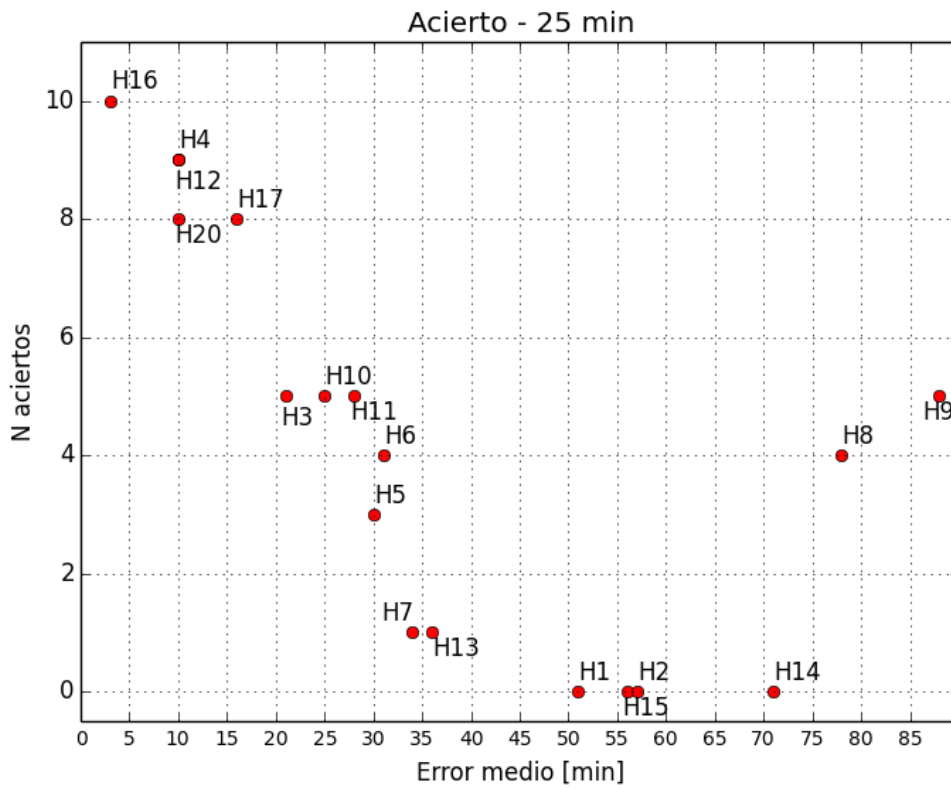


Gráfica 4: Representación gráfica de los resultados con error límite 5%

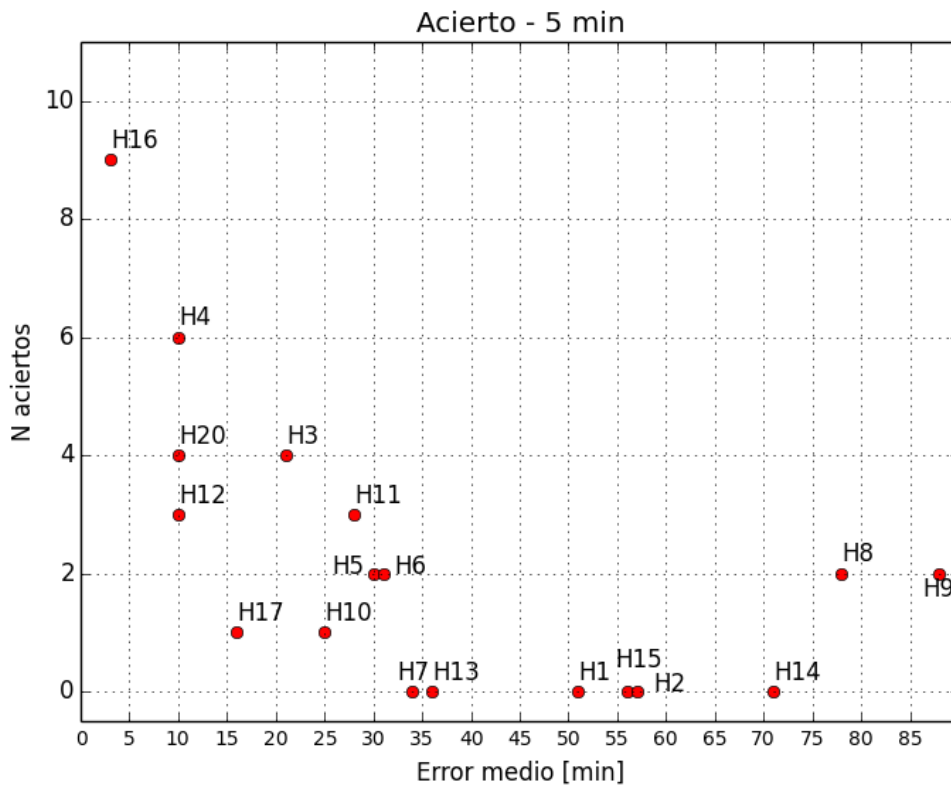
Analizando los *clusters* se puede distinguir un conjunto de heurísticos que puede considerarse bueno y otro malo. En concreto, el *cluster* superior izquierdo es el que contiene los heurísticos con mejores resultados. Analizando dicho cluster se podría hacer una primera distinción entre los heurísticos con buenos resultados y malos. De modo que H3, H4, H10, H12, H16, H17 Y H20 serían los mejores heurísticos; además se observa que tan solo 2 de los 6 heurísticos asignan una ordenación en sentido creciente mientras que el resto lo hace en sentido decreciente.

Por otro lado, dos de estos heurísticos que se consideran buenos son los que se centran en la repetitividad de las piezas mientras que solo hay uno de los que se centran en la exclusividad; 2 de los que se centran en la no exclusividad; y uno que se centra en asignar prioridades sobre la mitad de las piezas.

Del mismo modo que antes se representaron los resultados expresados tanto en porcentaje como en minutos, a continuación se representa una gráfica similar a las dos anteriores, donde en lugar de fijar el error límite, se fija una desviación límite en minutos.



Gráfica 5: Representación gráfica de los resultados con tiempo límite 25 min



Gráfica 6: Representación gráfica de los resultados con tiempo límite 5 min

Por último, observando las gráficas 3, 4, 5 y 6 se puede concluir que los dos mejores heurísticos son H16 y H4. Siendo H16 el mejor de todos los heurísticos analizados, donde tan solo se obtiene un fallo con una diferencia límite de 5 minutos; en concreto dicho error se debe a que en el caso6 (tabla 67) se obtiene una diferencia de 17 minutos.

Asimismo, si se observa la tabla 68 se aprecia que la peor media se obtiene con H9, pero en este caso existía la presencia de un *outlier* que incrementa notablemente la media, al igual que sucede con H8 que es el heurístico con la segunda peor media de todas. Si se sustituye el valor de dichos *outliers* por el máximo valor obtenido en cada uno de los dos heurísticos, la peor media se obtiene con H14, de modo que podría considerarse H14 como el peor heurístico de todos los analizados.

Se puede reducir la carga computacional en gran medida mediante la asignación de prioridades que a priori puede parecer que carecen de utilidad, como se ve en el caso de H20; pues los resultados de este heurístico están entre los mejores.

Para concluir, es necesario comentar que estos resultados solamente son de aplicación en aquellos casos en los que no se habilite el uso de cubas equivalentes ya que como se ha comprobado, los resultados no son robustos cuando se habilitan dichas cubas

6 FUTUROS DESARROLLOS

- Análisis del comportamiento de los heurísticos con cubas equivalentes. Pues como se ha demostrado, los resultados obtenidos del análisis de los heurísticos de este proyecto no son aplicables ante la existencia de cubas equivalentes.
- Análisis del impacto de un brazo transportador más rápido o más lento, pues existen tareas lo suficientemente cortas como para que no de tiempo a que el brazo pueda ir a por otras piezas
- Análisis de los resultados ante la fijación de unos tiempos de traslados acordes con el *layout* o disposición de la planta
- De la mano de lo anterior, análisis del impacto de los diferentes *layouts* que consiguen reducir los tiempos de traslado de forma que se realicen recorridos lo más cortos posibles

7 BIBLIOGRAFÍA

- [1] DANIEL PEÑALOZA [2009]. Procesos especiales para el sector aeronáutico. <http://www.measurecontrol.com/procesos-especiales-para-el-sector-aeronautico/>
- [2] Apuntes UPC. Selección y diseño de sistemas productivos.
- [3] Albert Jones, Ph.D. [1998]. Survey of Job Shop Scheduling Techniques.
- [4] Christoph J. Schuster [2004]. No-wait job shop scheduling: tabu search and complexity of subproblems .
- [5] Jen Shian Chen [2006]. Model formulations for the machine scheduling problem with limited waiting time constraints.
- [6] Simon French, B.A., M.A., D.Phil. [1982]. Sequencing and scheduling: an introduction to the mathematics of the Job-Shop.
- [7] Peter Lennartz [2006]. No wait job shop scheduling: a constraint propagation approach.
- [8] Samarghandi, ElMekkawy and Al-Mehdi [2012]. Studying the effect of different combinations of timetabling with sequencing algorithms to solve the no-wait job shop scheduling problem.
- [9] <https://www.gams.com>
- [10] http://www.valoryempresa.com/archives/tutoriales/diagGantt_1.htm

